Sensor Data Fusion for Improving Traffic Mobility in Smart Cities

Kennedy John Offor

Department of Automatic Control and Systems Engineering
University of Sheffield

A Thesis submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

July 2020

I dedicate this thesis to God Almighty
and
My family for their support and prayers.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this thesis are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This thesis is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

Kennedy John Offor
July 2020

Acknowledgements

First and foremost, I would like to appreciate God who has been with me from the inception till now through the intercession of the Blessed Virgin Mary.

Next, my profound gratitude goes to my supervisor, Prof. Lyudmila S. Mihaylova who has been instrumental to the formulation of the research topic and continued support from beginning till now. Without her guide, corrections, and financial support, I would not have been able complete this studies.

My gratitude also goes to my second supervisor, Prof. Mahdi Mafouf, for his guidance and support.

I will not forget to mention Dr Matthew Hawes, Lubos Vaci, Peng Wang for their readiness to offer technical support. Your constructive criticisms has been instrumental to the successful completion of this research.

My deepest appreciation also goes to my sponsors, Chukwuemeka Odumegwu Ojukwu University and Tertiary Education Trust Fund, whose financial grant made it possible for me to embark on this study.

Thank you to my parents late Mr Nzomiwu Offordum and Mrs Rita Nwanamba Offordum, whose love and care nurtured me to this point. My gratitude also goes to my eldest sister, Mrs Fidelia Chinasa Chigbu without whom I would have stopped my education after primary school. I owe you every of my educational achievements. Thank you big sister. I also appreciate my second sister, Mrs Ann Ibuodinma and my brother Charles Offor. Your support and care made me who I am today.

Lastly, I want to appreciate my dearest wife, Mrs Maureen Ogoo Offor for her endless love, care and support when the going became tough. You took care of the family needs to enable me focuss on my studies. Thank you also to my daughter Chinwendu M. Offor (Ada Engineer) and my son Chidiebube J. Offor (Nwa Mummy) for praying for me always and putting up with my short absense under the care of your mother alone before joining me.

My special appreciation goes to those I have not mentioned their names here. May God reward each of you for the various help you rendered to me.

Abstract

The ever-increasing urban population and vehicular traffic without a corresponding expansion of infrastructure have been a challenge to transportation facilities managers and commuters. While some parts of transportation infrastructure have big data available, so many other locations have sparse data. This has posed a challenge in traffic state estimation and prediction for efficient and effective infrastructure management and route guidance. This research focused on traffic prediction problems and aims to develop novel spatial-temporal and robust algorithms, that can provide high accuracy in the presence of both big data and sparse data in a large urban road network.

Intelligent transportation systems require the knowledge of current traffic state and forecast for effective implementation. The actual traffic state has to be estimated as the existing sensors do not capture the needed state. Sensor measurements often contain missing or incomplete data as a result of communication issues, faulty sensors or cost leading to incomplete monitoring of the entire road network. This missing data pose challenges to traffic estimation approaches. In this work, a robust spatio-temporal traffic imputation approach capable of withstanding high missing data rate is presented. A particle-based approach with Kriging interpolation is proposed. The performance of the particle-based Kriging interpolation for different missing data ratios was investigated for a large road network.

A particle-based framework for dealing with missing data is also proposed. An expression of the likelihood function is derived for the case when the missing value is calculated based on Kriging interpolation. With the Kriging interpolation, the missing values of the measurements are predicted, which are subsequently used in the computation of likelihood terms in the particle filter algorithm.

In the commonly used Kriging approaches, the covariance function depends only on the separation distance irrespective of the traffic at the considered locations. A key limitation of such an approach is its inability to capture well the traffic dynamics and transitions between different states. This thesis proposes a Bayesian Kriging approach for the prediction of urban traffic. The approach can capture these dynamics and model changes via the covariance

matrix. The main novelty consists in representing both stationary and non-stationary changes in traffic flows by a discriminative covariance function conditioned on the observation at each location. An advantage is that by considering the surrounding traffic information distinctively, the proposed method is very likely to represent congested regions and interactions in both upstream and downstream areas.

Contents

List of Figures

List of Tables

List of Symbols and Abbreviations

$\mathbf{b}, \mathbf{g}$	Vector of Interpolation and Interpolating Points
$\mathbf{C}$	Matrix of Interpolating Points
ξ	Observation Error
η	State Error
$E(.)$	Mathematical Expectation
ε	Residual (Error)
Γ	Gamma Function
γ	Variogram
h	Lag, Separation distance between two locations
k	Time Index
μ	Mean Value
z	Observation or Measurement
$\mathscr{N}$	Normal Distribution
n	Number of Segments (Cells)
ν	Degrees of freedom
θ	Learning/scaling parameter
ω	Weight
q, Q	Traffic Flow

R	Observation Noise Covariance Matrix
$\mathbf{r}$	Location Vector
ρ	Traffic Density
λ	Lagrange Multiplier
s	Sampling Time Index
t	Time
υ	Innovation Error
v	Traffic Speed
x	State

Acronyms / Abbreviations

ANN	Artificial Neural Network
APE	Absolute Percentage Error
ARIMA	Auto Regressive Integrated Moving Average
ARND	Approximate Road Network Distance
ATIS	Advanced Traveller Information System
ATMS	Advanced Traffic Management System
BEM	Basic Ensemble Method
CBMD	Column-Based Matrix Decomposition
CCC	Cophenetic Correlation Coefficient
CNN	Convolutional Neural Networks
CTM	Cell Transmission Model
DBN	Deep Belief Networks
DNN	Deep Neural Networks
EKF	Ensemble Kalman Filter

ENN	Ensemble Neural Network
FHWA	Federal Highway Administration
FOA	Fruitfly Optimization Algorithm
GEM	Generalized Ensemble Method
GMRF	Gaussian Markov Random Field
GMRF	Gaussian Markov random field
GPS	Global Positioning Systems (or Satellites)
GSPF	Gaussian Sum Particle Filter
HCA	Hierarchical Cluster Analysis
HCA	Hierarchical Cluster Analysis
ITS	Intelligent Transportation System
KF	Kalman Filter
kNN	k Nearest Neighbour
LSSVM	Least Squares Support Vector Machine
LSSVR	Least Squares Support Vector Regression
LWR	Lighthill-Whitham-Richards
MA	Moving Average
MAE	Mean Absolute Error
MCK	Markov-Cube Kriging
MMBK	Multi-Model Bayesian Kriging
MRF	Markov Random Field
NLP	Natural Language Processing
NMF	Non-negative Matrix Factorisation
NRMSE	Normalised Root Mean Squared Error

PCA	Principal Component Analysis
PCU	Passenger Car Unit
PDF	Probability Density Function
PF	Particle Filter
PFCS	Particle Filter with Compressive Sensing
PGSPF	Parallelized Gaussian Sum Particle Filter
PPF	Parallelized Particle Filter
PSO	Particle Swarm Optimization
RBF	Radial Basis Function
RMSE	Root Mean Squared Error
RPCA	Robust PCA
SCM	Stochastic Compositional Model
SSNN	State Space Neural Network
SUMO	Simulation of Urban Mobility
SVD	Singular Value Decomposition
SVM	Support Vector Machine
SVR	Support Vector Regression
UKF	Unscented Kalman Filter

Chapter 1

Introduction

1.1 Motivation

The ever-increasing human and vehicular population in the urban areas with limited road infrastructure is giving rise to more congestion with its associated social and economic problems. Some of these include environmental pollution, safety issues, economic loss, increase in travel time, stress and sometimes loss of life. Congestion has been a major concern to the authorities as it has been on the increase with no sign of decreasing so long as population continues to increase. According to [3] congestion was estimated to cost UK over £15 billion/year in 1998, representing 1.55% of GDP. A study released by INRIX in 2014 indicated this figure would increase to more than £300 billion (a 63% increase in annual cost) cumulatively between 2013 and 2030 [4]. The situation is the same in other countries, and the estimates were 1.3% and 0.9% of GDP for France and Germany respectively [5].

A 2009 report by The Texas Transportation Institute shows that the situation is worse in the US with an estimated cost of $87 billion [6]. This is equivalent to 0.6% of US GDP in 2007. The same report released in 2015 showed an increase of 82.3% to $159 billion in 5 years [7]. The total travel cost was 6.9 billion hours and 3.1 billion excess gallons of fuel. A more recent study conducted by INRIX in 2016 [8], covering a total of 1,064 cities in 38 countries, which was published in February 2017 shows an even worse scenario. On average, the cost of congestion in 2016 across three countries, US, UK and Germany was close to £350 billion ($450 USD) equivalent to £756 ($971) per capita. The cost for the UK alone was £30.6 billion (which averages at £968 per driver) with drivers wasting an average of 32 hours. In addition to these financial costs, the other costs are not quantified such as emotional, stress, pollution and other problems caused by delay. Surely, a situation like this requires serious effort to tackle because if left unattended, it could become worse in 50 years' time.

Congestion, which results from a mismatch between supply (road capacity) and demand (human and vehicular traffic) could be tackled in two ways [9], [5]: build more roads and/or optimize the usage of existing roads using Intelligent Transportation Systems (ITS). The first solution is more difficult to achieve as it is very costly, and there are limited available spaces for expansion, leaving the later option. ITS require knowledge of past, present and future traffic states. The past traffic state is readily available via measurements made with inductance loops, magnetic loops, video cameras, floating car data, social media data sources. The present and future traffic state could be estimated and predicted respectively using past measurements and some computer simulation and traffic modelling techniques.

One of the major challenge faced in traffic prediction is the issue of missing or sparse data. Traffic measurements are generally captured with different types of sensors and transmitted through a communication infrastructure for processing and utilization. These infrastructures are subject to failure and malfunction, occasionally leading to incomplete/missing data, sometimes more than 40% [10]. The problem of sparse data is caused by the high cost of installing and managing traffic measurement devices, making it impractical to cover all locations needed for effective observation of the full road network.

Researchers resorted to various methods and approach to address these challenges of missing/sparse data, which can typically be subdivided into model-based approaches, data-driven approaches or a combination of both. An overview of the different modelling methodologies is given in [11, 12]. These modelling methodologies include microscopic, macroscopic and mesoscopic approaches. Microscopic traffic models [11–14] describe the motion of each individual vehicle with a high level of detail.

In macroscopic models [15, 16], traffic state is represented by aggregating behaviour of the traffic, usually in terms of the average speed and the average density over a given period. Mesoscopic models [11] employs some features of the microscopic and macroscopic approaches by utilizing varying levels/degrees of detail to model traffic behaviour. This is achieved by modelling some locations with aggregated measurements as in macroscopic, and the remaining locations are modelled down to the details of individual vehicles as is done in the case of microscopic.

Macroscopic models are enough to produce acceptable estimation accuracy when compared to the computational overhead of the microscopic models. Hence, they are proffered choice for most practical purposes such as traffic control/management, road pricing and changes in infrastructure. Most traffic estimation approaches are model-based [17], while the new trend is to develop data-driven approaches [18].

Data-driven methods rely on historical data or streaming/real-time data. Within the last decade, there has been growing interest in applying Kriging for various traffic state prediction: directional traffic volume using global position system (GPS) data [19], annual average traffic count interpolation using origin-destination data [20], estimating annual average daily traffic [21, 22], traffic volume prediction [23, 24], and traffic volume imputation [25].

We are in the era of big data where traffic measurements (vehicle counts, occupancy and average speed), recorded by inductive loop detectors, floating car data, automatic vehicle identification, acoustic sensors, video/image processors, cellular geolocation systems, the Global Position System (GPS), social media and weather information are now widely available. To increase the accuracy of parameter and state estimation in this era of big data, there is a great need to fuse these homogeneous and heterogeneous data. Many pieces of research have focused on this direction for the last ten years.

1.2 Traffic Measurement and Estimation

The knowledge of historical, present and future traffic state is required for effective traffic management and control. The actual traffic state measured with sensors which often contain errors and missing data. The result of this is that the actual traffic state is not known Hence, the need for estimation and prediction In estimation, the current state is inferred from the physical model and/or current and past measurements. In prediction, the future traffic state is forecast. The research conceptual model is shown in figure 1.1.

Observations from numerous sensors and other sources are usually fused to decrease the impact of uncertainty in measurements. A comprehensive review of sensor data fusion for ITS was presented by [26] while [27] gave a conceptual and ideological review of data fusion in general. The terms multi-sensor and data fusion mean different things but sometimes used interchangeably. Multi-sensor data fusion deals with combining more than one sensor in order to obtain more accurate data of the event or object under observation. This could be necessary when the monitored events are affected by the spatial location of the sensor or when the object has features that a given sensor may not capture. Data fusion deals with the combination of data from single or multiple sources. Thus, data fusion involves combining data from different sources and events to generate more useful data, while multi-sensor fusion involves combining data from different sensors about the same object or event [27]. Multi-

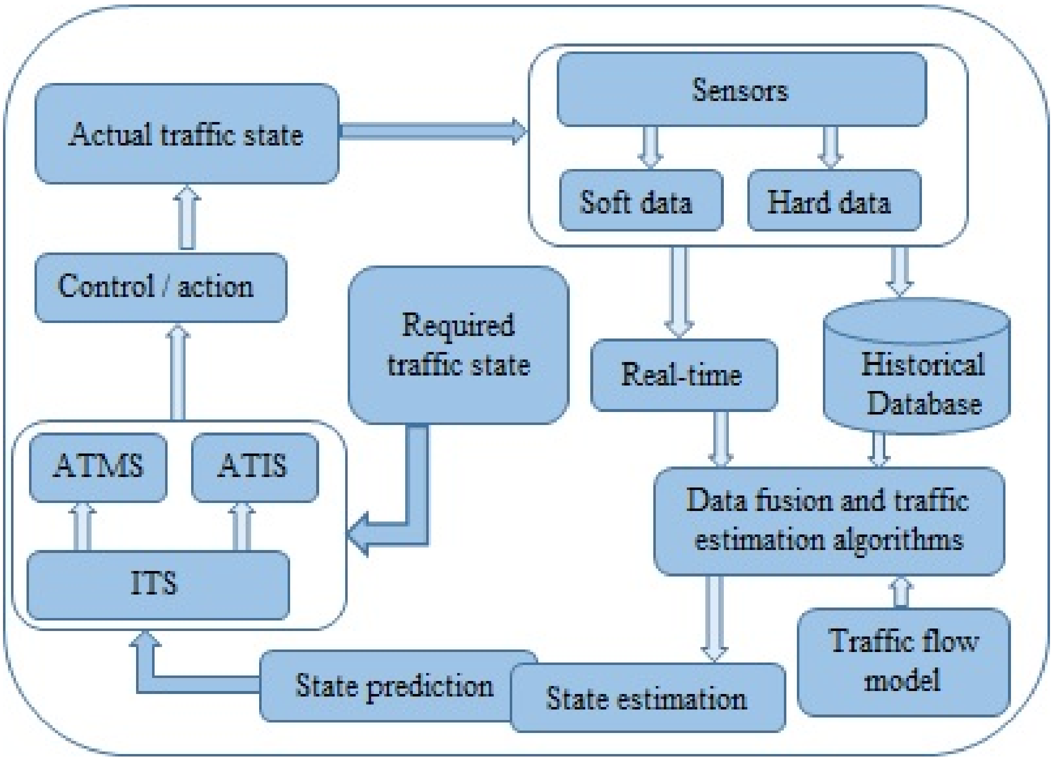

Figure 1.1 Conceptual Model

sensor data fusion is defined as a way of combining data from different sensors and events for obtaining a more representative and accurate state of the phenomena under investigation. Traffic state estimation requires observations from different locations of the road network and hence, fusing these different data sources is inevitable for improved accuracy. A review of data fusion in traffic estimation is covered in Chapter 2.

1.3 Problem Definition

As discussed in Chapter 2, several open issues are still the focus of research in traffic state estimation and prediction. Some of these challenges include the following:

1. Big and sparse data: While some parts of transportation infrastructure have “big data” available, so many other locations have sparse data. This has posed a challenge of how to design an algorithm that will give consistent performance in both scenarios of traffic state estimation and prediction for efficient and effective infrastructure management and route guidance.

2. Large-scale road networks: One of the challenges facing data fusion in ITS is how to combine the data from various sensors to handle spatial measurements taken at points and sections with temporal ones taken at discrete time step or aggregated over a period of time for consistent performance in both constrained/unconstrained flow situations and different road network configuration such as motorways, arterial/urban roads [28, 29].

3. On-line / off-line prediction: Efficient traffic control requires real-time/online prediction of traffic state. The continuous increase in historical and real-time data requires a fast algorithm that would handle big data in real-time.

Some notable works on data fusion application in traffic management include [30–38]. A comprehensive literature review carried out by [27, 26, 39, 40] indicated that there are many challenges facing data fusion, especially in traffic management. These challenges include:

i. Obtaining data with the necessary accuracy to make the application effective.

ii. Dynamic and real-time issues associated with data quality as traffic flow changes.

iii. Processing framework - whether central or decentralized.

iv. Data imperfection - sensor data are not always a correct representation of the actual state.

v. Outliers and spurious data - inconsistent readings.

vi. Data correlation - the dependence or independence of different sensor data in distributed sensing.

vii. The development of methods to combine sensor or hard data with human-generated or soft data.

viii. Conflicting data - some sensor data may conflict with others.

ix. Data modality - sensor data from different sensors (heterogeneous data such as audio, visual, text).

x. Data association - being able to associate data to a particular target, especially in multi-target tracking.

xi. Data alignment/registration - transforming different sensor data to a common frame of reference.

xii. Static vs dynamic phenomena - combining time-varying and time-invariant data.

xiii. Data dimensionality - converting the data to lower dimension for improved bandwidth utilization.

The focus of this research is the development of efficient sensor data fusion techniques and algorithms that will tackle challenges i - v above.

1.4 Aim and Objectives

The main objective of this research is the development of algorithms and approaches for online traffic state prediction and filtering in urban cities. This will be achieved via the following specific objectives:

1. To develop robust spatio-temporal responsive algorithms for traffic state prediction that could give consistent performance in the presence of both big and sparse data.
2. To develop robust prediction algorithms that will yield consistent performance in both constrained/unconstrained flow situations and different road network configuration such as motorways, arterial/urban roads by applying parallel processing.
3. To develop robust algorithms for both online/off-line prediction.
4. To validate the proposed algorithms using both synthetic and real data and hence determine the optimal parameters for different scenarios.

1.5 Key Contributions

This work builds on the existing approaches by employing a discriminative covariance model conditioned on the observation at each location. Thus, the proposed approach can account for congested regions and interactions in the upstream and downstream of the congestion. Normally, the covariance function is only dependent on the separation distance irrespective of the traffic situation at the locations. This makes it impossible for the model to capture traffic dynamics and transitions from free-flow to congested state, congested state to free-flow, etc. Our proposed approach can capture these dynamics and model it into the covariance matrix.

The main contributions of the thesis are the following:

- An approach to tackle the problem of missing and sparse data in traffic estimation is proposed. This approach entails interpolating the missing values using Kriging with a level of confidence assigned to the predicted values by computing their interpolation error variance.
- A multi-step ahead traffic estimation approach that captures the dynamic and stochastic nature of traffic using discriminative covariance functions conditioned on the data at each location is developed.
- An approach that reduces the computational overhead of large-scale road networks by using column-based matrix decomposition to select the most influential segments based on the road network is proposed. Missing measurements are then imputed using Kriging prior to particle filter measurement update step. This reduces the effect of higher missing data ratio in traffic estimation.
- A traffic estimation approach that captures the dynamic and stochastic nature of traffic using discriminative covariance functions conditioned on the data at each location is formulated.
- A robust spatio-temporal traffic imputation approach adaptive to stationary and non-stationary traffic data capable of withstanding high missing data ratio is implemented.
- A multi-model Bayesian Kriging approach is developed for traffic state estimation. Generally, a given dataset could be represented by different models. Traditional Kriging makes use of the "best" model that explains the whole dataset. This often leads to over-fitting and underfitting with different scenarios. Using a weighted average of all the models has been shown to outperform a single model. This is evident in the results.

1.6 Publications

Conference papers The main results of this PhD research are disseminated through the following peer reviewed publications. **K. J. Offor**, M. Hawes, and L. S. Mihaylova (2018). "Short Term Traffic Flow Prediction with Particle Methods in the Presence of Sparse Data," 2018 21st International Conference on Information Fusion (FUSION), Cambridge, July 2018, pp. 1185-1192. doi: 10.23919/ICIF.2018.8455496.

K. J. Offor, P. Wang, and L. S. Mihaylova (2019). "Multi-Model Bayesian Kriging for Urban Traffic State Prediction," 2019 Sensor Data Fusion: Trends, Solutions, Applications (SDF), Bonn, Germany, 2019, pp. 1-6. doi: 10.1109/SDF.2019.8916655

Journal Papers

K. J. Offor, L. Vaci, and L. S. Mihaylova (2019). "Traffic Estimation for Large Urban Road Network with High Missing Data Ratio," Sensors, vol. 19, no. 12, p. 2813, 2019. doi: 10.3390/s19122813

1.7 Thesis Outline

The current chapter discussed the motivation Section 1.1, problem formulation Section 1.3, objectives for the research Section 1.4, key contributions Section 1.5 and publications Section 1.6. The remainder of the thesis is organised as follows:

Chapter 2 presents the theoretical background of intelligent transportation systems and review of related works. Section 2.1 presents a brief overview of intelligent transportation systems and traffic estimation approaches. Section 2.2 presents traffic flow models such as microscopic flow models Section 2.2.1, macroscopic traffic flow models Section 2.2.2, and mesoscopic flow models Section 2.2.3. Section 2.3 presents traffic state estimation approaches such as neural networks NN Section 2.3.1, deep learning Section 2.3.2, Kalman filters (KF) Section 2.3.3, principal component analysis (PCA) Section 2.3.4, support vector machine/regression Section 2.3.5. Finally, Section 2.4 presents a review of related works as it relates to this work with special emphasis on Kriging and PF state-of-the-art Section 2.4.1.

Chapter 3 presents short term traffic flow prediction with particle methods in the presence of sparse data. Section 3.1 provides the contextual significance of the proposed idea. Section 3.2 discusses the traffic flow and measurement model used in this work. Traffic state interpolation and prediction using Kriging and particle filters (PF) are presented in Sections 3.3 and 3.4, respectively. Performance evaluation is presented in Section 3.5, Results and discussion are 3.6 with conclusions being drawn in Section 3.7.

Chapter 4 expands the work presented in Chapter 3. Section 4.1 introduces the concept with a brief review of existing literature. This is followed by the presentation of traffic flow and measurement model used in this work in Section 4.2. Recursive Bayesian estimation and particle filters (PF) are presented in Section 4.3. The proposed method was discussed in Section 4.3.3. Section 4.4 discusses the performance evaluation with experimental/simulation design in 4.4.1 and results and discussion presented in Section 4.4.2. Section 4.5 concludes the chapter.

Chapter 5 explores the use of multi-model Kriging methods for urban traffic estimation. Section 5.1 explores the contextual background of the chapter. Section 5.2 gives a brief overview of the related works. Section 5.3 presents the formulation of the problem of interest. Section 5.4 presents the experimental setup and performance evaluation. Finally, Section 5.5 concludes this chapter.

Chapter 6 concludes the thesis in Section 6.1 with recommendations for future work in Section 6.2

Chapter 2

Review of State-of-the-Art

This chapter presents a review of relevant literature and state-of-the-art approaches in traffic estimation and control. First, the theoretical background of intelligent transportation systems and the analysis of related works are presented. Second is a brief overview of intelligent transportation systems and traffic estimation approaches. Thirdly, traffic flow models, namely: microscopic flow models, macroscopic traffic flow models, and mesoscopic flow models. In the following Sections, traffic state estimation approaches such as Kriging, neural networks, deep learning, Kalman filters, particle filters, principal component analysis, and support vector machine/regression were presented. Finally, a review of related works as it relates to this thesis with particular emphasis on Kriging and PF state-of-the-art was discussed.

2.1 Brief Overview of Intelligent Transportation Systems

Extensive research in traffic modelling began in 1933 with Greenshields' [41] formulation of the famous traffic theory called the fundamental diagram. He studied the relationship between the velocity, v of vehicles and the average distance, x between two consecutive vehicles. This model was later extended to involve other variables such as density, ρ and flow, q. The fundamental diagram is now expressed in terms of a plot of flow - density, Figure 2.1, speed - density Figure 2.2 and speed-flow Figure 2.3. In the figures shown, ρ_c, ρ_j and ρ_m is the critical, jam and maximum density respectively, v_f and v_m is free-flow and maximum speed respectively and q_m is the maximum flow. At low traffic density, there is free flow, Figure 2.1 and 2.2 and as the the density increases, flow rate rises while speed drops until the capacity is reached (critical density). Corresponding increase in density after the critical density results to decrease in flow and speed Figure 2.3 until the flow and speed

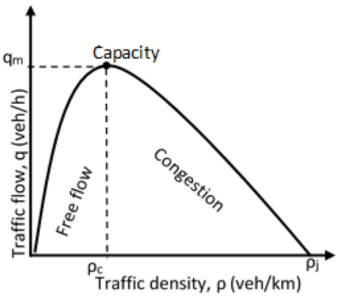

Figure 2.1 A flow - density diagram

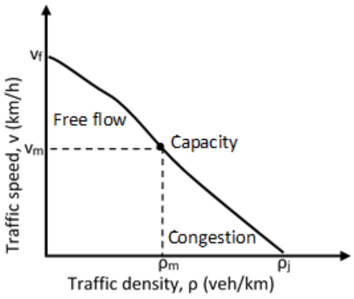

Figure 2.2 A speed - density diagram

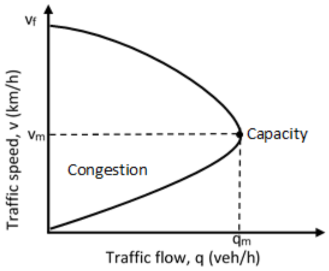

Figure 2.3 A speed - flow diagram

becomes zero at the jam density (ρ_j). This was followed by the formulation of other traffic flow models such as microscopic, macroscopic and mesoscopic in the 1950s. These traffic flow models were developed to estimate and forecast the state of traffic in a road network. This review will be carried out under the following sections, traffic flow models, traffic state estimation, and prediction methods.

2.2 Traffic Flow Models

Traffic flow model, as used in transportation engineering, is a mathematical model that describes the interactions between commuters such as pedestrians, cyclists, motorcyclists, motorists together with their vehicles and the transportation infrastructure such as the road network, traffic signs and control devices. Traffic modelling helps to understand the evolution of traffic and then use the formulation to estimate and forecast traffic state for management of traffic. In [42, 11, 12], an overview of different modelling approaches was given. These include the microscopic, macroscopic and mesoscopic. Some work combines these models to form a hybrid model.

2.2.1 Microscopic Traffic Models

In microscopic modelling, the behaviour of individual vehicles with other vehicles and the road network is modelled separately. This modelling in effect mirrors the dynamic behaviour of each successive vehicle called car-following models [13, 11, 12, 14]. The vehicle in front is called the leader, while the one behind is called a follower. Three parameters: position of the vehicle, x, its velocity, $v = dx/dt$ and its acceleration $a = dv/dt$ are used to model the behaviour of each individual vehicle. Microscopic traffic models are classified into three broad categories, namely car-following, cellular-automata and lane-change models.

The car-following model is based on the idea that a vehicle will maintain a minimum spatial and temporal separation between it and vehicles it is following. This is also divided into three different models. The first of this model was proposed in [13] and assumes that a vehicle will change its speed to maintain a minimum safe distance to the ones preceding it. Gipps [14] improved the model proposed in [13] by introducing acceleration and deceleration. The acceleration models the desire of the vehicle to maintain a maximum speed limited by the legal speed limit and vehicle's capability while the deceleration models the willingness of the driver to slow down to maintain a safe distance with the vehicles ahead of it. Newell (2002) revised his model from 1961 (Newell, 1961) by assuming that the follower driver chooses a velocity based on time spacing and acceleration based on the speed difference, which is proportional to his deviation from an equilibrium curve with relaxation time. A comprehensive review of car-following models is presented in [43].

The cellular automata (CA) model was proposed by [44] and was later adapted for real application by [45–48] to address these limitations. The CA model vehicles evolution either deterministically or stochastically and could be applied to single lanes as well as multi-lane networks. A set of rules govern the evolution and interaction of vehicles. One of the rules, called randomisation, models three different human driving behaviours namely retarded (noisy) acceleration, overreactions at braking, and fluctuations at maximum speed [47].

In cellular automata, the street is divided into cells i of length Δx, the time t is divided into j intervals of 1s duration. At any time Δt, each cell i is either empty or occupied by one vehicle with speed v_j given by Equation (2.1),

$$v_j = \hat{\alpha}_j \frac{\Delta x}{\Delta t}, \tag{2.1}$$

where: $\hat{\alpha}_j \in \{0, 1, 2, ..., \hat{v}_{max}\}$ is a quantity which takes a discrete value between zero and the maximum velocity, $\hat{v}_{max}$.

The space and time are in discrete form such that v positions of the cars in all cells are updated in parallel using rules (i) to (iv) [47, 49]:

1. Motion: Advance each vehicle by $\hat{v}_j$ cells and check if it has reached its maximum velocity $\hat{v}_{max}$

2. Acceleration: If $v < v_{max}$ and there is enough head way, increase the velocity to $v = v + 1$, else
3. Deceleration: If the vehicle is too fast, slow down
4. Randomization: If above steps yields $v > 0$, reduce the velocity by one with probability p.
5. Particle propagation: Move each particle v sites ahead.

Lane changing models involve the decision of a driver to change lane either mandatory or discretionary without affecting vehicles in the destination lane. A lane change becomes compulsory, especially when there is a lane closure and discretionary is a driver does so to improve perceived driving/road condition [50]. An extensive review of lane changing models is presented in [51, 52]. Two different models, lane selection and gap models, were identified. The first, lane selection model is based on two sets of rules (mandatory and discretionary part). The driver selects any lane from a set of available lanes based on a set of rules deterministically. Among the notable works in this deterministic rule-based process are [53–55]. To address the stochastic nature of traffic, [56] introduced a random variable to capture the trade-off between various factors that affects lane choices at any point in time. This was further improved in [57] where a three-level selection process was used to capture the mandatory and optional rules. The second step, the lane change execution, is modelled using gap acceptance models which describes how a driver decides to execute the lane change to ensure the safety of road users. A state-of-the-art review of lane changing models is in [58].

2.2.2 Macroscopic Traffic Models

Macroscopic traffic models use aggregated values (average speed, and average density) of traffic flow over a given space to describe the behaviour. The first macroscopic model was proposed independently in 1955 by Lighthill and Whitham [59], and in 1956 by Richards [60]. Hence, the model is named after the three as the Lighthill-Whitham-Richards (LWR) model. The LWR is based on first-order kinematic wave theory and describes the dynamics of traffic flow using partial differential equations. This is known as the conservation of vehicles equation (2.2). The flow is expressed to depend on the occupancy of the sending segment and not the receiving segment. Newell [61] showed that using the above approach to compute flow assumes that traffic cannot flow to the receiving segment that is free if the sending segment is congested. Thus, the simulation does not lead to convergence and appropriate solution. An attempt to solve these problems and ensure that occupancy of segments remained within zero and maximum value were proposed by [62, 63] by introducing constraints on the flow.

A higher-order macroscopic flow model was proposed by Payne in 1971 to address the problem of infinite deceleration and acceleration caused by the wrong assumption of instant speed change after a change in density. As this approach did not lead to convergence, the Cell Transmission Model (CTM) was proposed by Daganzo [16] to address the non-convergence. Another macroscopic model called METANET was proposed by Messmer and Papageorgiou [15]. Recently in [64], a variable-length CTM is introduced with three lumped state variables. The road network under consideration is subdivided into two cells with variable length, the densities of these two cells are then lumped, and the third state variable is used to account for the position of the congestion wavefront.

$$\frac{\delta \rho}{\delta t} + \frac{\delta q}{\delta s} = 0, \tag{2.2}$$

where ρ is density (veh/km), q is flow (veh/hr), s is distance (in km) and t is time (in hrs)

One major drawback of the macroscopic flow model is that it cannot adequately model lane behaviour, headway and speed choice at random cross-sections. However, for most practical purposes such as traffic management, road pricing and changes in infrastructure, the macroscopic model is enough to produce acceptable estimation and prediction. Among the most used macroscopic models are the cell transmission model (CTM), stochastic compositional model (SCM).

The cell transmission model (CTM), based on the macroscopic traffic flow model was proposed by Daganzo [16, 65]. In CTM, the road is divided into segments called cells. Each cell is then modelled by three states,

1. $N_i(t)$, the maximum number of vehicles admissible in cell i at time t,
2. $Q_i(t)$, the maximum number of vehicles that can flow into cell i at time t, and
3. $n_i(t)$ the actual number of vehicles in cell i at time t.

The first cell is called the inflow while the last cell is the outflow. Traffic evolution $y_i(t)$ from cell $i-1$ to i within a discrete time step Δt is the minimum of three parameters, $n_{i-1}(t)$, the number of vehicles in previous cell $i-1$ at time t, $Q_i(t)$ the maximum number of vehicles that can flow into cell i at time t, and the available empty space in cell i at time t denoted by $N_i(t) - n_i(t)$. The CTM has been applied successfully in literature for traffic estimation [38].

Boel and Mihaylova [66] extended it to factor in the stochastic nature of traffic by modelling the sending and receiving functions as a random process and dynamically capturing the average velocity in each cell. The (SCM) was successfully combined with particle filter by [2] to improve on-line estimation.

2.2.3 Mesoscopic Traffic Models

These models use probability distribution functions to describe the behaviour of vehicles in aggregated terms similar to macroscopic models but also define the behavioural rule for individual vehicles as in microscopic. Examples include the gas kinetic model [11, 67–69]. The mesoscopic model is most suited for applications that requires capturing some details of the interaction between groups of vehicles and not individual vehicles as in microscopic. Notable mesoscopic models which have been adopted by the Federal Highway Administration (FHWA) are the DynaMIT [70] and DYNASMART [71]. A recent work by Jamshidnejad [72] modelled vehicle emission and flow using the mesoscopic model and reported a higher performance compared to using either the macroscopic or microscopic model.

2.3 Traffic State Estimation Approaches

The last three decades have witnessed a growing demand for short term traffic state forecasting. This is of interest to either traffic management officers or road users to enable them to plan their journey and for safety purposes. Research in this field which dates to 1979 [73] has focussed in different areas such as travel time, speed, traffic density/congestion, traffic flow or a combination of these. The goal of traffic state estimation is to use different approaches in determining the current state of traffic conditions from noisy measurements. In contrast, traffic state prediction seeks to use a combination of techniques to forecast the future traffic condition using either traffic models, historical data or combination of both. Traffic state estimation and prediction could be carried out using either model-based approach, data-driven approach or a combination of these two. The first makes use of the traffic flow models presented in section 2.2, the second approach uses historical data to estimate the current state of traffic and predict a future state. At the same time, the third combines the two approaches. A comprehensive review of research in these areas up to 2014 was done by [74, 28, 75]. Various methods/techniques have been used to achieve these such as linear regression and its variants. The Box-Jenkins technique [73], autoregressive integrated moving average [76], and dynamic linear models [77]. The Bayesian inference based methods: Kalman filter and its variants [78–80], particle filter [2, 81]. And the Dimensionality reduction / coordinate transformation methods: principal component analysis [82], support vector machine [83], support vector regression [84].

Travel time prediction using instantaneous approach was investigated by [85, 86]. This assumes a stationary traffic condition for an indefinite period. In [87], a state-space neural network (SSNN) was used to train data offline to predict travel time in the presence of missing data. Because the training was performed offline, it was affected by changes in traffic conditions with substantial training time, and they proposed an online learning system to address these [88]. Ladino et al. [35] used clustered time series data and the Kalman filtering algorithm to predict travel time. They used k-means for the clustering and dictionary learning approach to correct the missing data problem. They reported a prediction accuracy of more than 35% in 90% of the cases. The research did not consider the effect of constraints like incidents, weather and road works. In [89] historical data and gradient boosting regression tree methods were used to predict travel time. This section will present a review of some important research in these areas and highlight their strengths and weaknesses.

2.3.1 Neural Networks (NN)

The neural network uses the principle of information transmission in human neurons. It is composed of an input neuron and an output neuron connected via weighted hidden neurons. As the data is transferred from one neuron to the other, they are scaled according to the weight assigned to the neurons [90, 91]. Neural networks have been used extensively in traffic estimation. Multivariate and univariate traffic data estimation using NN with genetic algorithm optimisation was investigated by [92]. Zheng et al. [93] combined the backpropagation algorithm and the radial basis function models in a Bayesian way to predict traffic flow and reported an improved performance over a single model. Instead of treating the NN as a “black box”, [94, 87] proposed the notion of state-space NN (SSNN) where the model is based on the traffic-related data of recurrent neural network. The problem of missing data was solved in [87] using simple imputation algorithm. In [95], lane changing behaviour in a dual carriageway was modelled using a NN.

Another extension of NN called ensemble neural network (ENN) combines many NN models to achieve better prediction results. Three standard ensemble methods are the Basic Ensemble Method (BEM) which uses the arithmetic average of the different models. The Generalised Ensemble Method (GEM) which uses a weighted sum of the various NN models and the Bagging (Bootstrap Aggregating) which replaces a subset of the training data with a random combination of the training data itself. In [96], the performance of Radial Basis Function (RBF) NN was compared with bagging RBF ENN for short term traffic flow prediction and concluded that the ensemble method outperforms the single model. A more recent study proposed in [97] uses least squares support vector regression (LSSVR) with the Gaussian kernel function and improved harmony search algorithm for short-term traffic flow prediction. The root mean squared error (RMSE) shows that the method outperforms five other methods compared in the study.

2.3.2 Deep Learning

Deep learning is a branch of machine learning based on the idea that observed data are related to some multiple underlying layers of abstraction. It models a system using many layers to connect the intrinsic relationships to provide a better representation of the interactions. The levels of abstraction have been modelled in different methods such as Convolutional Neural Networks (CNN) [98], Deep Neural Networks (DNN) [99] and Deep Belief Networks (DBN) [100]. They have been applied mostly in speech recognition, computer vision [101] and Natural Language Processing (NLP) [102]. The authors in [103] proposed a deep learning approach to estimate traffic flow. They used a stacked auto-encoder model to extract the correlated features of the data, which is subsequently trained using a greedy layer-wise unsupervised learning algorithm. They reported that the method outperformed existing similar methods. Deep learning with an l_1 regularisation fitting and $tanh$ layers sequence was proposed by [99] for short-term prediction of traffic flow. They reported improved performance and observed that more recent observations (less than 40 minutes) have a higher contributing effect to the model than older observations.

2.3.3 Kalman Filters and Variants

The Kalman filter (KF) is the best estimator of linear systems. It was proposed by Rudolf Kalman [104]. It consists of two basic steps, the prediction step and measurement update step. It works with the basic assumption that the model is linear with a zero-mean Gaussian noise. When the model is linear, it gives an optimal solution, and when the model is non-linear, it gives a sub-optimal solution. The extended Kalman filter was proposed to solve the estimation problem for non-linear models by linearising it. The linear approximation introduces some errors in the estimation and propagates it through the iteration process leading to less accurate solutions. Julier and Uhlmann [105, 106] proposed the unscented Kalman filter to address this problem. In Unscented Kalman Filter (UKF), some sample locations called sigma points are chosen carefully to capture the true mean and covariance of the Gaussian random process.

2.3.4 Principal Component Analysis

PCA tries to solve the estimation problem by coordinate transformation (reducing) the dimension of the input vector to a lower-dimensional output axis. The transformed dimensions are called the principal components. The PCA is sensitive to outliers. The basic idea is to reduce the dimension of the dataset by the removal of redundant and correlated data that gave rise to the higher dimensionality input by orthogonal projections. This is achieved by computing the variances of the components and assigning the one with the highest variance as the first principal component, the component with the next highest variance and orthogonal to the preceding one becomes the second principal component and so or until all the components are accounted for. One major issue with PCA is that principal components are skewed in the presence of large anomalies in the data [107, 108].

Tsekeris, [109] proposed the use of PCA for analysis of spatio-temporal traffic data. They reported that using the approach yielded a considerable reduction in complexity to a small set of eigen flows, thereby leading to the identification of error-prone links and the spatio-temporal stationary variations in traffic flow with abnormal events. Foerster [82] combined PCA and hierarchical cluster analysis (HCA) to estimate the correlation and redundancy of traffic data and concluded that the 11.5% of the sensors accounted for more than 50% of useful data. One drawback of the research is that there is no validation to show that the sensors will give optimal results. In the work of [110] PCA was used to unmask the hidden structures in a traffic dataset and applied compressive sensing to estimate missing data by exploiting the identified hidden trend. They were able to achieve 20% estimation error with 80% of the data missing (the Pareto principle of 80 / 20 rule).

An extension of the PCA called robust PCA (RPCA) was investigated in recent times by [111–113]. The authors of [113] used RPCA to separate abnormal traffic flow pattern caused by faulty sensors and incidents, and then detect the causes of the anomaly. RPCA was employed in video and image processing by [112] where they were able to detect objects in a cluttered background.

2.3.5 Support Vector Machine/Regression

Support vector machines are methods used in classification and regression analysis of data for prediction purposes. Vapnik proposed the idea in 1962, but the first application to machine learning was implemented in 1992 by Boser, Guyon and Vapnik [114]. Cortes and Vapnik proposed the current form of Support Vector Machine (SVM) in 1995 [115]. The basic idea of SVM is the transformation of input dataset into a higher or infinite-dimensional feature-space and then separating the features into hyperplanes that are maximised to optimise the solution. SVM finds its use in sparse data, computer vision [116], over-fitting and mapping non-linear data into a linear higher dimensional feature-space by the use of kernels and then solving the optimisation problem. Polynomial, linear and radial basis functions are the frequently used kernel functions. A general tutorial of the use of SVM was given by [117–119].

In the field of traffic engineering, several pieces of research have been conducted using SVM. Bin et al. [83] used SVM in predicting travel time by separating the dataset into four hyperplanes and then using historical travel time information of preceding vehicles in the next segment and the travel time of the vehicle in question in the current segment to predict the travel time of the vehicle in the following segment of a road network. They used the RBF kernel and concluded that SVM outperforms Artificial Neural Network (ANN) by 5% to 7% depending on the pattern used. Another variant of SVM used in regression analysis is the Support Vector Regression (SVR). Some notable works in SVR include [120, 121, 84, 122, 123]. The traditional SVR uses inequality constraints to evaluate the loss function for the maximisation of the hyperplane margins. Another variant of SVR that uses the least-squares approach for the evaluation of the loss function called least squares SVM (LSSVM) was used by [124]. They applied fruit fly optimisation algorithm (FOA)in LSSVM to optimise the loss function. They concluded that it outperforms three other variants, namely the radial basis function (RBF) neural network, single LSSVM model, and LSSVM combined with particle swarm optimisation (PSO) algorithm. They also showed that the LSSVM with FOA converges quicker than PSO.

2.4 Review of Related Works

As the core of this theses is based on Kriging and Particle filter; the remaining part of this chapter is devoted to a detailed review of related works using these techniques. The classic method of computing variance in Kriging is biased as it underestimates the true variance, [125] proposes using parametric bootstrapping for the computation of Kriging variance.

2.4.1 Kriging and State-of-the-Art Review

Kriging is a point-based estimation method that relies on the basic idea that spatial observations are correlated and decreases as the distance increases. It uses a weighted sum of variations (called semi-variogram) between input points with measurements to compute the values of output locations without measurements. The method which was originally developed for gold mining in South Africa by Krige [126] and formalised by Matheron [127] has become the choice algorithm in spatial interpolation. Three basic types of Kriging exist based on the trend of $\mu(\mathbf{r})$ term in the equation, (2.3). If it is constant within a given range, Ordinary Kriging [21] is used. A Simple Kriging [21] is used if the trend is not constant but known while Universal Kriging [21] is used if the trend is not known. Kriging differs from the Moving Average (MA) in the way weighting factors are calculated. MA computes the weight as a function of the distance between measurement points and the point where estimation is required, whereas Kriging uses semi-variograms between the points to compute the weights. Different types of functions are used for the computation of semi-variograms, namely Exponential, Gaussian, Circular, and Spherical model [21].

To use Kriging in spatial interpolation, the area of interest is divided into different windows and bins. Each window contains points with known observation locations and the points where estimation is needed. The measurement, $z(\mathbf{r})$ with position vector $(\mathbf{r})$ is decomposed into two parts called *residuals* $\varepsilon(\mathbf{r})$ and *drift* $\mu(\mathbf{r})$, equation (2.3).

$$z(\mathbf{r}) = \mu(\mathbf{r}) + \varepsilon(\mathbf{r}), \tag{2.3}$$

where $\mu(\mathbf{r})$ is the average value of measurements, and $\varepsilon(\mathbf{r})$ is zero-mean valued random error quantity. A detailed description of the Kriging procedure is given in Algorithm 2, Section 3.3.6.

Kriging has been applied in the field of traffic engineering for state estimation [128, 19, 21, 23] and crash detection [129]. Braxmeier et al. [128, 19] employed Kriging with moving neighbourhood to estimate the spatial location of traffic using data from GPS equipped vehicles. Manepalli and Bham [129] compared Kriging and empirical Bayes in assessing road crashes and reported that Kriging performed better when the prediction term is less than three years. Wang and Kockelman [21] applied Ordinary Kriging with exponential model function to estimate annual average daily traffic and reported an error of 31%. The limitation of previous traffic estimation/prediction using Kriging algorithms has focused on only spatial interpolation. Kriging method could be extended/modified to handle spatio-temporal prediction. For instance, [130] employed what he called modified Taylor Kriging to predict wind speed and reported 18.6% increase in performance compared to ARIMA. Other authors [131–134] have applied a modified form of Kriging to achieve spatio-temporal forecasting. This research will investigate the use of Kriging in performing traffic prediction forward in time augmented with spatial estimation, which will be used as a form of virtual sensor to replace faulty sensors or missing data.

The Kriging algorithm [126] is a point-based estimation method which relies on exploiting the spatial correlation, of the data points. The Kriging algorithm attempts to interpolate the values at an unobserved location using statistics of the spatial variation between pairs of observed locations in a given region. In [135] regression, Kriging was used to estimate radon concentration in an area with limited measurements. The author computed the reliability of the results by using 90% confidence interval. The result of the RMSE indicated an average error of 25%.

In [136] effects of traffic volume and its composition on speed and passenger car unit (PCU) factors for individual types of a vehicle under mixed-traffic conditions were examined using Kriging. They first predicted the speed of vehicles using regression Kriging and then used the predicted speed to compute the PCU under different traffic volume conditions, except the congested state. This was attributed to a lack of data. In [137], the performance of Kriging using the bootstrap method, conditional simulation and the classic method was compared. The report showed that there is no significant difference in performance to warrant using a bootstrap method or conditional simulation.

In [138] a semi-parametric bootstrap for spatially correlated functional data is proposed. The method allows for the evaluation of the uncertainty of a predicted curve, that ensures the maintenance of the spatial dependence structure in the bootstrap samples. Two main issues were specifically addressed in the extension. The first is how to specify and estimate the spatial dependence structure and, the second is how to order the function curves to obtain functional quantiles. Tapoglou [139] combined ANN and Kriging for temporal and spatial prediction of groundwater levels, respectively. In [140], the problem of growing complexity in computation with increasing datasets is proposed by using Markov-Cube Kriging (MCK). MCK can address spatio-temporal missing data, mismatch and misalignment in large spatio-temporal datasets. MCK utilises Gaussian Markov random field (GMRF) priors to model multi-level interactions across space and time, uses a Bayesian hierarchical model to incorporate covariates, time-space heterogeneity, and physically meaningful prior information into the model.

Kriging interpolation is usually achieved by choosing a drift function which is then fitted into a covariance model. The covariance function models the spatial correlation of the data points and could be either Matern, rational quadratic, exponential, spherical, or power [141] with some parameters to be fitted. The computation of the covariance matrix scales to order n^3. This becomes computationally intractable for a large non-stationary dataset as the different covariance structures must be inferred [142]. Three different approaches have been proposed in the literature to address this problem.

The first is covariance tapering [143]. This is achieved by ignoring (setting them equal to zero) the uncorrelated distant covariance functions resulting in a sparse matrix. The likelihoods are then computed using sparse matrix algorithms. Tapering only the model covariance matrix results to estimation bias when the taper range is small compared to the correlation range. When the model and sample covariance matrix is tapered, the bias is eliminated with a slight increase in the variance. The work assumes a stationary, isotropic system with zero mean and fails to converge if the assumptions are not met. The second method represents the covariance function by a low-rank matrix or a limited number of basis functions [144–148]. A nugget effect is added to account for local unstructured noise.

The third approach is windowing and binning, which divides the entire region into subregions with each bin treated separately. Some limitations of binning include: the mean computation does not account for redundancy among nearby observations. The mean could be computed from a different number of measurements across grid-cells, and the grid cells that may not contain observations in a given time window are not accounted for. Binning also fails to capture the correlation of distant locations resulting in a global sub-optimum solution. Tadic [149] proposed to solve this problem with the use of local covariance structures by an arbitrary selection of the sampling function, limiting the radius around estimation locations and adjusting the number of sampled points to a fraction of available measurement. Braxmeier [128] takes into consideration the inhomogeneous and anisotropic nature of time-series data in estimating road traffic parameters. The method considered only rush hour data set of only 30 minutes duration.

The classical Kriging method uses a generative model of the covariance function to estimate the weights. This approach is not dependent on the data or site but the joint probability of the locations. Bayesian Local Kriging (BLK) as proposed by [150] uses a discriminative model of the covariance function which are conditioned on the data to estimate the weights. This approach works well for both stationary and non-stationary system or observations. Their model addressed two assumptions usually made in Kriging: stationarity and homoscedasticity. The first problem was solved by using a discriminative covariance function conditioned on the data while the former was addressed by using a set of L local covariance functions. They considered a case where the regression part of the function is the same for all L local models.

Addressing the stationarity assumption, [145] proposed a fully Bayesian model based on non-stationary Matérn covariance function with random components and parameters. The parameters were assumed to vary smoothly across space as linear combinations of spatial basis functions. The problem of the large spatial area is addressed by combining a low-rank component with a tapered remainder component. The first component allows for flexible modelling of medium-to-long-range dependence via a set of spatial basis functions. In contrast, the second provides for modelling of local relationship using a compactly supported covariance function. Similar to the above, [151] addressed non-stationarity by modelling and estimating two parameters: range, which determines the variation distance and directionality, which determines the direction of variation in space by using Mahalanobis distance.

In [143], a method called tapering is used for large scale datasets to improve computational efficiency. This is achieved by ignoring (setting them equal to zero) the uncorrelated distant covariance functions resulting in a sparse matrix. The likelihoods are then computed using sparse matrix algorithms. Tapering only the model covariance matrix results to estimation bias when the taper range is small compared to the correlation range. When the model and sample covariance matrix is tapered, the bias is eliminated with a slight increase in the variance. The work assumes a stationary, isotropic system with zero mean and fails to converge if the assumptions are not met. In [152], a semi-parametric Gaussian process with additive components that could to model large or massive spatial datasets is proposed. The covariance structure of the GP consists of two parts. The first assumes a specific flexible covariance parameter that can achieve dimension reduction. The second covariance structure is parametric and simultaneously induces sparsity.

In [147] and [153, 154], a multi-resolution approximation using many linear combination basis functions was proposed to address computational complexity of large datasets. The novelty lies in the use of multiple basis functions computed at lower resolutions closer to observation locations and then combining them to capture the different covariance functions with varying properties. The division is achieved by dividing the spatial domain recursively into small regions and smaller sub-regions until the fine-scale dependencies are captured. Nychka [147] used radial basis functions (RBF) and a particular type of Gaussian Markov random field (GMRF) called spatial autoregressive (SAR) model to model the spatial correlation among the coefficients. At the same time, Katzfuss [154] automatically determines the appropriate basis/covariance function. Whereas in [153], it was assumed that the sub-domains are independent, [154] assumed depended sub-domains and performed full-scale approximation. As the computations are done locally in parallel, it is possible to fuse multi-sensor data sources, in which case, the different covariance functions are used for each data source. Although both mentioned that the approach could be extended to non-stationary functions, it was not implemented, nor was there a derivation for such.

The Kriging weight is only a function of the distance between the points. It does not incorporate the heteroscedastic/stochastic nature of the data. In [21, 155], it was shown that local accuracy of estimation using ordinary Kriging weights is affected by data values. The use of interpolation variance was used by [156] to correct the smoothing effect of the Ordinary Kriging variance. A modified-nugget effect was proposed by [155] to account for location-dependent non-constant variances.

The ordinary Kriging variance is given by

$$\sigma_{e_u}^2 = \gamma_u - \mathbf{w}^T \mathbf{b} - \lambda. \tag{2.4}$$

The heteroscedastic variance is dependent on the data values and is given by

$$s_u^2 = \sum_{i=1}^{n} \lambda_i (z_i - z_u). \tag{2.5}$$

For the variance to be positive, all the Kriging weights must be positive. These are corrected to be positive using one of three approaches proposed by [157]

First approach removes all negative weights by setting them to zero. The new weights then become:

$$\lambda_i^* = \frac{\lambda_i}{\sum_{j=1}^{n} \lambda_j}. \tag{2.6}$$

The second approach removes the sample with the largest negative weight:

$$\lambda_i^* = \frac{\lambda_i + c}{\sum_{j=1}^{n} \lambda_j + c}, \tag{2.7}$$

where $c = -min(\lambda_i, i = 1, ...n)$ for $min(\lambda_i > 0)$

Third approach is implemented in two steps
The first step is the same as in first approach. The second step is implemented using the

conditional statement: if $\lambda_i > 0$ and $Cov(u - u_i) < \bar{c}$ and $\lambda_i < \bar{\lambda}$, then set $\lambda_i = 0$ and the new set of weights become (2.6), where $Cov(.)$ is the covariance function, $\bar{\lambda} = \frac{1}{n^-}\sum_{j=1}^{n^-}|\lambda_j|$ and $\bar{c} = \frac{1}{n^-}\sum_{j=1}^{n^-} Cov(u - u_i)$, and n^- is number of locations with negative weights.

The new Kriging interpolated values now become:

$$Z_u^* = \sum_{i=1}^{n} \lambda_i^* Z_i. \tag{2.8}$$

In [158, 134] Semantic Kriging was employed to predict spatial and temporal missing attributes in GIS. In [159], the performance of semantic Kriging was compared with five other spatial estimation methods, namely simple co-Kriging, inverse distance weighting, multilayer perceptron, Bayesian network, Nearest neighbours. The results showed that the Semantic Kriging outperforms the other methods. The basic idea of Semantic Kriging is the inclusion of the semantics knowledge of the surrounding area to capture their correlations better. For road traffic prediction, there is the possibility that different road features such as intersections, shopping centres, bus stops, nearby parks, etc. affect traffic flow. Thus, computing the Kriging weights as dependent only on the distance between interpolating and interpolation points could potentially limit the accuracy. The contribution of the surrounding features could be modelled deterministically or learned from data. The measured data could be used to compute the correlation of the points and use this to capture the features of the locations.

The classic method of computing variance in Kriging is biased as it underestimates the true variance, [125] proposes using parametric bootstrapping for the computation of Kriging variance. In [137], the performance of Kriging using the bootstrap method, conditional simulation and the classic method was compared. The report showed that there is no significant difference in performance to warrant using a bootstrap method or conditional simulation. The use of Euclidean distance in computing separation between locations fails to accurately describe the spatial range in a road network. Zou [23] introduced road network distance to describe the spatial distance between road links for solving the problem of the invalid spatial covariance function in Kriging caused by the non-Euclidean distance metric. They called it approximate road network distance (ARND), based on the isometric embedding theory.

2.4.2 Particle Filter in Traffic Estimation: State-of-the-Art Review

The particle filter estimates the state of traffic by taking a sufficient number of random samples from the pdf with assigned weight to each particle. When a new measurement becomes available, it is used to compute what is known as likelihoods, and a normalised form of the weights is calculated. The new state of the system is then updated with the calculated weights. The importance weights tend to degenerate as the number of iterations increase. This problem is solved by resampling the weights. This involves replacing particles with low weights with a replica of those with high weights. The particle filter has been applied extensively in traffic state estimation and prediction. Mihaylova et al. [2] proposed estimating freeway traffic using the PF and extended form of CTM called SCM and compared the performance with UKF. The result indicated that the PF is of better quality in terms of RMSE with synthetic and real data. The particle filter was also employed in a distributed manner in what is called Parallelized Particle Filter (PPF) and Gaussian Sum Particle Filter (GSPF) by [160] to speed up estimation across a large road network.

PFs have been used with both microscopic, hybrid and macroscopic traffic models for traffic state estimation and prediction in real-time scenarios. If platoons of vehicles travelling at similar velocities are instead considered this model becomes a hybrid model rather than a microscopic model [2]. A key component of intelligent driver assistance systems in vehicles will be the ability to track the changes in location/motion of both the vehicle and those in the surrounding area. The work in [161] and [162] consider this issue. In both PFs are used to estimate the current states of the vehicles and also to make long term predictions (in this scenario, meaning 1-2 seconds). In [162], the high dimensionality problem is also addressed using principal component analysis.

On the other hand, [163] considers the problem of estimating the traffic states on arterial roads based on sparse probe vehicle data. Coupled Hidden Markov models (probabilistic graphical model) are used to model the evolution of the traffic states. A PF is used both in estimating the current states and predicting future states (over the short term) based on the data received up to the present time.

PFs can also be used for traffic state/model parameter estimation, with the estimates then being used in the models detailed in the next chapter. For example, in [2] PFs are used with the compositional model (CM), which is an example of a macroscopic traffic model. The state vector of interest is the number of vehicles within a segment of road, along with their corresponding average speeds. This is achieved with the help of the measurements of vehicle numbers and average speeds crossing the segment boundaries. Synthetic and real data (from Ghent in Belgium) showed that the PF has a more accurate tracking performance than a UKF. As a result of this, it is suggested that PFs are an appropriate choice for online traffic control strategies. PFs have also been implemented using both first order, [164, 165], second-order macroscopic traffic models [166], Hybrid approaches to modelling [167] and dynamic Bayesian networks [168].

PF-based methods can also be extended to consider multiple models to both estimate the traffic state and detect the traffic incidents [169]. Here an augmented state vector is considered. This contains information about the traffic state (continuous model) and model variable (discrete model) that gives details about the number of lanes open at a given time. An incident can then be detected if there is a significant disagreement between the predicted states and the measurements made (more accurate than detecting incidents in a dedicated algorithm). The dynamics of the highway can then be altered to account for this, allowing an improved performance compared to a model purely calibrated for ideal traffic conditions. Simulation results have indicated a right accuracy level is achievable. However, the authors suggest parallelisation may be required for large scale road networks. An efficient version of this algorithm has also been recently proposed in [170].

In [171], it is argued that it is difficult to model the transition between states exactly. Therefore, instead of using a state transition model $(i.e.p(x_k|x_{k-1}))$, historical data sequences are used. In other words, the change in the state of a particle is determined by comparing the state of the particle to those in the historical sequences. The corresponding change in state is then made. Also, the resampling used to avoid degeneracy has been altered. Instead of replicating the particles with high weights, the particles with low weights are instead replaced with particles from similar sequences of historical data (e.g. same time of day for a set day). Analysis from a test site between Richmond and Virginia Beach has shown improved results as compared to two variations of the KF and the k nearest neighbours' method.

In [160] Gaussian sum PF (GSPF) was implemented for traffic estimation in a parallelised form. In this form, the mean and covariance are transmitted between subnetworks, like the second parallelised PF method described above. The results presented show that the GSPF (or parallelised from) is more accurate than comparable PFs (or parallelised PFs). As for PFs, computation savings are made by using the parallelised form of the GSPF.

2.5 Performance Measures

This section presents a common performance measure used in all the chapters of the report. The root mean squared error (RMSE) (2.9) and normalised RMSE (2.10) were used to compare the proposed methods in this research with some existing methods and ground truth datasets.

$$RMSE = \left[\frac{1}{N_t} \sum_{i=1}^{N_t} [z_i - \hat{z}_i]^2 \right]^{1/2}, \tag{2.9}$$

$$NRMSE = \frac{RMSE}{z_{max} - z_{min}}, \tag{2.10}$$

where, z_i is the ground truth or actual measurement, z_i is the predicted value, z_{max} is the maximum value of the measurement, z_{min} is the minimum value and N_t is the number of test dataset.

2.6 Summary

The formulation of the traffic theory called fundamental diagram by Greenshields led to extensive research in traffic modelling in 1933. Greenshields was concerned with the relationship between the velocity of vehicles and the average distance between two consecutive vehicles. The model was later taken further to involve other variables like density and flow. In the 1950s, the formulation of other traffic flow models like microscopic, macroscopic, and mesoscopic started to emerge. Estimation and prediction of the state of traffic in a road network were their primary objective.

The microscopic model deals with the behaviour of individual vehicles in relations to the other vehicles and the road network. In this type of model, the variables are modelled separately. The vehicle in front is called the leader with the one in the rear is called a follower. There are three parameters used in this model type. They include the position, velocity and the acceleration of the vehicle. The microscopic model can be further classified into car-following, cellular automata, and lane-change models.

The car-following model centres on the idea that a vehicle will maintain a minimum safe distance between it and the one it is following. The car-following model further divides into three, the first proposing that a vehicle will change its speed to maintain a minimum safe distance to the ones in front of it. This model was improved upon by Gipps when introduces acceleration and deceleration. While the former suggests that a vehicle will maintain a maximum speed limited by the legal speed limit and vehicle's capability, the latter indicates that the vehicle will slow down to maintain a safe distance with it and those in front of it. Newell gives a revised approach that the follower-driver chooses a velocity based on time spacing and acceleration based on the speed difference, which corresponds to his deviation from an equilibrium curve with relaxation time.

The cellular automata model was adopted to address the limitations of the car-following. It can be applied to both single and multi-lane road network. It observes some set of rules in the evolution and interactions of vehicle, one of which is referred to as randomisation. These rule models three different human driving behaviours, namely, retarded (noisy) acceleration, overreactions at braking, and fluctuations at maximum speed. In this type of model, the street is divided into cells and the time into intervals of duration. The lane-changing model deals with the decision of the driver to switch lane either mandatory or discretionary without affection vehicles in the target lane. By mandatory lane-changing, it is meant when there is the closure of a lane while discretionary is for the driver to improve perceived driving or road condition.

Named after its three independent proponents, Lighthill-Whitham-Richards (LWR) in 1955 and 1956, the macroscopic traffic models make use of aggregated values like average speed and density of traffic flow over a given space to determine traffic behaviour. Based on first-order kinematic wave theory, the LWR uses partial differential equations to describe traffic flow such that it is expressed to depend on the occupancy of the sending segment and not the receiving segment. Newell had a different view that traffic cannot flow to the receiving segment that is free if the sending segment is congested. An attempt to solving this problem will, therefore, lead to constraints on the traffic flow.

Payne proposed a higher-order macroscopic traffic flow model in 1971, an approach which still did not lead to convergence, to address the issue of infinite acceleration and deceleration occasioned by the wrong assumption of instant change in speed after a change in density. This further led to the proposition of the Cell Transmission Model by Daganzo in a bid to address the non-convergence issue. Messmer and Papageorgiou also proposed another macroscopic traffic flow model call METANET.

Failure to effectively model the lane behaviour, headway, and choice of speed at random cross-sections is a significant setback of the macroscopic traffic flow models. However, it is enough to produce acceptable estimate and prediction for most practical purposes like traffic management, road pricing, and changes in infrastructure. The commonly used macroscopic traffic flow models are the Cell Transmission Model (CTM) and the Stochastic Compositional Model (SCM).

The mesoscopic traffic models use indefinite distribution functions to determine vehicular behaviour in gross terms like the macroscopic models and define the behavioural rule for individual vehicles as in microscopic traffic models. The mesoscopic traffic models are most suitable for capturing interactions among a group of vehicles. An example of this model is the gas kinetic model, and notable ones that have been applied by the Federal Highway Administration (FHWA) are the DynaMIT and DYNASMART.

Traffic state estimate and prediction use three approaches: the first uses traffic flow models, the second uses historical data to estimate the current state of traffic and forecast future state, and the third combines both approaches. Research up to 2014 shows various techniques/methods that have been applied to achieve these. They are linear regression and its variants: Box-Jenkins technique, autoregressive integrated moving average and dynamic linear models; Bayesian inference based methods: Kalman filter and its variants, particle filter; Dimensionality reduction/coordinate transformation methods: principal component analysis, support vector machine, and support vector regression.

Travel time prediction is an important aspect that has been investigated into using a state-space neural network to train data offline to predict time travel in the presence of missing data. But because this was carried out offline, there were impacts caused by changes in traffic conditions with considerable training time. So, an online learning system was then proposed to address these. Ladino et al. were worthy of note. They used clustered time series data and Kalman filtering algorithm to predict travel time and the k-means for the clustering and dictionary learning approach to correct the missing data problem. They reported a prediction accuracy of more than 35

The neural network has been widely used in traffic estimation. With genetic optimisation, NN can be used in the estimation of multivariate or univariate traffic data. Composed of input neurons and output neurons that are linked through weighted hidden neurons, the NN make use of the transmission of information in the human neurons. The issue of missing data was solved using simple imputation algorithm. Reports further show that lane changing behaviour on a dual carriageway was modelled using the NN. Another extension of the NN is the ensemble neural networks (ENN). It combines many NN models to achieve better prediction results. Three familiar ensemble neural methods are Basic Ensemble Method (BEM), Generalized Ensemble Method (GEM), and the Bagging (Bootstrap Aggregating).

Deep learning is a branch of machine learning based on the observation of the relatedness between data and some multiple underlying layers of abstraction. It models a system using many layers to link the intrinsic relationship to give a better representation of the interaction. Proposals have been put forward to use deep learning approach for traffic flow estimation as it uses a stacked auto-encoder model to extract the correlated features of the data, which is subsequently trained using a greedy layer-wise unsupervised learning algorithm. This method has been reported to outperform similar existing ones.

The principal component analysis seeks to solve the problem of estimation by reducing the dimension of the input vector to a lower-dimensional output axis. What is referred to as the principal component is the transformed dimension. The objective of the PCA approach is to use orthogonal projection and reduce the dimension of the dataset by removing the unnecessary and correlated data that warranted an increase in the dimensional input. In recent time, an extension of PCA called Robust PCA was investigated. It aims to separate abnormal traffic flow pattern caused by sensors and incidents that are faulty and then identify the cause of the anomaly.

Chapter 3

Traffic State Estimation with Particle Methods in the Presence of Sparse Data

3.1 Introduction

Traffic state estimation and forecasting is an essential part of the Intelligent Transportation System (ITS) for effective traffic monitoring and control. Most traffic estimation approaches are model-based [17], while the new trend is to develop data-driven approaches [33, 18]. Traffic modelling methods are used to understand the evolution of traffic and estimate the traffic state [73, 1, 41, 11].

An overview of different models is given in [42, 11, 12]. These include microscopic, macroscopic and mesoscopic models. Microscopic traffic models [13, 11, 12, 14], describe the motion of each individual vehicle with a high level of detail. Macroscopic models [15, 16] represent the aggregated behaviour of the traffic, usually in terms of the average speed and the average density. Mesoscopic models [172] use varying levels/degrees of detail to model traffic behaviour. Some areas are modelled with aggregated measurements as in macroscopic models and at other areas the detail goes down to individual vehicles as in microscopic models. Due to its computational efficiency for most practical purposes such as traffic management, road pricing and changes in infrastructure, the macroscopic model is enough to produce acceptable estimation.

The cell transmission model (CTM) [16] models traffic flow using macroscopic details by dividing the road into contiguous segments called cells. An extension of CTM, the stochastic compositional model (SCM) for traffic flow [1] uses probability distributions known as sending and receiving functions, which control the number of vehicles that could leave from one cell to the next, to model the stochastic nature of traffic state evolution.

The SCM was employed with the particle filter (PF) for estimating traffic state in motorways in [2]. Measurements at the boundaries were used to estimate the traffic state within the segments. It was reported that estimation accuracy is affected at boundaries without measurements.

A major challenge in traffic prediction is the problem of missing or sparse data. Communication infrastructure upon which traffic measurements are transmitted for processing and utilisation often experience failure leading to missing data, which could be more than 40% in some cases [10]. The cost of installing and managing traffic sensing devices is high, making it impractical to cover all locations needed for effective observation of the full road network resulting in sparse data. In [173], a method of detecting outliers or missing data using kNN is proposed. Various methods and approaches have been applied by researchers to address these problems such as missing data imputation [35], compressive sensing and historical averages [174], Kriging interpolation [23].

In [175], a review of three different missing data imputation methods, interpolation, prediction and statistical learning is presented. The interpolation method uses the historical average of measurements from a given sensor at similar times of day (e.g. all weekdays at 9 am) to help cope with missing data. Prediction methods use a deterministic mathematical description to model the relationship between historical and future data. The statistical methods differ from the other two by modelling the stochastic nature of the traffic pattern into the imputation algorithm.

This chapter presents an approach to combine Kriging and particle filtering to address the challenge of sparse traffic data. It uses Kriging to compute missing values at unobserved locations, which are subsequently used in the computation of likelihood terms in the particle filter algorithm. This approach combines the benefits of Kriging, which is a powerful geospatial method and a particle filter, which can capture the stochastic variations in traffic flow.

3.2 Traffic Evolution and Measurement Model

3.2.1 Traffic Flow Model

Consider a system with the following state equation

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \boldsymbol{\eta}_k, \tag{3.1}$$

and observation equation

$$\mathbf{z}_k = g(\mathbf{x}_k) + \boldsymbol{\xi}_k. \tag{3.2}$$

Here $f(.)$ and $g(.)$ are the state and observation model functions, $\mathbf{x}_k$ is state vector, $\mathbf{z}_k$ is the measurement vector, $\boldsymbol{\eta}_k$ and $\boldsymbol{\xi}_k$ are the state and observation errors, respectively and k is discrete time index.

The stochastic compositional model (SCM) for traffic flow [1], which is an adaptation of the cell-transmission model [16], uses sending and receiving functions to model the stochastic nature of traffic state evolution. The sending functions represent the vehicles that can leave a cell while the receiving functions determine the vehicles that are allowed to enter a cell. Figure 3.1 shows how the road is divided into n segments, also called cells, with length L_i and l_i lanes. The number of vehicles crossing the boundary between segments i and $i+1$ at time k is represented by $Q_{i,k}$. $N_{i,k}$ represents the number of vehicles in segment i with average speed given by $v_{i,k}$.

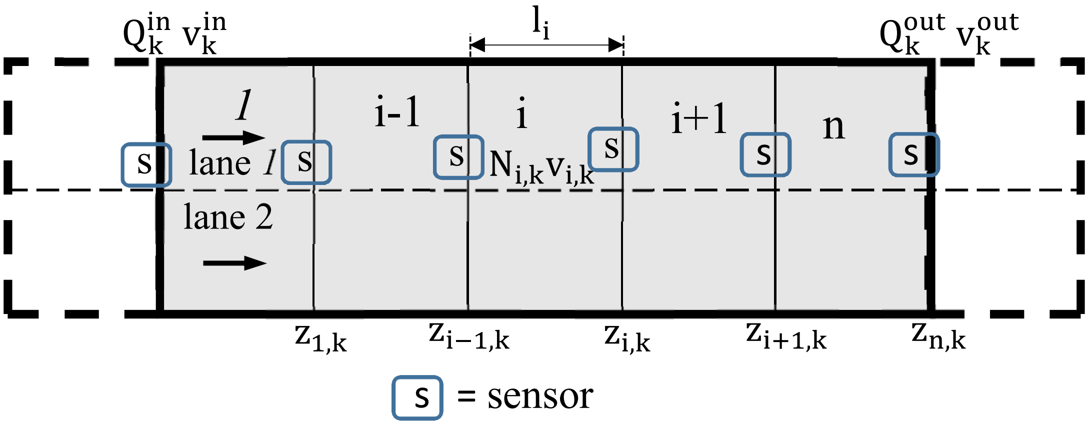

Figure 3.1 SCM road network showing segments and measurement points [1].

The overall state vector at time t_k is given by $\mathbf{x}_k = [\mathbf{x}_{1,k}^T, \mathbf{x}_{2,k}^T, ..., \mathbf{x}_{n,k}^T]^T$ where $\mathbf{x}_{i,k} = [N_{i,k}, v_{i,k}]^T$ is the local state vector at segment i.

The evolution of traffic state within the segments is modelled with equations (3.3) to (3.5).

$$\mathbf{x}_{1,k+1} = f_1(Q_k^{in}, v_k^{in}, \mathbf{x}_{1,k}, \mathbf{x}_{2,k}, \boldsymbol{\eta}_{1,k}). \tag{3.3}$$

$$\mathbf{x}_{i,k+1} = f_i(\mathbf{x}_{i-1,k}, \mathbf{x}_{i,k}, \mathbf{x}_{i+1,k}, \boldsymbol{\eta}_{i,k}). \tag{3.4}$$

$$\mathbf{x}_{n,k+1} = f_n(\mathbf{x}_{n-1,k}, \mathbf{x}_{n,k}, Q_k^{out}, v_k^{out}, \boldsymbol{\eta}_{n,k}). \tag{3.5}$$

The boundary conditions are the number of vehicles entering the first segment (inflow) Q_k^{in}, with average speed v_k^{in} and the number of vehicles leaving the last segment (outflow) Q_k^{out}, with corresponding average speed v_k^{out} within the time interval $\Delta t_k = t_{k+1} - t_k$. These are not estimated but supplied to the model by traffic sensors as boundary conditions. The reader is referred to as the Algorithm 1 for a detailed algorithm.

3.2.2 Measurement Model

For a road segment with n boundaries, the traffic state at a boundary $j \in \mathscr{J} = 1, 2, ..., n$ is sampled at discrete time steps t_s, $s = 1, 2, ...$, to give $\mathbf{z}_{j,s} = (Q_{j,s}, v_{j,s})^T$. The matrix of measurements taken at all of the n boundaries is given by $\mathbf{Z}_s = [\mathbf{z}_{1,s}^T, \mathbf{z}_{2,s}^T, ..., \mathbf{z}_{n,s}^T]^T$. The sampling interval Δt_s is usually split into q state update time steps Δt_k. That is, $\Delta t_s = q\Delta t_k$.

With the assumption of Gaussian noise, the measurement $\mathbf{z}_{j,s}$ can be expressed as:

$$\mathbf{z}_{j,s} = \begin{pmatrix} Q_{j,s} \\ \\ v_{j,s} \end{pmatrix} + \boldsymbol{\xi}_s. \tag{3.15}$$

Here, $Q_{j,s}$ is the number of vehicles crossing segment j within time step s with average speed $v_{j,s}$ and $\boldsymbol{\xi}_s = [\xi_{Q_{j,s}}, \xi_{v_{j,s}}]^T$ is the measurement error.

3.3 Spatial Traffic Flow Estimation Using Kriging

The Kriging algorithm [126] is a point-based estimation method which relies on exploiting the spatial correlation, of the data points. The Kriging algorithm attempts to interpolate the values at an unobserved location using statistics of the spatial variation between pairs of observed locations in a given region.

3.3.1 Random Set

Let $\mathbf{Z}(\mathbf{r})$ be a non-stationary set of m measurements $\mathbf{Z}(\mathbf{r}) = [\mathbf{z}^T(\mathbf{r}_1), \mathbf{z}^T(\mathbf{r}_2), \ldots, \mathbf{z}^T(\mathbf{r}_n)]^T$, in this work average vehicle speeds $\mathbf{v}(\mathbf{r})$ or vehicle counts $\mathbf{N}(\mathbf{r})$, observed in segments $i = 1, \ldots, n$ at locations $\mathbf{r}_i = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_n)$, where $\mathbf{r}_i = (r_i^x, r_i^z)^T$. Note that location of each

Algorithm 1 Stochastic Compositional Model (SCM) Algorithm [1]

1. Forward wave: for i = 1, 2, ..., n,

$$\mathbf{S}_{i,k} = max(N_{i,k}\frac{v_{i,k}.\Delta t_k}{L_i} + \eta_{S_i,k}, N_{i,k}\frac{v_{min}.\Delta t_k}{L_i}) \tag{3.6}$$

   then make $\mathbf{Q}_{i,k} = \mathbf{S}_{i,k}$

2. Backward wave: for $i = n, n-1, ..., 1$,

$$N^{max}_{i+1,k} = (L_{i+1}l_{i+1,l})/(A_l + v_{i+1,k}t_d) \tag{3.7}$$

$$\mathbf{R}_{i,k} = N^{max}_{i+1,k} - N_{i+1,k} + Q_{i+1,k}, \tag{3.8}$$

$$\text{if} \qquad \mathbf{S}_{i,k} < \mathbf{R}_{i,k}, \qquad \mathbf{Q}_{i,k} = \mathbf{S}_{i,k} \tag{3.9}$$

$$\text{else} \qquad \mathbf{Q}_{i,k} = \mathbf{R}_{i,k}, \qquad v_{i,k} = \mathbf{Q}_{i,k}\mathbf{L}_{i,k}/(\mathbf{N}_{i,k}\Delta t_k) \tag{3.10}$$

3. Recalculate the number of vehicles in each segment
   for $i = 1, 2, ..., n$,

$$\mathbf{N}_{i,k+1} = \mathbf{N}_{i,k} + \mathbf{Q}_{i-1,k} - \mathbf{Q}_{i,k} \tag{3.11}$$

4. Update the density

$$\rho_{i,k+1} = \mathbf{N}_{i,k+1}/(\mathbf{L}_i l_{i,k+1}) \tag{3.12}$$

$$\rho^{antic}_{i,k+1} = \phi\rho_{i,k+1} + (1-\phi)\rho_{i+1,k+1} \tag{3.13}$$

   where ϕ is a weighting coefficient

5. Recalculate the average speed of the vehicles in each segment

$$v^{interm}_{i,k} = \begin{cases} \frac{v_{i-1,k}\mathbf{Q}_{i-1,k} + v_{i,k}(\mathbf{N}_{i,k} - \mathbf{Q}_{i,k})}{\mathbf{N}_{i,k+1}}, & \text{for } \mathbf{N}_{i,k+1} \neq 0 \\ v_f, & \text{otherwise} \end{cases}$$

$$v^{interm}_{i,k+1} = max(v^{interm}_{i,k+1}, v_{min}) \tag{3.14}$$

   $v_{i,k+1} = \beta_{k+1}v^{interm}_{i,k+1} + (1-\beta_{k+1})v^e(\rho^{antic}_{i,k+1}) + \eta_{v_i,k+1}$
   where:
   $\beta_{k+1} = \begin{cases} \beta^I & if \ \ |\rho^{antic}_{i+1,k+1} - \rho^{antic}_{i,k+1}| \geq \rho_{threshold} \\ \beta^{II} & otherwise \end{cases}.$
   where β is a weighting coefficient.

sensor is uniquely defined by the segment topology (Fig. 3.1). Then, according to equation (3.16),

$$\mathbf{z}(\mathbf{r}) \sim \mathcal{N}\big(\boldsymbol{\mu}(\mathbf{r}), \mathbf{C}(\mathbf{r})\big) \tag{3.16}$$

the random set of variables $\mathbf{z}(\mathbf{r})$ can be approximated by a Gaussian process which is uniquely defined by the *drift*, i.e. the mean field $\boldsymbol{\mu}(\mathbf{r}) = E[\mathbf{z}(\mathbf{r})]$, and the corresponding *covariance* $\mathbf{C}(\mathbf{r}^T, \mathbf{r}) = \big(\mathbf{z}(\mathbf{r}) - \boldsymbol{\mu}(\mathbf{r})\big)^T \big(\mathbf{z}(\mathbf{r}) - \boldsymbol{\mu}(\mathbf{r})\big)$. Individual samples $z(\mathbf{r}_i)$ of the set (3.16) can be also expressed as

$$z(\mathbf{r}_i) = \mu(\mathbf{r}_i) + \delta(\mathbf{r}_i) \tag{3.17}$$

with *drift* $\mu(\mathbf{r}_i)$ and *residual* $\delta(\mathbf{r}_i)$. Based on the degree to which the moments, in this work mean and variance of the random set $\mathbf{z}(\mathbf{r})$ are dependent (or independent) on a spatial relationship between points $\mathbf{r}_i = (\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n)$, the 1^{st} or the 2^{nd}-order of stationarity can be recognised. While, the 1^{st}-order stationary random set assumes a constant mean $E[\mathbf{z}(\mathbf{r})] = \boldsymbol{\mu}(\mathbf{r}) = \mu$, the 2^{nd}-order stationary random field assumes a linear drift between the increments $E[\mathbf{z}(\mathbf{r})] = \boldsymbol{\mu}(\mathbf{r}) = \sum_{n=0}^{N} \alpha_n \mathbf{f}(\mathbf{r})$. In the rest of this sequel, it is assumed that the random set $\mathbf{z}(\mathbf{r})$ is *intrinsic* and stationary on the 2^{nd}-order.

3.3.2 Covariance

By assuming that random variables $\mathbf{z}(\mathbf{r})$ are 2^{nd}-order stationary and isotropic, the covariance function $\mathbf{C}(\mathbf{r}_i, \mathbf{r}_j)$ in (3.16) reads as follows:

$$\begin{aligned} \mathbf{C}(\mathbf{r}_i, \mathbf{r}_j) &= E\big[(\mathbf{z}(\mathbf{r}_i) - \mu)^T (\mathbf{z}(\mathbf{r}_j) - \mu)\big], \\ \mathbf{C}(\mathbf{r}_i, \mathbf{r}_j) &= E\big[(\mathbf{z}(\mathbf{r}_i) - \mu)^T (\mathbf{z}(\mathbf{r}_i + \mathbf{h}) - \mu)\big]. \end{aligned} \tag{3.18}$$

The role of covariance function $\mathbf{C}(\mathbf{r}_i, \mathbf{r}_j)$ is to model the correlation between measurements $z(\mathbf{r}_i)$ and $z(\mathbf{r}_j)$ observed at locations $\mathbf{r}_i$ and $\mathbf{r}_j$ based on their separation distance called a *lag* h. Since the correlation between two random variables solely depends on their spatial distance, and not at all on their location, the *lag* can be conveniently expressed as an Euclidean l_2 norm, defined as $h = ||\mathbf{r}_j - \mathbf{r}_i||_2$. Above statements can be summarized into the following equalities, i.e. the isotropy assumptions:

$$\begin{aligned} &\mathbf{h} = \mathbf{r}_j - \mathbf{r}_i \implies ||\mathbf{h}||_2 = h \implies \\ &\mathbf{C}(\mathbf{r}_i, \mathbf{r}_j) = \mathbf{C}(\mathbf{r}_i, \mathbf{r}_i + \mathbf{h}) = \mathbf{C}(h). \end{aligned} \tag{3.19}$$

For a straight stretch of motorway (Fig. 3.1) this is equivalent to the path length through the road network.

The process of covariance function $\mathbf{C}(\mathbf{r}_i,\mathbf{r}_j)$ modelling requires to find a covariance curve that has the best fit to the empirical data (3.18), possibly being a subject to constraints. The covariance models to choose from include exponential, spherical, Gaussian, linear or power model [21]. In this work, the best fit for the dataset was achieved by the exponential model given by

$$\mathbf{C}(\mathbf{r}_i,\mathbf{r}_j) = \begin{cases} c - c_0 & if|\mathbf{r}_j - \mathbf{r}_i| = 0, \\ (c-c_0)e^{-\frac{3|\mathbf{r}_j-\mathbf{r}_i|}{a}}\} & if|\mathbf{r}_j - \mathbf{r}_i| > 0, \end{cases} \tag{3.20}$$

where c_0 is the *nugget*, a is the *range* and c is *sill*.

3.3.3 Variogram

The nature of the 2^{nd}-order intrinsic stationary and isotropic random process $\mathbf{z}(\mathbf{r})$ (5.1) can be also described by a variogram

$$\begin{aligned} \boldsymbol{\gamma}(\mathbf{r}_i,\mathbf{r}_j) &= \frac{1}{2}\text{Var}[z(\mathbf{r}_i) - z(\mathbf{r}_j)], \\ \boldsymbol{\gamma}(\mathbf{r}_i,\mathbf{r}_j) &= \frac{1}{2}\text{Var}[z(\mathbf{r}_i) - z(\mathbf{r}_i + \mathbf{h})], \end{aligned} \tag{3.21}$$

in which the equalities (3.19), established for covariance function $\mathbf{C}(\mathbf{r}_i,\mathbf{r}_j)$, also apply to the variogram $\boldsymbol{\gamma}(\mathbf{r}_i,\mathbf{r}_j)$. The variogram shows how the dissimilarity between $z(\mathbf{r}_i)$ and $z(\mathbf{r}_i+\mathbf{h})$ evolves with a separation h. The variogram, unlike the covariance, does not require the knowledge of the mean $\boldsymbol{\mu}(\mathbf{r})$. In practice, this mean is not known and has to be estimated from the data, which introduces a bias.

The model of the variogram $\boldsymbol{\gamma}(\mathbf{r}_i,\mathbf{r}_j)$, or so called a theoretical variogram, is obtain from the data by the curve fitting. By using the exponential curve the theoretical variogram $\boldsymbol{\gamma}(\mathbf{r}_i,\mathbf{r}_j)$ can be computed as:

$$\boldsymbol{\gamma}(\mathbf{r}_i,\mathbf{r}_j) = \begin{cases} 0 & if|\mathbf{r}_j - \mathbf{r}_i| = 0, \\ a + (c-a)\big(1 - e^{-\frac{3|\mathbf{r}_j-\mathbf{r}_i|}{b}}\big) & if|\mathbf{r}_j - \mathbf{r}_i| > 0. \end{cases} \tag{3.22}$$

The variogram can be in terms of covariance function reconstructed by elements as:

$$\gamma(\mathbf{r}_i,\mathbf{r}_j) = c - C(\mathbf{r}_i,\mathbf{r}_j). \tag{3.23}$$

$$\begin{aligned}\gamma(\mathbf{r}_i,\mathbf{r}_j) &= \frac{1}{2}\mathrm{Var}[z(\mathbf{r}_j)-z(\mathbf{r}_i)] \\ &= \frac{1}{2}\big(\sigma(\mathbf{r}_i,\mathbf{r}_i)+\sigma(\mathbf{r}_j,\mathbf{r}_j)-2\sigma(\mathbf{r}_i,\mathbf{r}_j)\big) \\ &= \frac{1}{2}\big(\sigma^2(\mathbf{r}_i)+\sigma^2(\mathbf{r}_j)-2\sigma(\mathbf{r}_i,\mathbf{r}_j)\big), \\ &= \sigma^2-\sigma(\mathbf{r}_i,\mathbf{r}_j).\end{aligned} \tag{3.24}$$

Note, that by assuming an isotropic covariance matrix the following equalities $\sigma(\mathbf{r}_i,\mathbf{r}_i) = \mathrm{Var}(z(\mathbf{r}_i)) = \sigma(h=0) = \sigma^2$ and $\sigma^2(\mathbf{r}_i) = \sigma^2(\mathbf{r}_j) = \sigma^2$ were used in the calculations (3.24), where σ^2 is the *still*.

3.3.4 Interpolation of the Random Set

Typically, one is not only interested in describing the random set $\mathbf{z}(\mathbf{r})$ from the observed locations $\mathbf{r}_i = (\mathbf{r}_1,\mathbf{r}_2,\ldots,\mathbf{r}_m)$, but the interest is to predict the values of this random set of observations at the n new (unobserved) locations $\mathbf{r}_u = (\mathbf{r}_1,\mathbf{r}_2,\ldots,\mathbf{r}_n)$ with $u=(1,\ldots,n)$, i.e. to find $\mathbf{z}(\mathbf{r_u}) = [z(\mathbf{r}_1),z(\mathbf{r}_2),\ldots,z(\mathbf{r}_n)]$ such that $\mathbf{z}(\mathbf{r_u}) \subset \mathbf{z}(\mathbf{r})$. In general, the process of the interpolation can be described in these three steps:

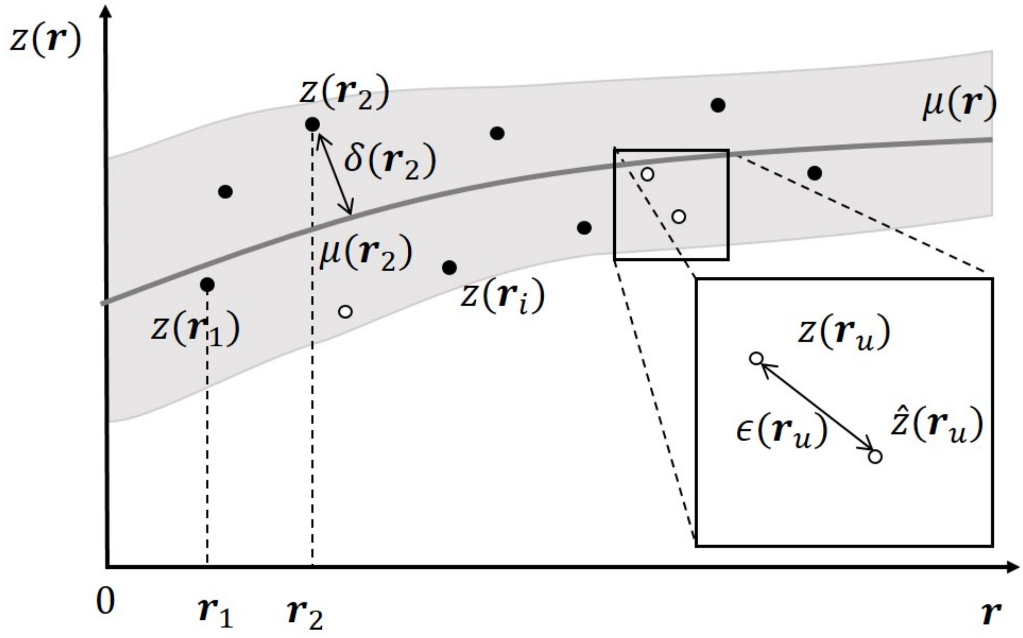

Figure 3.2 Interpolation at point $z(\mathbf{r}_u)$ within the set $\mathbf{z}(\mathbf{r})$.

1. Construct an estimator $E[\hat{\mathbf{z}}(\mathbf{r})] = \hat{\boldsymbol{\mu}}(\mathbf{r})$ of the drift $E[\mathbf{z}(\mathbf{r})] = \boldsymbol{\mu}(\mathbf{r})$ for the random set $\mathbf{z}(\mathbf{r})$ (3.16).
2. Compute an interpolation error $\boldsymbol{\varepsilon}(\mathbf{r_u}) = \mathbf{z}(\mathbf{r_u}) - \hat{\mathbf{z}}(\mathbf{r_u})$ in terms of mean $E\left[\boldsymbol{\varepsilon}(\mathbf{r_u})\right]$ and variance $\mathrm{Var}\left[\boldsymbol{\varepsilon}(\mathbf{r_u})\right]$. Interpolate the value of $\hat{\mathbf{z}}(\mathbf{r}_u)$ at locations $\mathbf{r}_u$ such that mean of the squared deviations $\mathrm{Var}\left[\boldsymbol{\varepsilon}(\mathbf{r_u})\right]$ will be minimal.
3. Finally, the interpolated values $\hat{\mathbf{z}}(\mathbf{r_u})$ alongside with their corresponding confident intervals are given as $\left[\hat{\mathbf{z}}(\mathbf{r_u}) + \mathrm{Var}\left[\boldsymbol{\varepsilon}(\mathbf{r_u})\right], \hat{\mathbf{z}}(\mathbf{r_u}) - \mathrm{Var}\left[\boldsymbol{\varepsilon}(\mathbf{r_u})\right]\right]$.

The interpolation technique utilized in this work is based on Kriging, discussed in Sec. 3.3.6, which yields a predictor of the value $\hat{z}(\mathbf{r}_u)$ of $z(\mathbf{r}_u)$, i.e. a single element of $\mathbf{z}(\mathbf{r_u})$, for the location $\mathbf{r}_u = [r_u^x, r_u^z]^T$ based on the knowledge of the random field variables $\mathbf{z}(\mathbf{r})$ expressed either by covariance function $\mathbf{C}(\mathbf{r}_i, \mathbf{r}_j)$ (3.18). In order to obtain $\hat{z}(\mathbf{r}_u)$ at the unobserved location $\mathbf{r}_u$, the distance measure $\mathbf{h} = \mathbf{r}_i - \mathbf{r}_u$, between all observed locations $\mathbf{r}_i$ and the point of an interest r_u, is used to calculate variance $\mathbf{C}(\mathbf{r}_i, \mathbf{r}_u)$ on the basis of theoretical covariance (3.20).

3.3.5 Nugget, Sill and Range

If a graph of the lag and variogram of the measurements are plotted, then Figure 3.3 is generated. Three points on the graph are useful in the computation of the model variogram for finding the Kriging weight. The variogram increases with the lag between points up to a point where it flattens out and remains constant. All pairs of location within the range are correlated while locations beyond than the range are not. The distance at which this flattening start is called the *range* and the value of the variogram at that point is called the *sill.* All points at a location less than the range are correlated, and all points with separation more than the range are not correlated. At a lag distance of zero $\gamma(0)$ is expected to be zero. This is not so in practice but exhibits what is known as *nugget* effect. This is attributed to either error in the measurements or when the spatial variation is less than the sampling rate.

The computed empirical semi-variogram from the given number of observed locations is used to determine the parameters (*nugget*, *range* and *sill* in the chosen variogram model. To obtain values at unobserved locations, the values of these parameters are then used to calculate the co-variogram matrix between all the observed locations and the unobserved locations required to find the Kriging weights. The variogram models to choose from are exponential, spherical, Gaussian, linear or power model [21]. The exponential model was

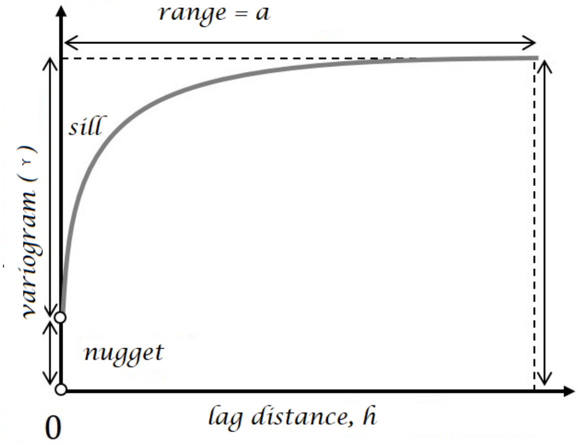

Figure 3.3 Plot of empirical variogram γ and lag h

used in this work because it gives the best fit for the dataset. This is given by,

$$\gamma(h) = c_0 + c\{1 - e^{-\frac{h}{a}}\}, \tag{3.25}$$

where c_0 is the nugget, and $c = sill - c_0$ is the maximum of the correlated variance and a is the range.

3.3.6 Ordinary Kriging

The underlying idea of Kriging is to obtain the response $z(\mathbf{r}_u)$, interpreted as a random variable positioned at the location $\mathbf{r}_u$, by interpolating random variables $\mathbf{z}(\mathbf{r}) = [z(\mathbf{r}_1), z(\mathbf{r}_2), \dots, z(\mathbf{r}_m)]^T$ from (3.16), i.e. observations $z(\mathbf{r}_i)$, $i = 1, \dots, m$ at locations $\mathbf{r}_i$. The Kriging predictor incorporates the covariance structure among the observation points $z(\mathbf{r}_i)$ into the weights $\mathbf{w}(\mathbf{r}_u) = [w_1(\mathbf{r}_u), w_2(\mathbf{r}_u), \dots, w_m(\mathbf{r}_u)]^T$ for predicting $\hat{z}(\mathbf{r}_u)$ as a linear combination:

$$\hat{z}(\mathbf{r}_u) = \sum_{i=1}^{m} w_i(\mathbf{r}_u) z(\mathbf{r}_i) = \mathbf{w}^T(\mathbf{r}_u)\mathbf{z}(\mathbf{r}). \tag{3.26}$$

In order to assess the accuracy of the Kriging prediction $\hat{z}(\mathbf{r}_u)$ w.r.t the real (true) value $z(\mathbf{r}_u)$ an error $\varepsilon(\mathbf{r}_u)$ is declared

$$\varepsilon(\mathbf{r}_u) = z(\mathbf{r}_u) - \hat{z}(\mathbf{r}_u). \tag{3.27}$$

The following criteria, evaluated in terms of mean $E\left[\varepsilon(\mathbf{r}_u)\right]$ and variance $\mathrm{Var}\left[\varepsilon(\mathbf{r}_u)\right] = \sigma_\varepsilon^2(\mathbf{r}_u)$ of the prediction error $\varepsilon(\mathbf{r}_u)$ (3.27), apply to any type of Kriging interpolation:

- *Lack of bias*, implies that $E\left[\varepsilon(\mathbf{r}_u)\right]=0$ and therefore $E\left[\hat{z}(\mathbf{r}_u)\right]=E\left[z(\mathbf{r}_u)\right]$. Thus, the Kriging interpolation is said to be globally unbiased (3.28).

$$E[\mathbf{z}(\mathbf{r_u})-\hat{\mathbf{z}}(\mathbf{r_u})]=0 \tag{3.28}$$

- *Minimum variance*, implies that the mean of the squared deviations $\mathrm{Var}[\varepsilon(\mathbf{r}_u)]=\sigma_\varepsilon^2(\mathbf{r}_u)$ must be minimal (3.29),

$$\min_{\mathbf{1}^T\mathbf{w}=1} \mathrm{Var}[z(\mathbf{r}_u)-\hat{z}(\mathbf{r}_u)] \quad \text{or} \quad \min_{\mathbf{1}^T\mathbf{w}=1} \sigma_\varepsilon^2(\mathbf{r}_u), \tag{3.29}$$

subject to

$$\sum_{i=1}^{m} w_i=\mathbf{1}^T\mathbf{w}=\begin{bmatrix} 1\ldots1 \end{bmatrix}\begin{bmatrix} w_1 \\ \vdots \\ w_m \end{bmatrix}=1. \tag{3.30}$$

The ordinary Kriging model was initially developed in spatial statistics by Krige [126] and subsequently extended by Matheron [127] and Cressie [176]. The goal of the ordinary Kriging is to interpolate a value of the function z at an unobserved location $\mathbf{r}_u$ by computing a linear combination (3.26) of the observations $z(\mathbf{r}_i)$ at locations $\mathbf{r}_i$. Kriging weights $\mathbf{w}(\mathbf{r}_u)$ are chosen such that the mean squared prediction error $\sigma_\varepsilon^2(\mathbf{r}_u)$ (3.29), also known as Kriging variance or Kriging error is minimized as $\min\limits_{\mathbf{1}^T\mathbf{w}=1} \sigma_\varepsilon^2(\mathbf{r}_u)$ overall $\mathbf{z}(\mathbf{r_u})$ (3.29) subject to the condition of unbiased $E[\mathbf{z}(\mathbf{r_u})-\hat{\mathbf{z}}(\mathbf{r_u})]=0$ (3.28) and $\sum_{i=1}^{m} w_i=\mathbf{1}^T\mathbf{w}=1$ (3.30).

$$\begin{aligned}
\sigma_\varepsilon^2(\mathbf{r}_u) =&\mathrm{Var}[z(\mathbf{r}_u)-\hat{z}(\mathbf{r}_u)] \\
=&\mathrm{Var}\left[z(\mathbf{r}_u)\right]+\mathbf{w}^T(\mathbf{r}_u)\mathrm{Var}\left[\mathbf{z}(\mathbf{r})\right]\mathbf{w}(\mathbf{r}_u) \\
&-\mathbf{b}\left(\mathbf{r}_i,\mathbf{r}_u\right)^T\mathbf{w}(\mathbf{r}_u)-\mathbf{b}\left(\mathbf{r}_i,\mathbf{r}_u\right)\mathbf{w}^T(\mathbf{r}_u) \\
=&\mathbf{b}\left(\mathbf{r}_u,\mathbf{r}_u\right)+\mathbf{w}^T(\mathbf{r}_u)\mathbf{C}\left(\mathbf{r}_i,\mathbf{r}_j\right)\mathbf{w}(\mathbf{r}_u) \\
&-\mathbf{b}\left(\mathbf{r}_i,\mathbf{r}_u\right)^T\mathbf{w}(\mathbf{r}_u)-\mathbf{b}\left(\mathbf{r}_i,\mathbf{r}_u\right)\mathbf{w}^T(\mathbf{r}_u) \\
=&\sigma^2(\mathbf{r}_u)+\sum_{i=1}^{m}\sum_{j=1}^{m} w_i(\mathbf{r}_u)w_j(\mathbf{r}_u)\mathbf{C}(\mathbf{r}_i,\mathbf{r}_j) \\
&-2\sum_{i=1}^{m} w_i(\mathbf{r}_u)\mathbf{b}(\mathbf{r}_i,\mathbf{r}_u).
\end{aligned} \tag{3.31}$$

Note that the Kriging variance $\sigma_\varepsilon^2(\mathbf{r}_u)$, introduced in (3.31), differs from the variance of the Kriging predictor $\hat{z}(\mathbf{r}_u)$ itself (3.32).

$$\mathrm{Var}(\hat{z}(\mathbf{r}_u)) = \mathrm{Var}\sum_{i=1}^{m} w_i z(\mathbf{r}_i) = \sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j C(\mathbf{r}_i, \mathbf{r}_j). \tag{3.32}$$

The minimization of the Kriging variance (3.31) can be in terms of covariance model expressed as (3.33)

$$\begin{aligned}
&\min_{\mathbf{1}^T\mathbf{w}=1}\Big\{\mathrm{Var}\big[z(\mathbf{r}_u)\big] + \mathbf{w}^T(\mathbf{r}_u)\mathrm{Var}\big[\mathbf{z}(\mathbf{r})\big]\mathbf{w}(\mathbf{r}_u) \\
&\qquad -\mathbf{b}\big(\mathbf{r}_i,\mathbf{r}_u\big)^T\mathbf{w}(\mathbf{r}_u) - \mathbf{C}\big(\mathbf{r}_i,\mathbf{r}_u\big)\mathbf{w}(\mathbf{r}_u)^T\Big\} \\
&\min_{\mathbf{1}^T\mathbf{w}=1}\Big\{\sigma^2(\mathbf{r}_u) + \sum_{i=1}^{m}\sum_{j=1}^{m} w_i(\mathbf{r}_u)w_j(\mathbf{r}_u)C(\mathbf{r}_i,\mathbf{r}_j) \\
&\qquad -2\sum_{i=1}^{m} w_i(\mathbf{r}_u)C(\mathbf{r}_i,\mathbf{r}_u)\Big\},
\end{aligned} \tag{3.33}$$

where matrix $\mathbf{C}(\mathbf{r}_i, \mathbf{r}_j)$ is the covariance between the individual samples given by,

$$\mathbf{C}(\mathbf{r}_i,\mathbf{r}_j)\begin{bmatrix} C(\mathbf{r}_1,\mathbf{r}_1) & \dots & C(\mathbf{r}_1,\mathbf{r}_m) \\ \vdots & \ddots & \vdots \\ C(\mathbf{r}_m,\mathbf{r}_1) & \dots & C(\mathbf{r}_m,\mathbf{r}_m) \end{bmatrix}, \tag{3.34}$$

and column vector $\mathbf{b}(\mathbf{r}_i, \mathbf{r}_u)$ is a covariance function between samples and the point to interpolate.

$$\mathbf{b}(\mathbf{r}_i,\mathbf{r}_u) = \begin{bmatrix} b(\mathbf{r}_1,\mathbf{r}_u) \\ \vdots \\ b(\mathbf{r}_m,\mathbf{r}_u) \end{bmatrix}. \tag{3.35}$$

Equation (3.33) is minimized by introducing a Lagrange multiplier, -2λ with the constraint $\mathbf{w}^T\mathbf{1} = 1$, where $\mathbf{1}$ is a vector of ones.

$$MSE = \gamma(\mathbf{y}_u) + \mathbf{w}^T\mathbf{A}\mathbf{w} - 2\mathbf{w}^T\mathbf{b} + 2\lambda(\mathbf{w}^T\mathbf{1} - 1). \tag{3.36}$$

By partial differentiation wrt $\mathbf{w}$ we get:

$$\frac{\partial MSE}{\partial \mathbf{w}} = 2\mathbf{A}\mathbf{w} - 2\mathbf{b} + 2\lambda\mathbf{1} = 0. \tag{3.37}$$

This implies that:

$$\mathbf{Cw} = \mathbf{b} - \lambda \mathbf{1}. \tag{3.38}$$

The Lagrange multiplier, λ is computed by solving equation (3.38) by direct substitution of values,

$$\begin{array}{lcl} \lambda \mathbf{1} & = & \mathbf{b} - \mathbf{Cw} \\ \lambda \mathbf{C}^{-1}\mathbf{1} & = & \mathbf{A}^{-1}\mathbf{b} - \mathbf{w} \\ \lambda \mathbf{1}^T\mathbf{C}^{-1}\mathbf{1} & = & \mathbf{1}^T\mathbf{C}^{-1}\mathbf{b} - \underbrace{\mathbf{1}^T\mathbf{w}}_{1} \\ \lambda & = & \frac{\mathbf{1}^T\mathbf{C}^{-1}\mathbf{b} - 1}{\mathbf{1}^T\mathbf{C}^{-1}\mathbf{1}}. \end{array} \tag{3.39}$$

The weights $\mathbf{w}(\mathbf{r}_u)$ for ordinary Kriging can be found from the following system of m linear equations (3.40), known as the Kriging equation.

$$\mathbf{w}(\mathbf{r}_u) = \mathbf{C}(\mathbf{r}_i, \mathbf{r}_j)^{-1}\mathbf{b}(\mathbf{r}_i, \mathbf{r}_u) - \mu \mathbf{1}. \tag{3.40}$$

Algorithm 2 gives a summary of this Kriging interpolation procedure.

After computing the weights, estimated values at unknown location are given by (3.26) and their variance computed as:

$$\begin{array}{lcl} \sigma_{e_u}^2 & = & Var[\mathbf{z}_u - \hat{\mathbf{z}}_u] \\ & = & \gamma_u + \mathbf{w}^T\mathbf{Cw} - 2\mathbf{w}^T\mathbf{b} \\ & = & \gamma_u + \mathbf{w}^T(\mathbf{b} - \lambda \mathbf{1}) - 2\mathbf{w}^T\mathbf{b} \\ & = & \gamma_u - \mathbf{w}^T\mathbf{b} - \lambda. \end{array} \tag{3.41}$$

Equation (3.41) provides useful information about how confident the results are with the estimation accuracy. This information is used to compute the weighting factor to improve the computation of particle predictor likelihood.

The next section describes traffic estimation using Bayesian inference and particle filtering approaches. It presents a method of improving the particle filter likelihood computation using Kriging to estimate any missing measurements.

Algorithm 2 Kriging Algorithm for Spatial Interpolation [177]

```
1. Determine location of all sensors (measurement/input points)
2. Compute the distance (lag, $h$) between all measurement locations
   For $u = 1$ to $U$, $U$ number of unknown locations to be computed
   Do the following
     (a) Determine the measurement locations that will contribute to the interpolation at each unknown location $y_u$.
     (b) Compute the distance between all measurement locations in the above step.
     (c) Compute the empirical semivariogram of all the contributory measurement location pairs above.
     (d) Fit the exponential semi-variogram model to obtain the nugget, sill and range.
     (e) Compute the distances of point $y_u$ to all the measurement locations identified in step (a).
     (f) Compute the semivariogram of the distances above.
     (g) Compute the vector $\mathbf{w}$ containing the weight factors of the point $u$ using (3.40).
     (h) Compute the estimated value of this point $u$ using (3.26).
```

3.4 Traffic Estimation via Bayesian Inference and Particle Filtering

3.4.1 Bayesian Estimation of Traffic State

In Bayesian estimation the posterior probability density function (PDF) $p(\mathbf{x}_k|\mathbf{Z}^k)$ of the traffic state $\mathbf{x_k}$ at time t_k is evaluated, given a set of measurements $\mathbf{Z}^k = \{\mathbf{z}_{1:k}\}$, collected up to time t_k using Bayes' rule as:

$$p(\mathbf{x}_k|\mathbf{Z}^k) = \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{Z}^{k-1})}{p(\mathbf{z}_k|\mathbf{Z}^{k-1})}. \tag{3.42}$$

The likelihood, $p(\mathbf{z}_k|\mathbf{x}_k)$ is defined by the observation model (3.2), and $p(\mathbf{z}_k|\mathbf{Z}^{k-1})$ is a normalizing constant. The prior or state prediction $p(\mathbf{x}_k|\mathbf{Z}^{k-1})$ is updated recursively using the Chapman-Kolmogorov equation given by [178]:

$$p(\mathbf{x}_k|\mathbf{Z}^{k-1}) = \int_{\mathbb{R}^{n_x}} p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{Z}^{k-1})d\mathbf{x}_{k-1}. \tag{3.43}$$

When the system model (3.1) is linear, equations (3.42) and (3.43) are analytically tractable and the Kalman filter [104] is used to obtain optimal solutions under certain constraints. When the system is highly non-linear, the recursive solution becomes expensive to compute, and numerical approximations methods such as the extended Kalman filter [105, 106] and particle filter [178, 179, 2] are often employed to obtain acceptable solutions.

3.4.2 Particle Filtering for Traffic State Estimation

The particle filter estimates the traffic state by taking enough random samples from the PDF with assigned weights. When a new measurement becomes available, it is used to compute what is known as the likelihood and a normalised form of the weights computed. The new state of the system is then updated with the computed weights. Degeneracy can be avoided by re-sampling, i.e. removing particles with low weights and replicating those with high weights [178].

Improved Likelihood Computation

The likelihood function term $p(\mathbf{z}_k|\mathbf{x}_k)$, is computed when a new measurement arrives. For the multivariate Gaussian distribution, the PDF is given by:

$$p(\mathbf{z}_k|\mathbf{x}_k) = \frac{1}{\sqrt{2\pi|\mathbf{R}|}} e^{-0.5\boldsymbol{\upsilon}\mathbf{R}^{-1}\boldsymbol{\upsilon}^T}, \tag{3.44}$$

where $\mathbf{R}$ is the covariance matrix of the measurement data, $|\mathbf{R}| \equiv det(\mathbf{R})$ is the determinant of $\mathbf{R}$ and $\boldsymbol{\upsilon}$ is the difference between the PF predicted value ($\bar{\mathbf{z}}_s$) and measurement ($\mathbf{z}_s$), given by:

$$\boldsymbol{\upsilon} = \mathbf{z}_s - \bar{\mathbf{z}}_s. \tag{3.45}$$

The measurement matrix $\mathbf{z}_s$ can be expressed as,

$$\mathbf{z}_s = \left\{ \begin{array}{ll} \mathbf{z}_s^{meas} & \text{measurement from sensor;} \\ \\ \hat{\mathbf{z}}_s^{krig} \circ \boldsymbol{\beta}_s & \text{estimated by Kriging} \end{array} \right\}, \tag{3.46}$$

where $\hat{\mathbf{z}}_s^{krig}$ represents the value estimated by Kriging (when measurement is not available), $\circ$ is the Hadamard product, $\boldsymbol{\beta}_s = [\beta_{s,1}, \beta_{s,2}, ..., \beta_{s,n}]^T$ is a weighting factor introduced to vary the level of confidence placed on the Kriged values. The value of β is 1 if estimator is fully confident about the Kriging result and less than 1 otherwise.

The modified particle filter procedure is shown in Algorithm 2.

3.5 Performance Evaluation

3.5.1 Investigation with Synthetic Data

Here, the performance of the particle filter with Kriging for traffic state estimation is evaluated using synthetic data from a 4km stretch of the motorway over a period of three hours. This is split into eight segments, each with a length of 0.5km and three lanes. For more details on the process for obtaining the synthesised data, see [2]. The benchmark for comparison is the Kriging model which has been reported in literature [136, 19]. These approaches used the traditional Kriging method without modification.

The modelling consists of periods of normal flow and congestion which was modelled by random changes (increase and decrease) in the inflow between time interval of $(1.12 \leq t < 1.17) hours$ and $(1.70 \leq t < 1.82) hours$ and outflow speed (decrease) between $(2.40 \leq t \leq 2.65) hours$.

To test the effect of assigning different values between 0 and 1 to $\boldsymbol{\beta}_s$, the simulation was repeated three times each with 200 independent Monte Carlo runs. First, by using all the measurements available at the segment boundaries. Second, removing measurements at two locations (boundary segments) and interpolating them using Kriging with equal weights assigned to the interpolated measurement and actual measurement in the likelihood computation. Lastly, by assigning a weight of 0.2 to the Kriging interpolated values.

In order to test the prediction accuracy for different levels of sparsity, three statistical measures namely the root mean squared error (RMSE) (2.9), the absolute percentage error (APE) (3.47)and the mean absolute error (MAE) (3.48) were computed. Note, $\mathbf{z}_i$ is the ground

Algorithm 3 PF Algorithm for Prediction with Improved Likelihood [2]

```
1. Initialization
   At $k = 0$; define all boundary conditions: number of samples, weight of samples as below,
   For $i = 1, ...N_p$, $N_p$ number of particles;
      • generate $N_p$ samples $\{\mathbf{x}_0^{(i)}\}$ from the initial distribution $p(\mathbf{x}_0)$
      • initialize the particle weights $w_0^{(i)} = \frac{1}{N_p}$.
   End for
2. Start the iteration for $k = 1, 2, ...$
   (a) Prediction stage
       For $i = 1, ..., N_p$,
       sample $\mathbf{x}_k^{(i)} \sim p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)})$ according to SCM model equations
       End for
   (b) Use measurements to compute likelihoods and update the weights
       This step is performed when the sampling time $t_s$ equals the iteration count $t_k$
       i. Estimate missing measurements with Kriging using Algorithm 1
       ii. Compute the likelihoods
       Use model (3.15) to compute the likelihood, $p(\mathbf{z}_s|\mathbf{x}_s^{(i)})$ of the particles
       iii. Update the weights of the particles using the likelihood $p(\mathbf{z}_s|\mathbf{x}_s^{(i)})$ calculated from model (3.15)
       For $i = 1, ..., N_p$
       $\omega_s^{(i)} = \omega_{s-1}^{(i)} p(\mathbf{z}_s|\mathbf{x}_s^{(i)})$
       End For
       iv. Normalize the weights: $\hat{\omega}_s^{(i)} = \frac{\omega_s^{(i)}}{\sum_{i=1}^{N_p} \omega_s^{(i)}}$.
   (c) Update the predicted states (Output): $\hat{\mathbf{x}}_s = \sum_{i=1}^{N} \hat{\omega}_s^{(i)} \mathbf{x}_s^{(i)}$
   (d) Re-sample the weights (Selection) only when $t_k = t_s$
```

truth or actual measurement, $\hat{\mathbf{z}}_i$ is the estimated value and m_r is the number of independent Monte Carlo runs.

$$APE = \sum_{i=1}^{m_r} \frac{|[\mathbf{z}_i - \hat{\mathbf{z}}_i]|}{\mathbf{z}_i} * 100, \tag{3.47}$$

$$MAE = \frac{1}{m_r} \sum_{i=1}^{m_r} |[\mathbf{z}_i - \hat{\mathbf{z}}_i]|. \tag{3.48}$$

The RMSE and APE for 200 independent Monte Carlo runs were plotted in Figures 3.4 and 3.5, respectively. The results show that there was an improvement in the estimation accuracy when the Kriging interpolated values were assigned weights. The results for the two Kriged examples reach the same accuracy for location 4. When the middle sensor is removed, there is still information up and downstream from the missing sensor. As a result, more information can be applied to the interpolation process allowing a more accurate estimate. Instead, when there is no sensor downstream; for example, there is less information; therefore, the interpolation is less accurate.

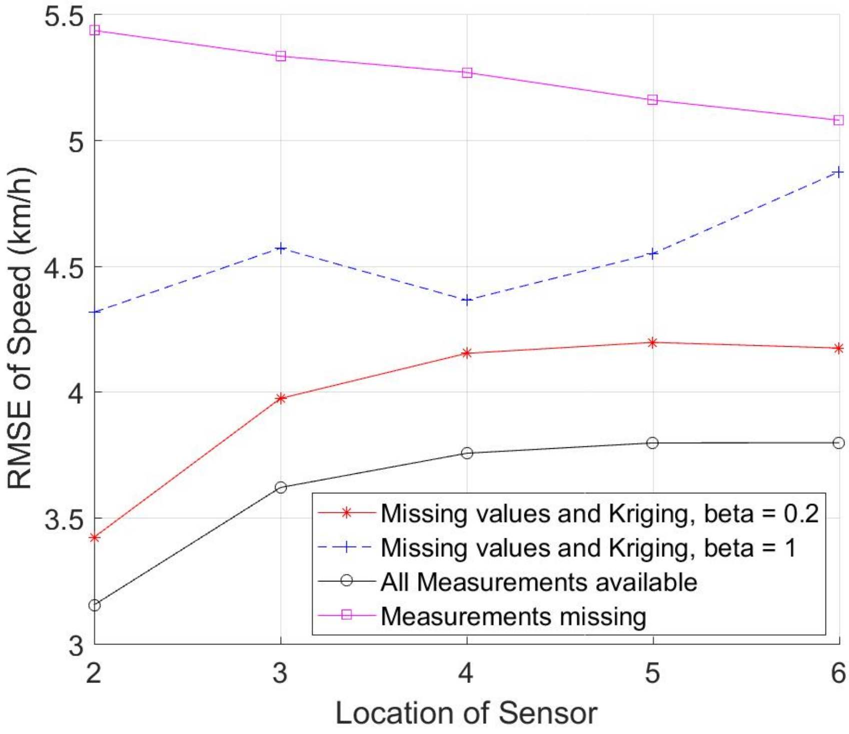

Figure 3.4 The root mean squared error (RMSE) of speed over locations

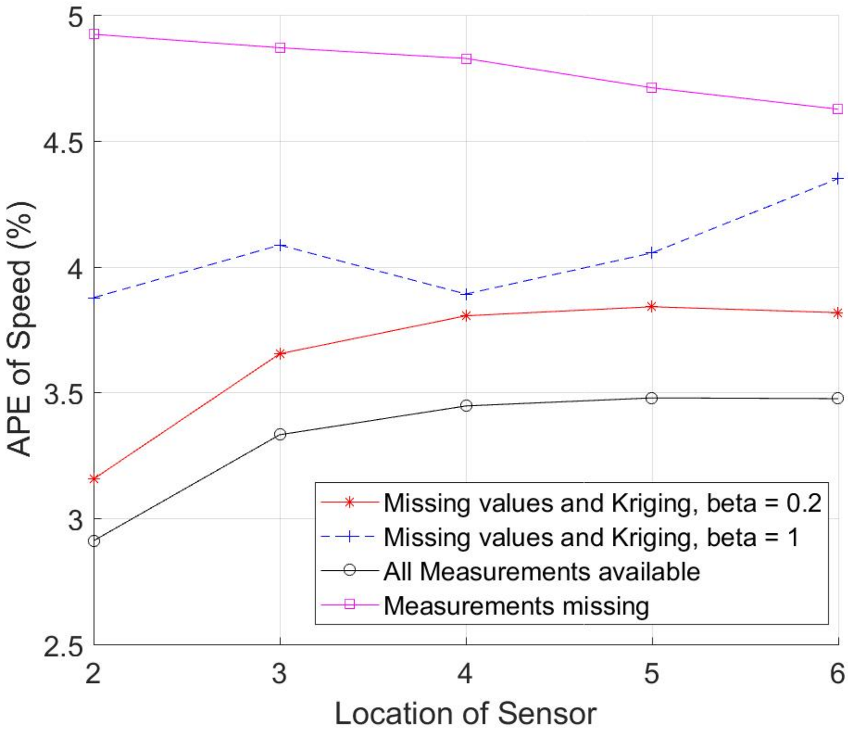

Figure 3.5 The absolute percentage error (APE) of speed over locations

The results in Figures 3.6 and 3.7 show that estimation accuracy is better when all the measurements are used in computing the likelihood, followed by that computed with Kriging interpolated measurements. The accuracy of the estimation performed without using any measurement is the least as can be seen in the figures. The results show improvement of 23% to 36.34% at different segments in RMSE values for the synthetic data used.

Figures 3.8 and 3.9 shows the plot of velocity and flow at segments boundaries 2 to 6. Segments 1 and 8 were not included as they were the inflow and outflow segments. It is evident that the estimation when all measurements (second plot from bottom) are used provides the best accuracy, followed by that where Kriging was used to estimate the missing measurements (third plot from bottom). The least accurate result was obtained when the missing values were not included in the likelihood computation (first plot from the top). This is apparent during congestion between time interval $(2.40 \leq t \leq 2.65)$ hours.

Another observation from the figures is that the estimation accuracy is consistent under free-flow conditions and begins to get worse as congestion sets in. During the period when the network is congested from time interval $(2.40 \leq t \leq 2.65)$ hours the estimate without the full measurements used in the likelihood computation could not capture the decrease in

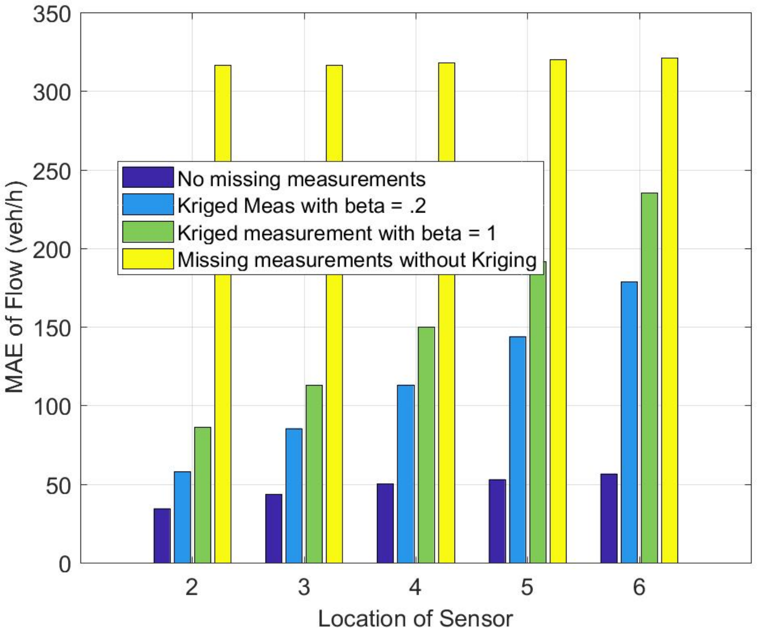

Figure 3.6 The mean absolute error (MAE) of flow for locations

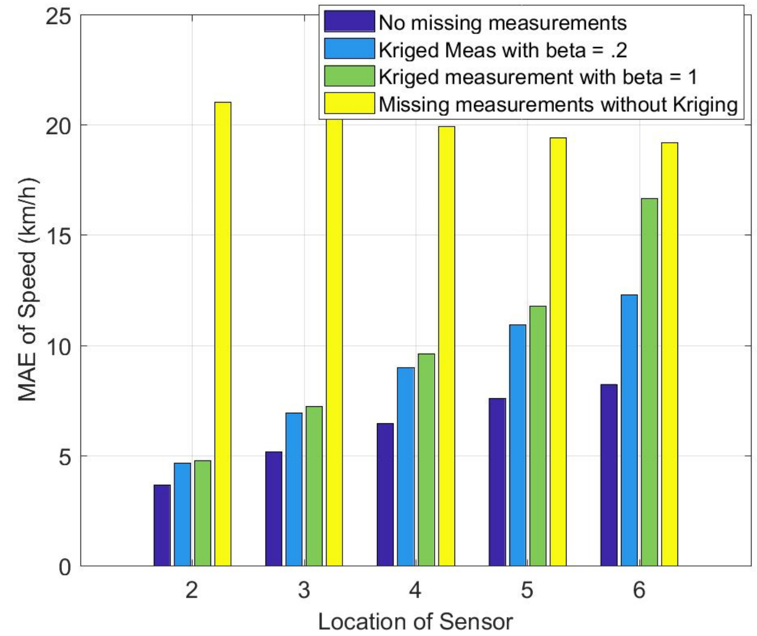

Figure 3.7 The mean absolute error (MAE) of speed for locations

speed. Incorporating the Kriged values improves the accuracy a little while the estimate with full measurements is closest to the true value.

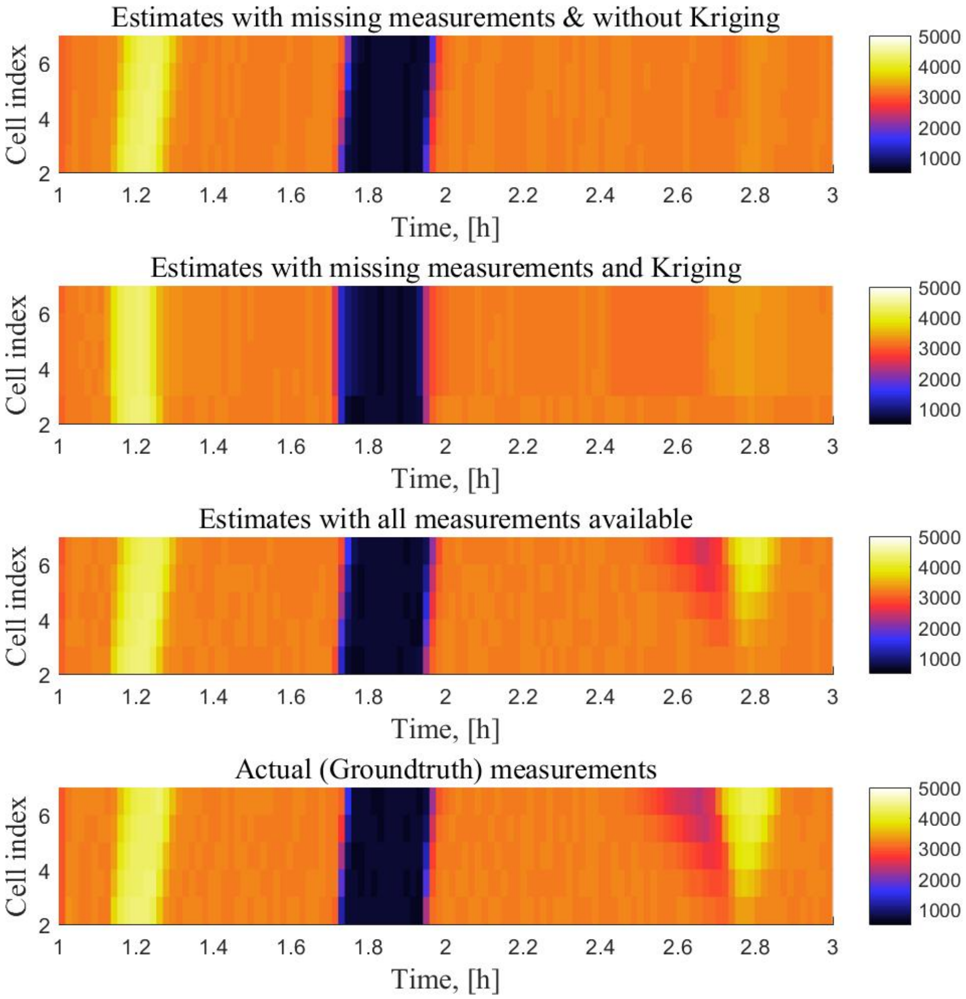

Figure 3.8 Traffic flow of segments 2 to 6

3.5.2 Investigation with Real Data

The modified algorithm was further tested with real data from the E-17 motorway in Belgium [2], which is usually congested. The test data consist of a day measurement recorded by sensors installed at locations CLOF to CLO9 as shown in Figure 3.10. Measurements at location CLOE to CLOB were removed and then interpolated using Kriging with the following parameters, free flow speed v_{free} = 120 km/h, minimum speed v_{min} = 7.4 km/h, critical density ρ_{crit} = 20.89 veh/km/lane, jam density ρ_{jam} = 180 veh/km and a $\beta = 0.5$.

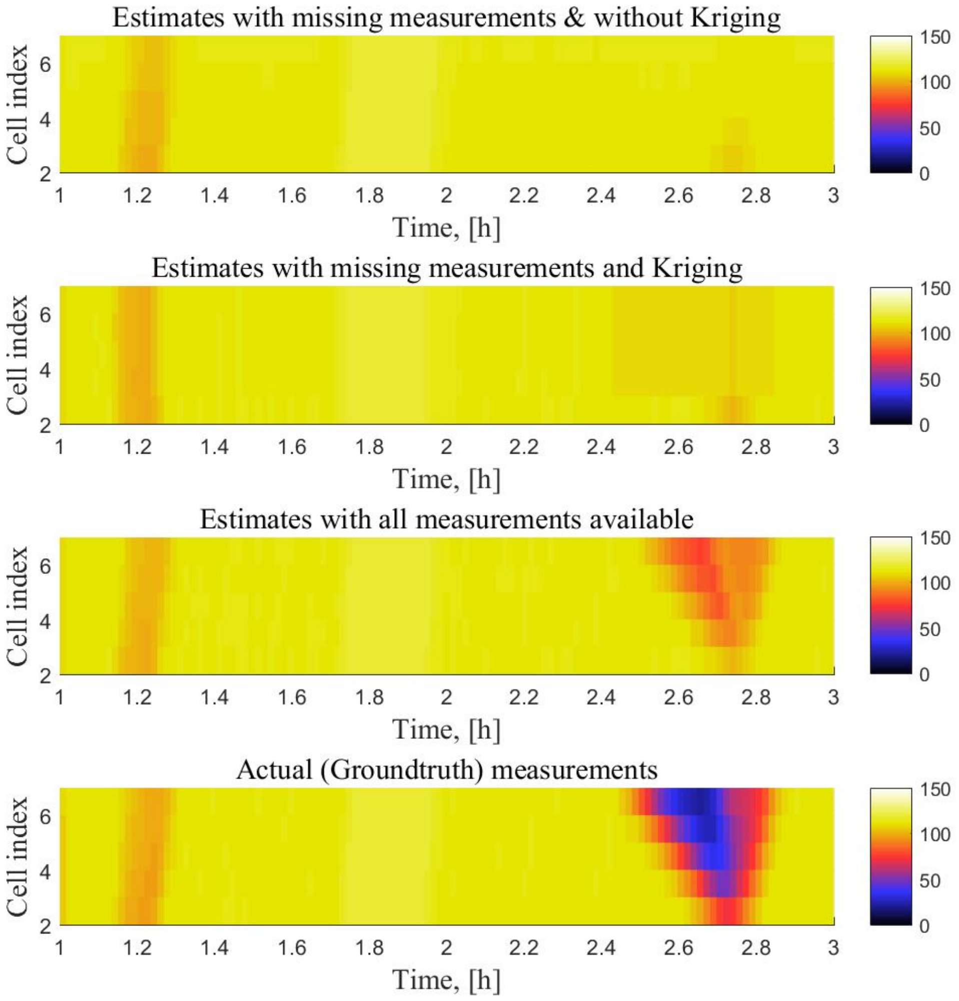

Figure 3.9 Speed of vehicles across segments 2 to 6

Figures 3.11 to 3.13 show the plot of the estimated traffic flows and the ground truth. The estimates follow the pattern of measured states at most of the points as seen from the plot. The speed-flow and flow-density diagrams are plotted in Figures 3.14 and 3.15. The shape of the figures resembles the fundamental diagram of traffic flow, confirming the validity of the approach.

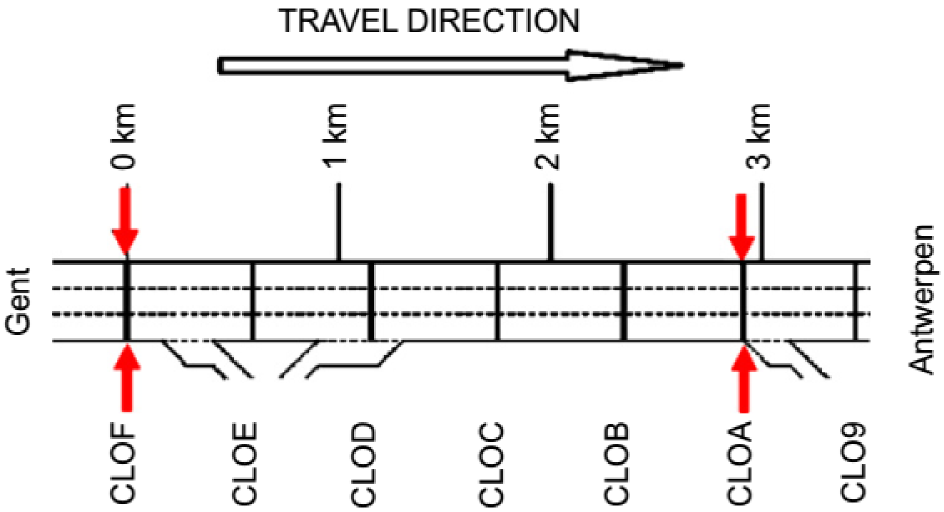

Figure 3.10 Schematic diagram of the E17 motorway between Ghent and Antwerp Kruibeke, Belgium [2]

3.5.3 Multi-Step Ahead Prediction Using Particle Filter

The particle filter estimates the state of traffic by taking a sufficient number of random samples from the posterior density function (pdf) with assigned weight to each particle. When a new measurement becomes available, it is used to compute what is known as likelihoods and a normalized form of the weights computed. The new state of the system is then updated with the computed weights. The importance weights tend to degenerate as the number of iterations increase. This problem is solved by re-sampling the weights. This involves replacing particles with low weights with a replica of those with high weights. The procedure is outlined below with a detailed explanation of each step.

3.5.4 Description of Experiment

As mentioned in section 3.5.3, the particle filter algorithm was used for the one-step-ahead prediction of the traffic states, namely (speed, flow and density). Figure 3.10 [2] shows the road network (the E17 motorway between Ghent and Antwerp in Belgium) used for the study. It was divided into six segments, labelled CLOF to CLO9. Measurements were received every minute at the boundaries. The research used particle filtering to estimate the traffic volume or flow, speed and density over the road network. For the purpose of estimating the traffic state, only measurements at CLOF and CLOA were used while the traffic states at CLOE to CLOB were estimated using the proposed algorithm. Traffic flow in all on-ramp and

Algorithm 4 PF Algorithm for Prediction

```
1. Initialization
   At $k = 0$; Define all boundary conditions: number of samples, weight of samples as below,
   For $i = 1..to..N_p$, $N_p$ number of particles;
      • generate $N_p$ samples from the initial distribution $p(\mathbf{x}_0)$
      • initialize the particle weights $w_0^{(i)} = \frac{1}{N_p}$
2. Start the Iteration for $k = 1, 2, ..., N_p$
   (a) Prediction Stage
       Predict the speed, flow and density using equations (3.6) to (3.14)
       For $i = 1, ..., N_p$, sample $\mathbf{x}_k^i \sim p(\mathbf{x}_k | \mathbf{x}_{k-1}^i)$
   (b) Use Measurements to Compute Likelihoods and update the weights
       This step is performed when the sampling time $t_s$ equals the iteration count $t_k$
       i. Compute the likelihoods
       Use equation 3.15 to compute the likelihood, $p(\mathbf{z}_s | \mathbf{x}_s^{(i)})$ of the particles
       ii. Update the weights of the particles using the likelihood
       $\boldsymbol{\omega}_s^{(i)} = \boldsymbol{\omega}_{s-1}^{(i)} p(\mathbf{z}_s | \mathbf{x}_s^{(i)})$
       iii. Normalize the weights:
       For i = 1 , .... $N_p$,
       $\hat{\boldsymbol{\omega}}_s^{(i)} = \frac{\boldsymbol{\omega}_s^{(i)}}{\sum_{i=1}^{N} \boldsymbol{\omega}_s^{(i)}}$,
       End For
   (c) Update the predicted values (Output): $\hat{\mathbf{x}}_s = \sum_{i=1}^{N} \hat{\boldsymbol{\omega}}_s^{(i)} \mathbf{x}_s^{(i)}$
   (d) Re-sample the weights (Selection) only when $t_k$ = $t_s$
       When some weights degenerates (become too low or too high) suppress or multiply the low/high weights to obtain N random samples approximately distributed according to $p(\mathbf{x}_s^i | \mathbf{Z}^s)$
   (e) M-step Ahead Prediction
       For prediction, based on the particles obtained in re-sampling step, evaluate the pdf of $\bar{\mathbf{x}}_{k+M+1|k+1}$ with the augmented state model $\bar{\mathbf{A}}\hat{\mathbf{x}}_k + \bar{\mathbf{B}}\hat{\mathbf{w}}_k$ where M is prediction step and $\bar{\mathbf{A}}$ and $\bar{\mathbf{B}}$ are the augmented state and noise matrix respectively
   (f) Increment the iteration count k, and return to step (1)
```

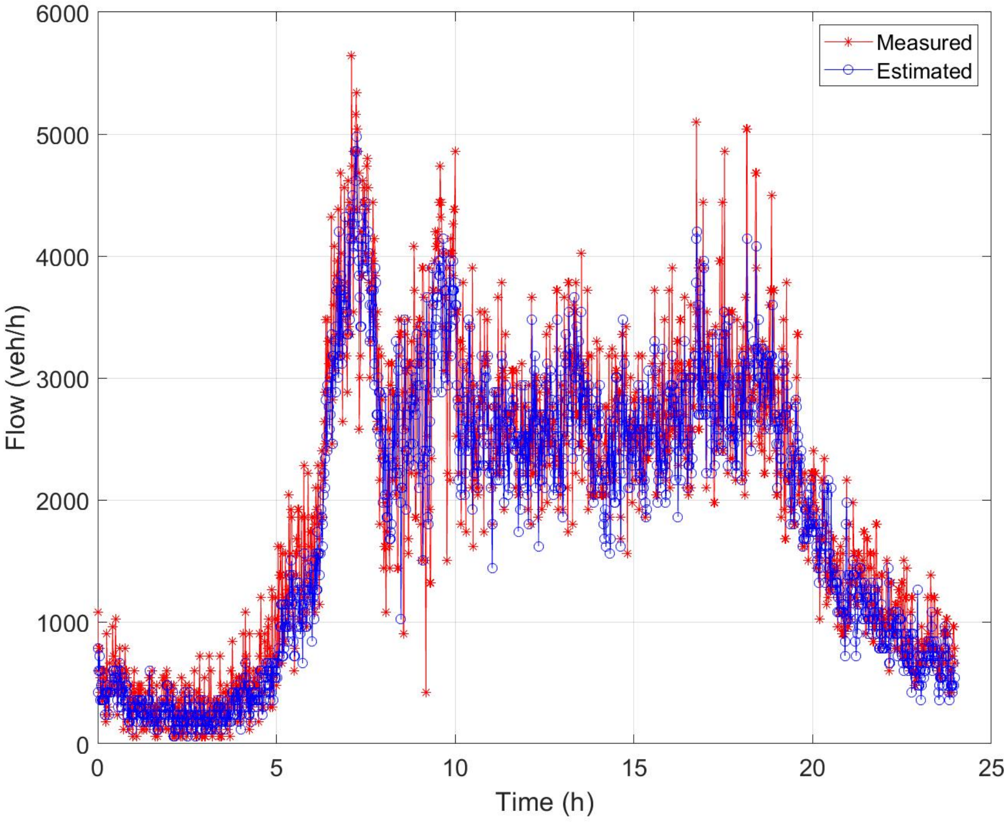

Figure 3.11 Predicted states and actual measured states at CLOC

off-ramps were assumed negligible in order to ensure the conservation of vehicle equation. In this report, the algorithm was modified to perform one-step-ahead prediction and the results compared with unscented Kalman predictor.

3.6 Results and Discussion

This section presents the result of simulation carried out using particle filter algorithm and the modified version of the cell transmission model (called stochastic compositional model, SCM) (see Algorithm 1 for details) [1].

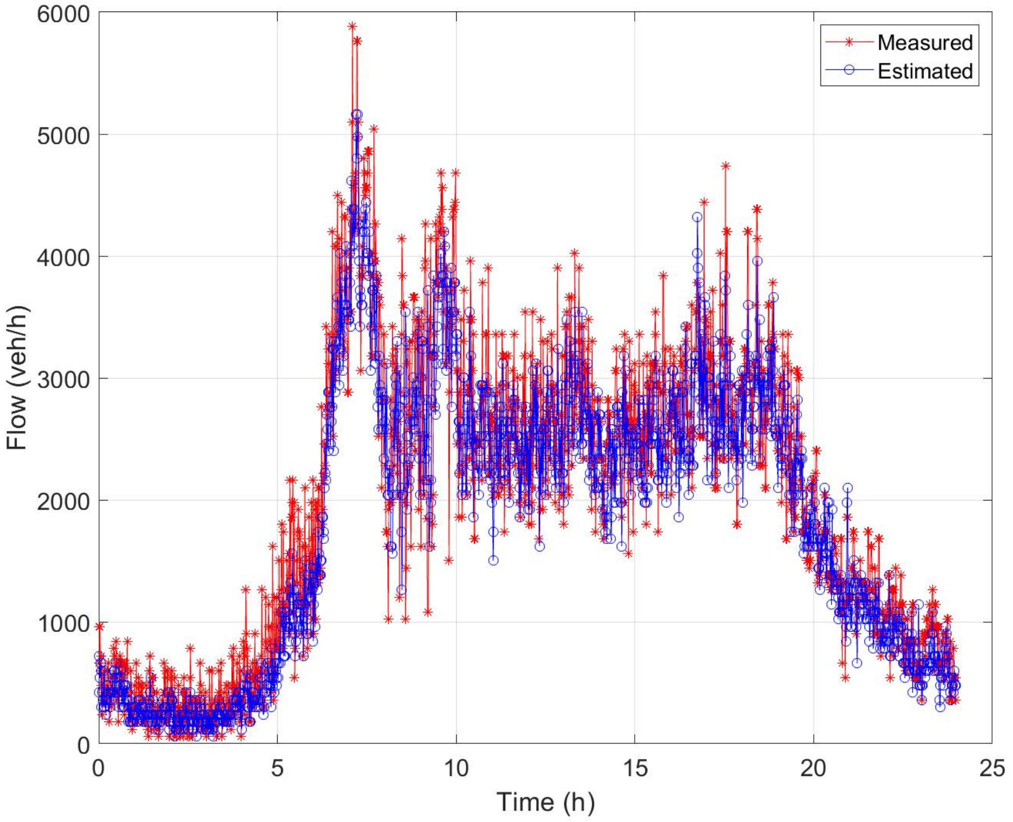

Figure 3.12 Predicted states and actual measured states at CLOD

3.6.1 Validation Using Root Mean Square Error

The results of the particle filter and unscented Kalman filter were compared for 1 step ahead prediction. The simulation was run from 5:40 am to 11:30 pm for 100 Monte Carlo times each and the mean of the errors plotted. Another observation from the simulation was that the number of times the simulation was run does not affect the prediction or estimation accuracy of the UKF whereas the accuracy of PF gets better by repeating the simulation several times and taking the mean. This could be attributed to the fact that the PF uses a stochastic model to sample the PDFs, whereas UKF uses a deterministic sigma point. The root mean squared (RMSE) error metric was used to evaluate the accuracy of the prediction.

Figures 3.16 and 3.17 shows the speed and flow RMSE for multiple steps ahead prediction of using modified particle filter algorithm. It is observed that the error increases as the prediction horizon increases.

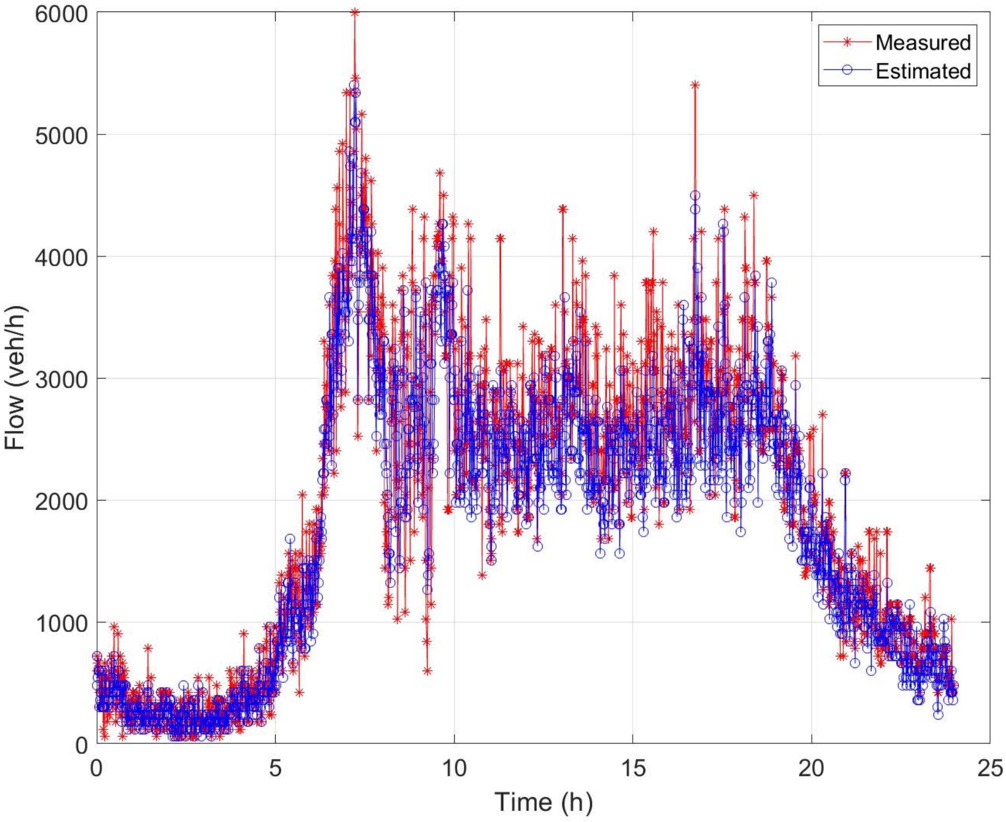

Figure 3.13 Predicted states and actual measured states at CLOE

3.6.2 Comparison with Existing Work

The method proposed in this section is compared with the work of [174] using the root mean squared error. The results for the traffic flow is shown in Table 3.1. In [174], compressive sensing is used to fill the missing data in the particle filter update step. This is abbreviated as PFCS (particle filter with compressive sensing). The RMSE of the traffic flow at cell boundaries CLOB to CLOE was computed, as shown in the table. It could be observed that the proposed method outperformed the compressive sensing approach.

Table 3.1 Minimum, maximum, mean and percentage improvement

	CLOB	CLOC	CLOD	CLOE
RMSE Proposed (veh/h)	98	101	107	96
RMSE PFCS (veh/h)	154	152	150	148

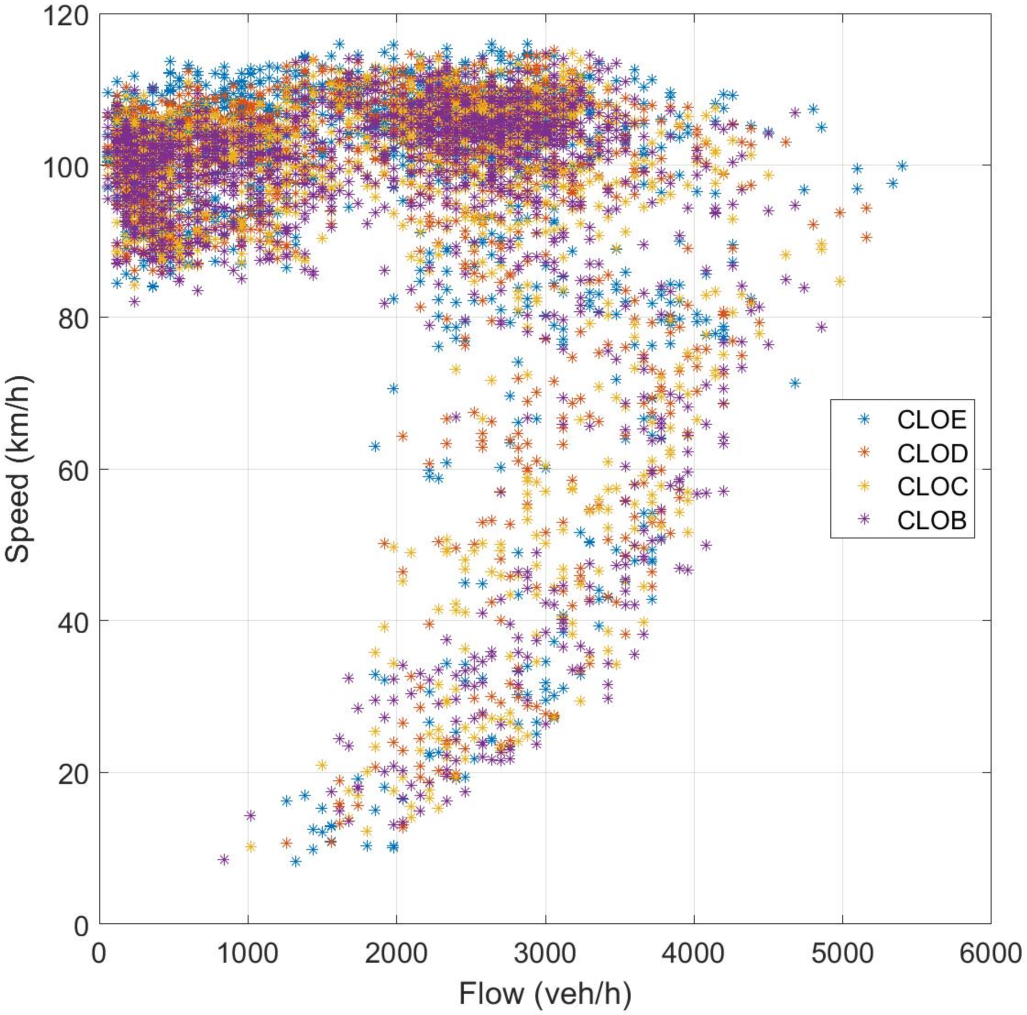

Figure 3.14 Speed-flow diagram for the PF with Kriging estimated measurements at CLOE, CLOD, CLOC and CLOB

3.7 Summary

This chapter proposed a novel approach to tackle the problem of missing and sparse data in traffic estimation. This approach entails interpolating the missing values using Kriging with a level of confidence assigned to the Kriged values by computing their interpolation error variance. This level of confidence is then used to compute the weight to be assigned during the computation of innovation terms used in PF. This was tested using simulated and real data by assigning fixed test-values to the weighting factor. From the results presented benefit of adjusting the weighting of interpolated values as compared to actual measurements has

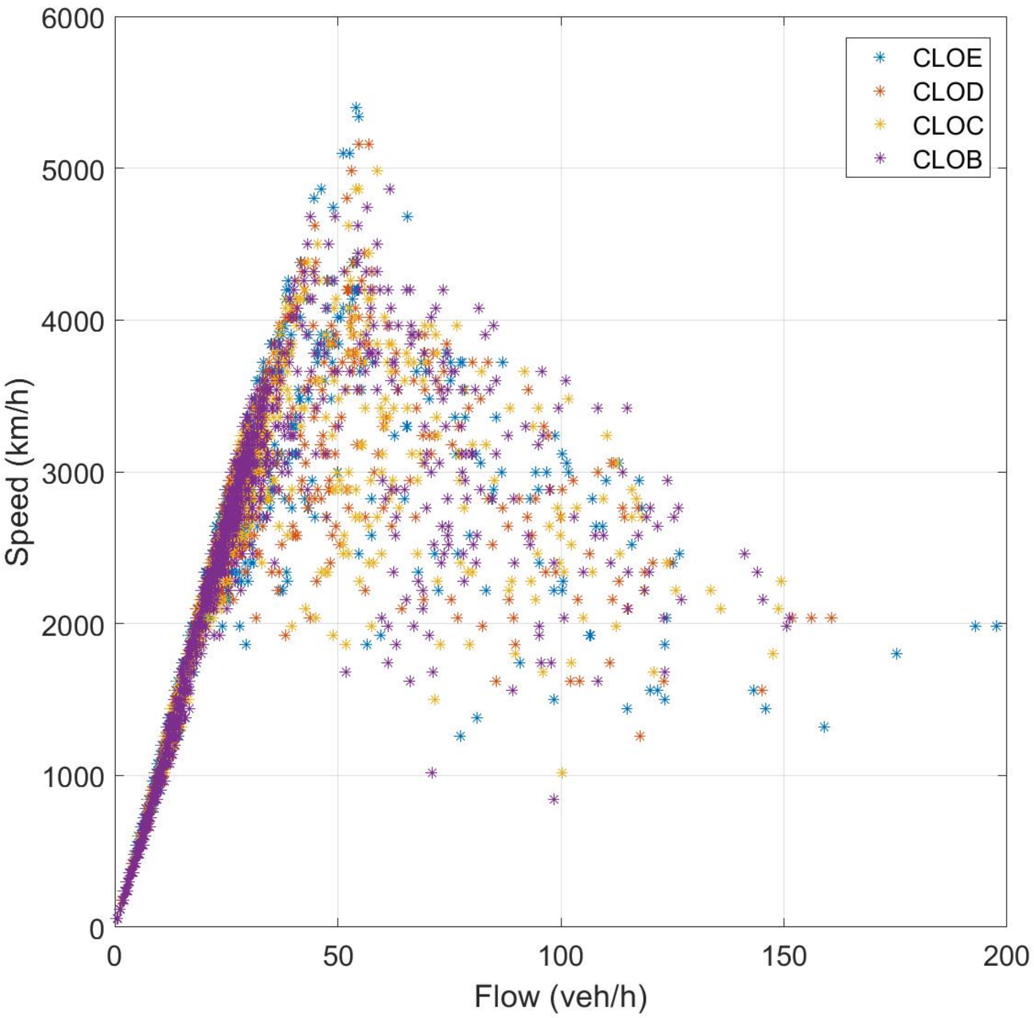

Figure 3.15 Flow-density diagram for the PF with Kriging estimated measurements at CLOE, CLOD, CLOC and CLOB

offered an improvement. In chapter 4, this will be extended to a large road network with varying levels of missing data.

The next chapter extends the approach presented in this chapter to a large-scale road network. It also considers a scenario where some lanes are closed as a result of incidence or road work resulting in congestion. An approach to tackle the effect of the congestion on a large-scale road network is presented and analysed.

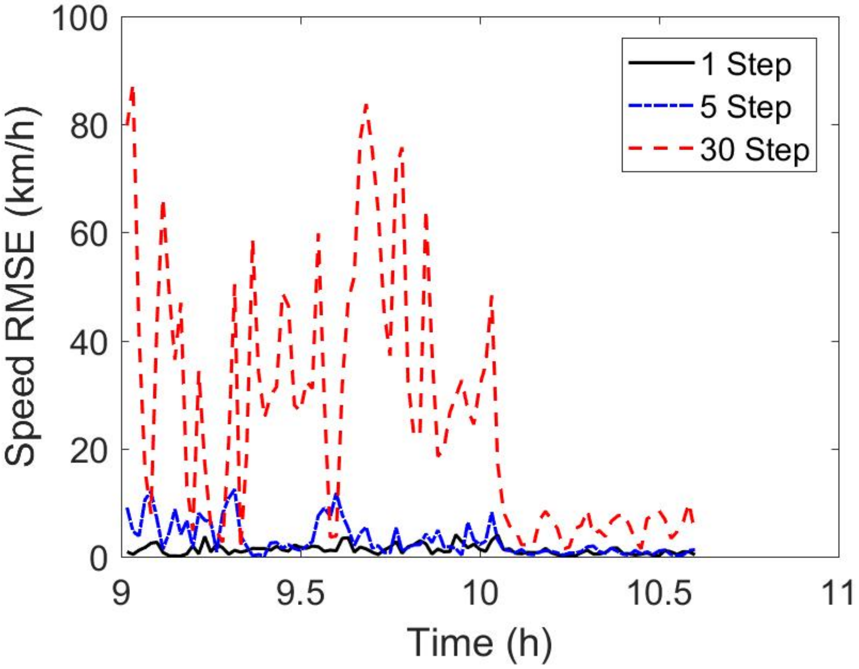

Figure 3.16 Plot showing RMSE of speed at Segment CLOA

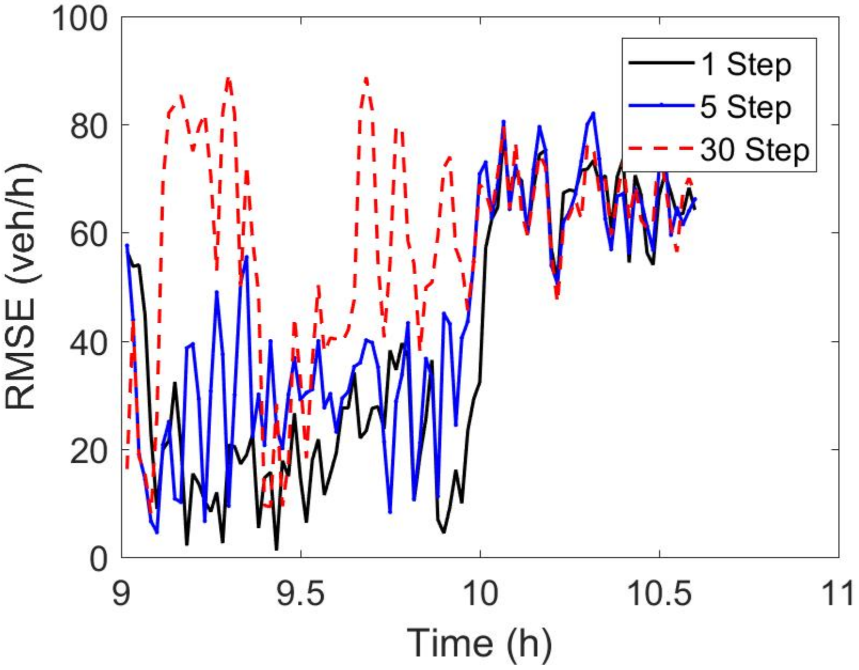

Figure 3.17 Plot showing RMSE of flow at Segment CLOF

Chapter 4

Traffic Estimation for Large Urban Road Network with High Missing Data Ratio

4.1 Introduction

Intelligent Transportation System (ITS) require accurate traffic state for effective traffic monitoring and control. Traffic states are usually estimated from noisy sensor measurements using various approaches, which can typically be subdivided into model-based approaches, data-driven approaches or a combination of both. An overview of the different modelling methodologies is given in [11, 12]. These modelling methodologies include microscopic, macroscopic and mesoscopic approaches. Microscopic traffic models [11–14] describe the motion of each individual vehicle with a high level of detail.

In macroscopic models [15, 16], traffic state is represented by aggregating behaviour of the traffic, usually in terms of the average speed and the average density over a given period. Mesoscopic models [11] employs some features of the microscopic and macroscopic approaches by utilising varying levels/degrees of detail to model traffic behaviour. This is achieved by modelling some locations with aggregated measurements as in macroscopic, and the remaining locations are modelled down to the details of individual vehicles as is done in the case of microscopic.

Macroscopic models are enough to produce acceptable estimation accuracy when compared to the computational overhead of the microscopic models. Hence, they are proffered choice for most practical purposes such as traffic control/management, road pricing and changes in infrastructure. Most traffic estimation approaches are model-based [17], while the new trend is to develop data-driven approaches [18].

Data-driven methods rely on historical data or streaming/real-time data. Within the last decade, there has been growing interest in applying Kriging for various traffic state prediction: directional traffic volume using global position system (GPS) data [19], annual average traffic count interpolation using origin-destination data [20], estimating annual average daily traffic [21, 22], traffic volume prediction [23, 24], and traffic volume imputation [25].

Kriging is one of the leading data-driven methods originally employed in geostatistics or spatio-temporal analysis. Kriging exploits the spatial correlation (either covariance or variogram) using a weighted sum of observed data points to interpolate the values at locations of interest. It was initially used for Copper mining by Krige [126] and was developed further by Matheron [127]. Since then, it has been applied in other fields such as spatial analysis, computer experiments. In recent times, the method has been applied in traffic prediction [19].

One of the major challenges faced in traffic prediction is the issue of missing or sparse data. Traffic measurements are generally captured with different types of sensors and transmitted through a communication infrastructure for processing and utilisation. These infrastructures are subject to failure and malfunction, occasionally leading to incomplete/missing data, sometimes more than 40% [10]. The problem of sparse data is caused by the high cost of installing and managing traffic measurement devices, making it impractical to cover all locations needed for effective observation of the full road network.

To address these challenges of missing/sparse data, researchers resorted to various methods and approaches such as missing data imputation [35], compressive sensing and historical averages [174], Kriging interpolation [23]. In [2], particle filter (PF) with the stochastic compositional model (SCM) as the proposal distribution was used to estimate traffic state in freeways/motorways. Boundary measurements (inflow and outflow) were used to estimate the traffic state within the segments. The study reported that missing boundary measurements affected the estimated accuracy. A solution to the problem of missing data in particle filter technique was proposed in [180] where Kriging methods were used to interpolate the missing data which is subsequently used for the computation of the PF likelihood for traffic state estimation. This approach was limited to a small road network and a few missing data.

This present work extends the approach of [180]. In the previous work, a small road network of 8 segments was considered whereas this present work considered a larger road network of several kilometres with 1000 segments under the influence of missing data and/or sensor failure. The task of training such large segments would be resource-intensive, hence the reduced measurement space proposed by [181] was used to select the most influential segments in the road network. The drawback of [181] is that the most influential segments are selected based on available measurements. This would mean that if some interconnected and more correlated segments have missing data, they won't be used leading to information loss. To address this, the most probable segments are first selected using the column-based matrix decomposition (CBMD). Then when there are missing measurements in the segments, Kriging is used to estimate the measurements before the measurement update step (see Section 4.3.3).

A review of three different missing data imputation methods was presented by [175]. These include interpolation, prediction and statistical learning. The interpolation method imputes missing measurement at the location by averaging all historical measurements at that location at similar times of the day. Prediction methods use a deterministic mathematical description to model the relationship between historical and future data. The statistical methods, on the other hand, treat the traffic as a random variable and tries to capture the stochastic nature of the traffic pattern into the imputation algorithm.

In [147], [153, 154], a multi-resolution approximation using linear combinations of basis functions was proposed to address computational complexity of large datasets. The novelty lies in the use of multiple basis functions computed at lower resolutions closer to observation locations and then combining them to capture the different covariance functions with varying properties. The division is achieved by dividing the spatial domain recursively into small regions and smaller sub-regions until the fine-scale dependencies are captured.

Nychka [147] used radial basis functions (RBF) and a special type of Gaussian Markov random field (GMRF) called spatial autoregressive (SAR) model to model the spatial correlation among the coefficients while Katzfuss [154] automatically determines the appropriate basis/covariance function. Whereas in [153], it was assumed that the sub-domains are independent, [154] assumed depended sub-domains and performed full-scale approximation. As the computations are done locally in parallel, it is possible to fuse multi-sensor data sources, in which case, the different covariance functions are used for each data source. Although both mentioned that the approach could be extended to non-stationary functions, it was not implemented, nor was there a derivation for such. The different methods of missing traffic data imputation considered small road network in the range of a few kilometres.

4.2 Model Description

Consider a general discrete time state space system of a form,

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + \mathbf{g}_k(\mathbf{v}_k) \quad \text{or} \quad p(\mathbf{x}_{k+1}|\mathbf{x}_k), \tag{4.1}$$

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{w}_k \quad \text{or} \quad p(\mathbf{z}_k|\mathbf{x}_k), \tag{4.2}$$

where $\mathbf{f}_k$ and $\mathbf{g}_k$ are nonlinear functions of the target state vector $\mathbf{x}_k$ and process noise $\mathbf{v}_k$, respectively. Variable $\mathbf{h}_k$ represents a nonlinear relationship between sensor output $\mathbf{z}_k$ and target state vector $\mathbf{x}_k$ affected by a measurement noise $\mathbf{w}_k$. Also $p(\mathbf{x}_{k+1}|\mathbf{x}_k)$ is the probability density function of the new state $\mathbf{x}_{k+1}$ given the previous state $\mathbf{x}_k$, and $p(\mathbf{z}_k|\mathbf{x}_k)$ is the likelihood function of the measurement $\mathbf{z}_k$ given the state $\mathbf{x}_k$.

4.2.1 Stochastic Compositional Traffic Flow Model

In this work, a stochastic compositional model (SCM) [1] is considered for modelling of a motorway/freeway vehicle traffic evolution expressed as $p(\mathbf{x}_{k+1}|\mathbf{x}_k)$ in (4.1). This cell-transition model incorporates traffic speed and uses forward and backward waves to describe the complex relationship between traffic behaviour, especially for a large road network. SCM utilises sending and receiving functions which model the stochastic nature of traffic state evolution. The vehicles that are able to leave a cell are represented by receiving functions while those that are allowed to enter a cell are determined by the receiving functions. The receiving functions are usually less than or equal to the sending functions to obey the law of conservation of vehicles.

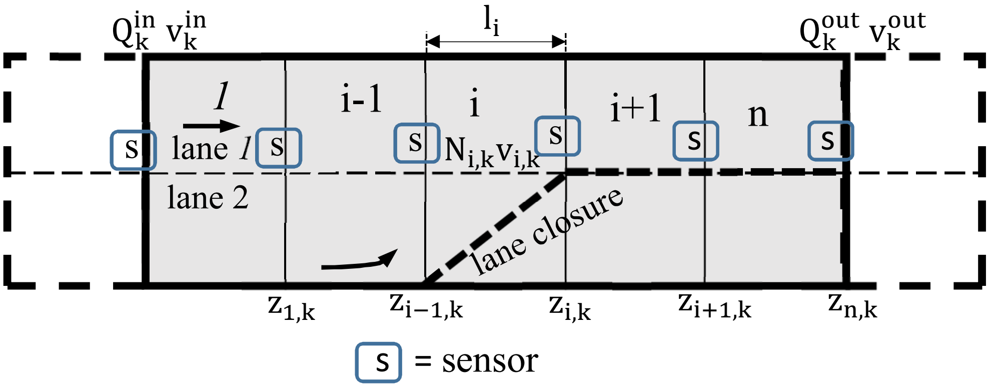

Figure 4.1 SCM road network showing segments and measurement points.

In SCM, the road network is divided into a given number of cell, n, also called segments. Each segment has a length L_i and number of lanes l_i as shown in Fig. 4.1, where $i = (1,\dots,n)$. At any given time period k, a certain number of vehicles $Q_{i,k}$ crosses the boundary between two segments i and $i+1$. The number of vehicles in a given cell i within the same time period k is represented by $N_{i,k}$ with their average speed given by $v_{i,k}$. Observe that in Fig. 4.1 there is a lane closure which differs from Fig. 3.1 without a lane closure.

The overall state vector at time k is given by $\mathbf{x}_k = [\mathbf{x}_{1,k}^T, \mathbf{x}_{2,k}^T, \dots, \mathbf{x}_{n,k}^T]^T$ where $\mathbf{x}_{i,k} = [N_{i,k}, v_{i,k}]^T$ is the local state vector at segment i. Equations (3.3) to (3.5) models traffic state evolution within the cells.

4.2.2 Traffic Measurement Model

Consider the road network shown in Fig. 4.1 divided into different segments with n boundaries. The traffic state at a boundary $j \in \mathscr{J} = 1,2,...,n$ is sampled at discrete time steps t_s, $s = 1,2,...$, to give $\mathbf{z}_{j,s} = (Q_{j,s}, v_{j,s})^T$. The measurements at all the boundaries are collected into a matrix given by $\mathbf{Z}_s = (\mathbf{z}_{1,s}^T, \mathbf{z}_{2,s}^T, ..., \mathbf{z}_{n,s}^T)^T$. The relationship between the sampling interval Δt_s and the state update time step Δt_k (4.1) is such that sampling interval is split into q state update time steps. That is, $\Delta t_s = q\Delta t_k$. The measurement model and noise is represented by equation (3.15) with error $\boldsymbol{\xi}_s = (\xi_{Q_{j,s}}, \xi_{v_{j,s}})^T$.

4.3 Recursive Bayesian Estimation

4.3.1 Bayesian Estimation

Consider a general discrete time state space system represented as in (4.1) and (4.2). The goal of Bayesian estimation is to infer the state variable $\mathbf{x}_k$ as defined in Section 4.2.1 with the available sensor measurements $\mathbf{z}_{1:k}$. By using the Bayesian framework, this estimation problem relates to the recursive evaluation of the probability density function (PDF) $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ in two consecutive steps, the prediction and the measurement update of the state vectors.

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \xrightarrow[\text{Update}]{\text{Prediction}} p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) \tag{4.3}$$

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) \xrightarrow[\text{Update}]{\text{Measurement}} p(\mathbf{x}_k|\mathbf{z}_{1:k}) \tag{4.4}$$

The prediction state density $p(\mathbf{x}_k|\mathbf{z}_{1:k-1})$ of state $\mathbf{x}_k$ is calculated from the prior PDF $p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$ by using Chapman-Kolmogorov equation

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})d\mathbf{x}_{k-1}. \tag{4.5}$$

Equality (4.5) follows the 1^{st} order Markov property which assumes that $p(\mathbf{x}_k|\mathbf{z}_{1:k-1})$ only depends on state $\mathbf{x}_k$ and $\mathbf{x}_{k-1}$ at time k and $k-1$ respectively. The measurement update $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ is computed from the prior distribution (4.5) and measurements $\mathbf{z}_{1:k}$ by a Bayesian formula which results in

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} \tag{4.6}$$

The 1^{st} order Markov property for equation (4.6) implies that $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ only depends on measurement $\mathbf{z}_k$ at time k.

4.3.2 Developed Particle Filter for Large Scale Road Nework

Arguably, the most popular algorithm for nonlinear recursive estimation is the particle filter (PF), extensively evaluated in [182]. PF represents any arbitrary probability density function $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ by samples or particles x_k^l, where $l = 1,\ldots,N_p$ is the number of particles. i.e.

$$x_k^l \approx p(\mathbf{x}_k|\mathbf{z}_{1:k}), \tag{4.7}$$

The particles are used to form an approximative distribution as

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \hat{p}(\mathbf{x}_k|\mathbf{z}_{1:k}) = \sum_{i=1}^{N_p} w_{k|k}^l \delta(\mathbf{x}_k - x_k^l), \tag{4.8}$$

where $\hat{p}(\mathbf{x}_k|\mathbf{z}_{1:k})$ is an approximated distribution, $\delta(\mathbf{x}_k - x_k^l)$ is a the Dirac delta function and $w_{k|k}^l$ the weights of the particles satisfying $\sum_{l=1}^{N_p} w_{k|k} = 1$. The time update of the Bayesian

Algorithm 5 Particle Filter for Traffic State Estimation with Kriging Estimated Measurements [2]

```
1. Road network approximation
   Use compressed sensing to select $m$ most significant locations out of the $n$ segments to
   be used for the measurement update step as defined in Section 4.3.3.

2. Initialization
   At $k = 0$; define all boundary conditions: number of samples, weight of samples as
   below,
   For $l = 1,...N_p$, $N_p$ number of particles;
     • generate $N_p$ samples $\{\mathbf{x}_0^{(l)}\}$ from the initial distribution $p(\mathbf{x}_0)$
     • initialize the particle weights $w_0^{(l)} = \frac{1}{N_p}$.
   End for

3. Start the iteration for $k = 1, 2, ...$
   (a) Prediction stage
       For $l = 1, ..., N_p$,
         sample $\mathbf{x}_k^{(l)} \sim p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(l)})$ according to SCM model equations
       End for
   (b) Measurement Update:
       This step is performed when the sampling time $t_s$ equals the iteration count $t_k$ as
       defined in Section 4.2.2
       i. Estimate missing measurements in the $m$ most significant locations with
       Kriging using equations (3.26).
       ii. Compute the likelihoods
       Use model (3.15) to compute the likelihood, $p(\mathbf{z}_s|\mathbf{x}_s^{(l)})$ of the particles
       iii. Update the weights of the particles using the likelihood $p(\mathbf{z}_s|\mathbf{x}_s^{(i)})$ calculated
       from model (3.15)
       For $l = 1, ..., N_p$
         $\omega_s^{(l)} = \omega_{s-1}^{(l)} p(\mathbf{z}_s|\mathbf{x}_s^{(l)})$
       End For
       iv. Normalize the weights: $\hat{\omega}_s^{(l)} = \frac{\omega_s^{(l)}}{\sum_{l=1}^{N_p} \omega_s^{(l)}}$.
   (c) Update the predicted states (Output): $\hat{\mathbf{x}}_s = \sum_{l=1}^{N_p} \hat{\omega}_s^{(l)} \mathbf{x}_s^{(l)}$
   (d) Re-sample the weights (Selection) only when $t_k = t_s$
```

recursion (4.1) is in case of PF evaluated as

$$\begin{aligned} &p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) \\ &\approx \int p(\mathbf{x}_k|\mathbf{x}_{k-1}) \sum_{l=1}^{N_p} w^l_{k-1|k-1} \delta(\mathbf{x}_{k-1} - x^l_{k-1}) \mathbf{dx}_{k-1}, \\ &\approx \sum_{l=1}^{N_p} w^l_{k-1|k-1} \int p(\mathbf{x}_k|\mathbf{x}_{k-1}) \delta(\mathbf{x}_{k-1} - x^l_{k-1}) \mathbf{dx}_{k-1}, \\ &\approx \sum_{l=1}^{N_p} w^l_{k-1|k-1} p(\mathbf{x}_k|\mathbf{x}_{k-1}). \end{aligned} \tag{4.9}$$

The particles x^l_{k-1} in above equations (4.9) are sampled from proposal distribution $\pi(\mathbf{x}_k|x^l_{k-1})$, i.e. $x^l_{k-1} \approx \pi(\mathbf{x}_k|x^l_{k-1})$. Proposal distribution is very often defined by the state transition PDF, that is $\pi(\mathbf{x}_k|x^l_{k-1}) = p(\mathbf{x}_k|x^i_{k-1})$. In this case, the weights updates result to

$$\begin{aligned} w^l_{k|k-1} &= \frac{p(\mathbf{x}_k|x^l_{k-1})}{\pi(\mathbf{x}_k|x^l_{k-1}, z_k)} w^l_{k-1|k-1}, \\ &= \frac{p(\mathbf{x}_k|x^l_{k-1})}{p(\mathbf{x}_k|x^l_{k-1})} w^l_{k-1|k-1} = w^l_{k-1|k-1}. \end{aligned} \tag{4.10}$$

The measurement update $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ (4.2) is computed by a Bayesian formula (4.6), which can be in terms of the particles x^l_k represented as

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{z}_{1:k}) &\propto p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{z}_{1:k-1}), \\ &\approx \sum_{l=1}^{N_p} w^l_{k|k-1} p(\mathbf{z}_k|\mathbf{x}_k) \delta(\mathbf{x}_k - x^l_k). \end{aligned} \tag{4.11}$$

Similarly, the particle filter weights are updated as

$$w^l_{k|k} = \frac{w^l_{k|k-1} p(\mathbf{z}_k|x^l_k)}{\sum_{l'=1}^{N_p} w^{l'}_{k|k-1} p(\mathbf{z}_k|x^{l'}_k)}. \tag{4.12}$$

Denominator in (4.6) and (4.12) is only a normalizing factor independent of $\mathbf{x}_k$ thus can be safety omitted if the distribution is numerically normed as shown by (4.11) and (4.12).

$$\begin{aligned} w^l_{k|k} &\propto w^l_{k|k-1} p(\mathbf{z}_k|x^l_k), \\ &\approx w^l_{k|k-1} p(\mathbf{z}_k|x^l_k). \end{aligned} \tag{4.13}$$

The MC recursion tends to degrade over time as all relative weights would tend to zero except for one that tends to one. Therefore, when particle depletion ratio reaches 0.5, a Sampling Importance Resampling (SIR) or Sampling Importance Sampling (SIS) techniques are applied in the recursion.

4.3.3 Column Based Matrix Decomposition and Improved Likelihood Computation

For a large road network with many segments, using all the measurements in the particle filter measurement update step becomes computationally intensive. Column based matrix decomposition approach similar (as earlier stated in Section 4.1) to the work of [181] is employed to select the most probable segments that would give acceptable accuracy.

The idea is to select m most influential segments from all available segments n using CBMD and then estimating missing measurements (if any) of the most influential segments using Kriging for improved particle likelihood computation. Let $\mathbf{Z}_n \in \mathbb{R}^{k\times n}$ represent a set of all segments or measurement locations in a given time period, where the rows, k is the number of time instances at which the measurements were taken and columns n number of road segments.

The goal of CBMD is to approximate the measurements $\mathbf{Z}_n$ with a subset of measurements $\mathbf{Z}_m \in \mathbb{R}^{k\times m}$ where $m < n$ is a subset of the measurements using singular value decomposition (SVD) as [183]:

$$\mathbf{Z}_m = \mathbf{Z}_n \mathbf{\Phi} \tag{4.14}$$

where $\mathbf{\Phi} \in \mathbb{R}^{n\times m}$ is the transformation matrix that expresses every column of all measurement $\mathbf{Z}_n$ in terms of the basis in $\mathbf{Z}_m$. Having computed the SVD, the right singular matrix is used to assign a probability P_{z_i} to each selected location according to:

$$P_{z_i} = \frac{1}{r}\sum_{j=1}^{r} v_{i,j}^2, i = 1, ..., n \tag{4.15}$$

where $v_{i,j}$ is the i^{th} element of the j^{th} right singular vector and r is the rank of the matrix. From the probabilities computed, m locations with the highest probabilities are chosen as the reduced measurement to approximate the entire network which is used in computing the particle filter likelihood during the measurement update step.

The likelihood function term $p(\mathbf{z}_k|\mathbf{x}_k)$ in equation (4.12), is computed when a new measurement arrives. The performance of the PF degrades substantially when there is a missing measurement. For the multivariate Gaussian distribution, the PDF is given by:

$$p(\mathbf{z}_k|\mathbf{x}_k) = \frac{1}{\sqrt{2\pi|\mathbf{R}|}} e^{-0.5\boldsymbol{\upsilon}\mathbf{R}^{-1}\boldsymbol{\upsilon}^T}, \tag{4.16}$$

where $\mathbf{R}$ is the covariance matrix of the measurement data, $|\mathbf{R}| \equiv det(\mathbf{R})$ is the determinant of $\mathbf{R}$ and $\boldsymbol{\upsilon}$ is the difference between the PF predicted value ($\bar{\mathbf{z}}_s$) and measurement ($\mathbf{z}_s$), given by:

$$\boldsymbol{\upsilon} = \mathbf{z}_s - \bar{\mathbf{z}}_s. \tag{4.17}$$

The measurement matrix $\mathbf{z}_s$ can be expressed as,

$$\mathbf{z}_s = \begin{cases} \mathbf{z}_s^{meas} & \text{measurement from sensor;} \\ \hat{\mathbf{z}}_s^{krig} & \text{estimated by Kriging.} \end{cases} \tag{4.18}$$

where $\hat{\mathbf{z}}_s^{krig}$ represents the value estimated by Kriging (when measurement is not available). Note that the sampling time index t_s is split into q update time indices t_k as mentioned in Section 4.2.2. The measurement update state of the PF is performed only when $t_k \equiv t_s$. The modified particle method is presented in Algorithm 1.

4.4 Performance Evaluation

A road network with 1000 segments was simulated using SUMO software [184] to validate the proposed method. The segments are spaced 0.5km apart and measurements (number of vehicles crossing each segment boundary with their average speed) taken every second. Traffic signs were installed at some locations to model the effect of congestion.

This is an extension of the previous work [180] where a smaller number of segments was considered. The aggregate traffic flow and speed were sampled every 60seconds and the results collected over a period of 10800 seconds (3hrs). Two types of vehicles, bus and passenger car was defined with the parameters as in Table 4.1. The vehicles were added randomly into the network through the inflow boundary every one second and they travel through the network until they get to the last boundary when they leave the network. As a vehicle crosses each induction loop, it is counted with its speed. The average speed of the vehicles arriving at an induction loop over a period is recorded as the average speed.

Table 4.1 SUMO simulation parameters

	Car	Bus
Max speed	25 m/s	20 m/s
Acceleration	1.0 m/s2	0.8 m/s2
Deceleration	4.5 m/s2	4.5 m/s2
Sigma (driver perfection)	0.5	0.5
Length	5 m	10 m
Minimum Separation	2.5 m	3 m

The entire statistics, flow, occupancy, and speed are collected in an output file for further processing.

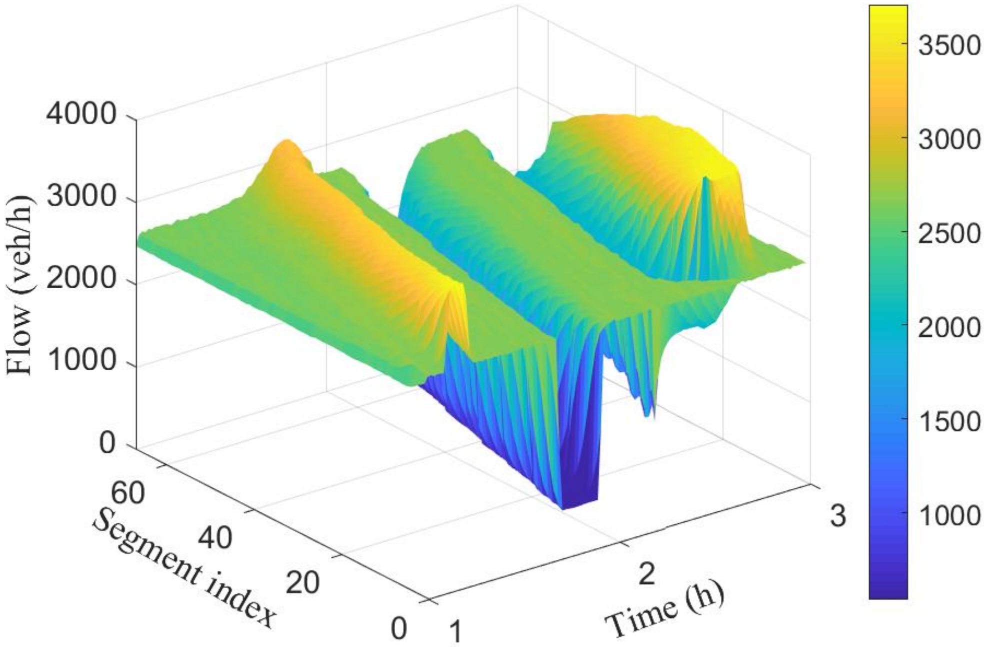

Figure 4.2 Spatio-temporal evolution of traffic flow for the 100 segments.

4.4.1 Simulation Design

To simulate different scenarios such as congestion and free flow, (i) the number of lanes were decreased from 3 to 2, and (ii) the rate of vehicle injection into the network is varied at, different time periods. Figures 4.2 and 4.3 show the spatio-temporal evolution of the traffic and their corresponding average speed, respectively, for a 100-segment section. The average speed of the vehicles varies around 100km/h when the flow was around 2000veh/h. Between time interval [1.5h, 1.7h], the flow was increased slightly to cause congestion, this resulted to a decrease in the average speed as can be seen in the first spike from Fig. 4.3.

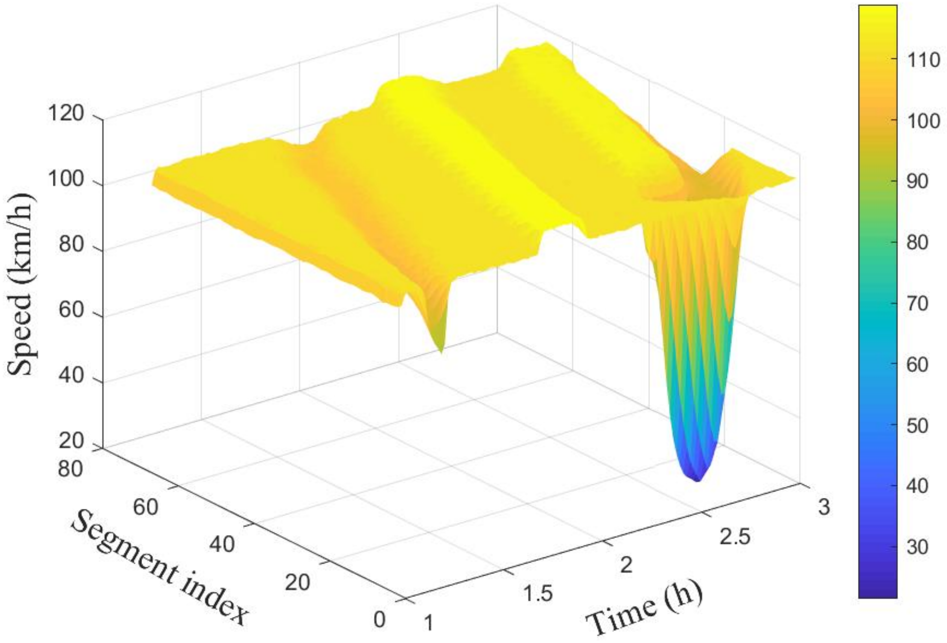

Figure 4.3 Spatio-temporal evolution of traffic speed for the 100 segments.

Observe that the effect was felt closer to the inflow boundary. The vehicles speed increases marginally as they move into the network. Between time interval [1.6h, 1.9h] the flow was decreased, leading to an increase in the average speed. Finally, the number of lanes in segments 10 to 14 were reduced from 3 to 2 between time interval [2.4h, 2.4h] while maintaining vehicle injection rate. This results in a substantive decrease in speed as can be seen (cone-shaped) in Fig. 4.3. As the vehicles pass the segments with closed lanes, there is an increase in their speed again.

4.4.2 Results and Discussion

In order to test the prediction accuracy for different levels of sparsity, a statistical measure, namely the root mean squared error (RMSE) (2.9), was computed. Note, z_i is the ground truth or actual measurement, $\hat{z}_i$ is the estimated value and m_r is the number of independent Monte Carlo runs. The measurements at some boundaries were randomly removed each time and then estimated using the proposed method. Different missing data rates (from 10, 20, ..., 70%) were investigated by randomly removing measurements at some locations using leave one out cross-validation. This was repeated for 100 Monte Carlo runs and the average value used.

Figures 4.4 and 4.5 shows the average estimation error for the different missing data ratios and number of segments. It would be observed that the prediction error increases with the missing data rate. This is expected as less data is available for the computation. This effect could be reduced further by incorporating a mechanism known as multi resolution approach [154].

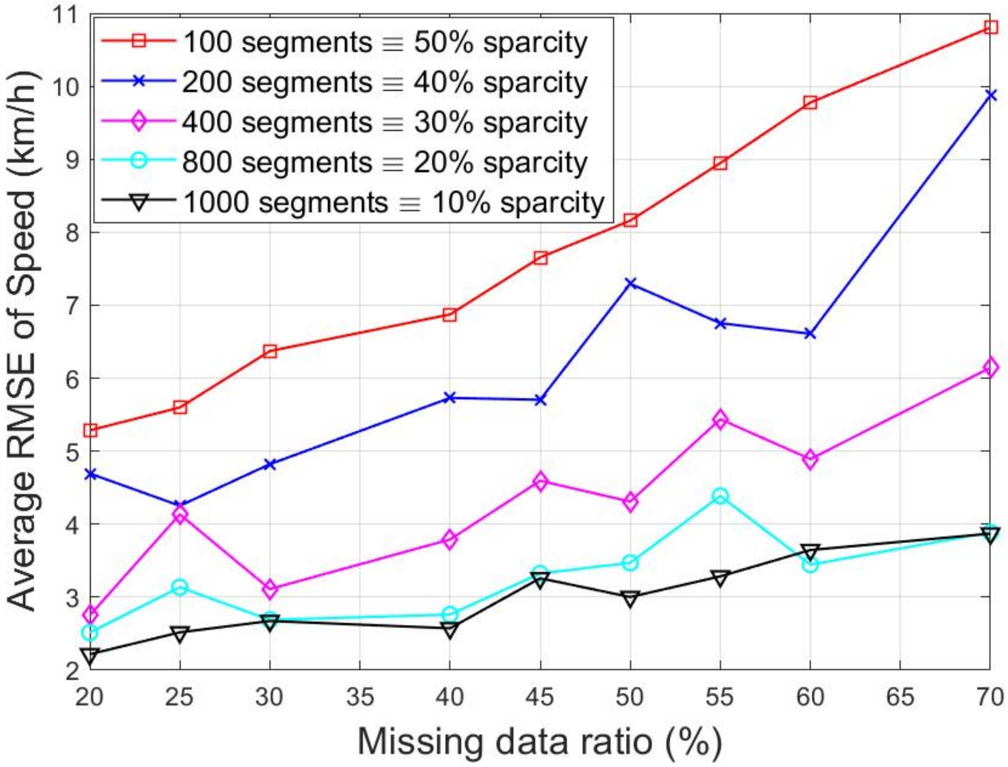

Figure 4.4 RMSE of speed at different missing data ratios.

The plot also shows that the higher the number of segments used, the better the accuracy. This could be attributed to the fact that there is better information exchange within the network and hence, on average, more segments with available data are used for the higher dimensional segments' scenario. This agrees with [160] where estimation accuracy in the presence of

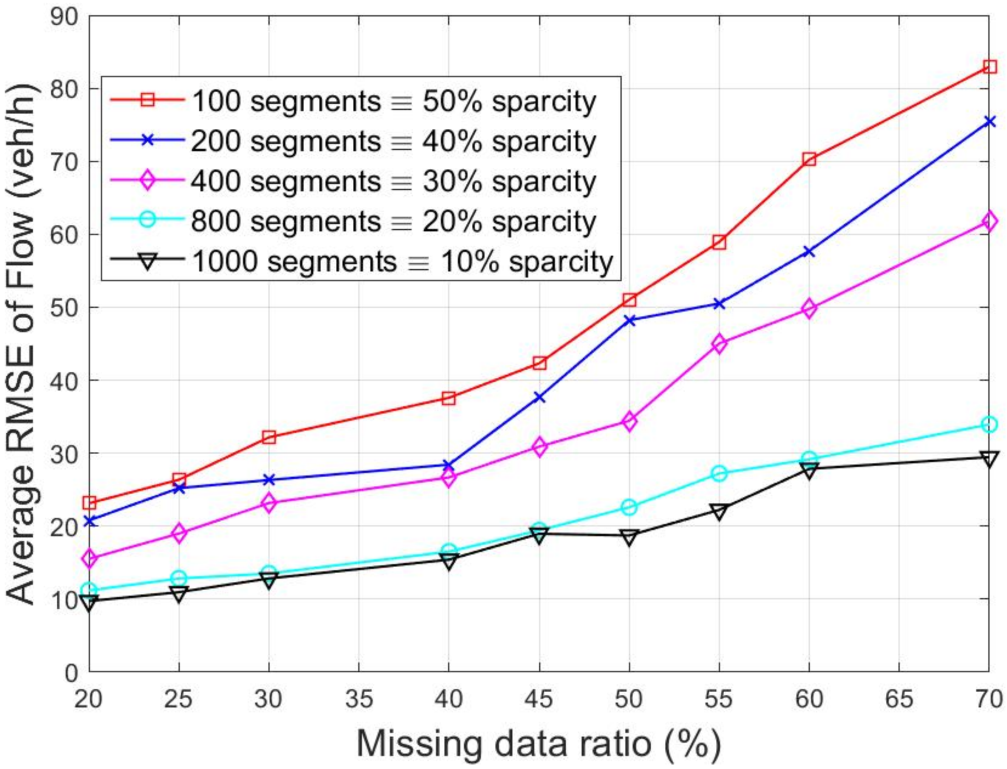

Figure 4.5 RMSE of flow at different missing data ratios.

sparse sensor data was improved by exchanging particle weights between segments. For instance, a 10-segment network with 70% missing data ratio will result in using only 3 data points to estimate the remaining seven missing locations. The chances of these 3 locations correlating with the other 7 are lower compared to when 30 out of 100 data points are available.

Figures 4.6 and 4.7 show the spatio-temporal evolution of the traffic flow and the corresponding average speed for the 100-segment scenario. The number of vehicles crossing each boundary in space and their associated average speed is represented by the colour bar and the z-axis.

Compared to the ground truth shown in Figures 4.2 and 4.3, it is evident that the estimated number of vehicles crossing segment boundaries and the associated speeds have been estimated with good accuracy. Observe also that the estimated flow and speed captured the periods where there is a drop in the number of vehicles and a decrease in speed.

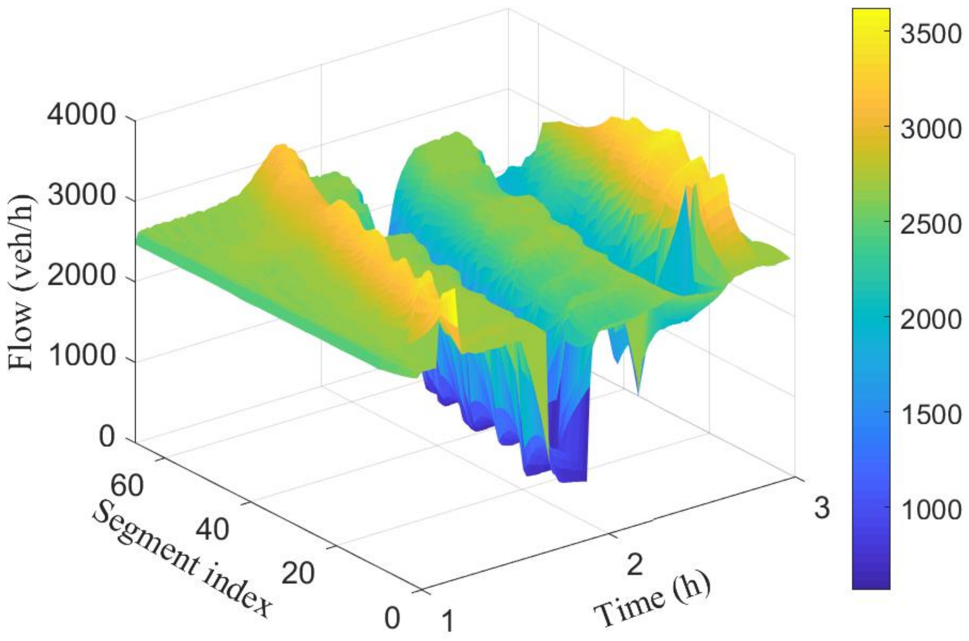

Figure 4.6 Estimated flow for the 100 segments with 30% missing data.

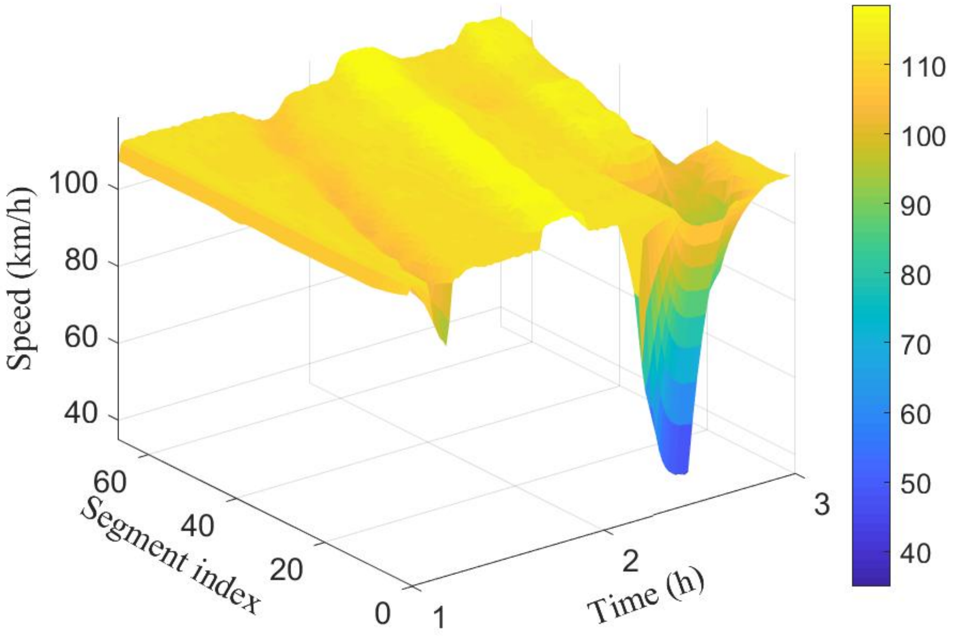

Figure 4.7 Estimated speed for the 100 segments with 30% missing data.

The box plot of the flow and speed is shown in Figs. 4.8 and 4.9, respectively. The figures show that the absolute error of estimated speed is in the range of 0 to 4km/h with some outliers. This is comparable to what is obtained in the literature.

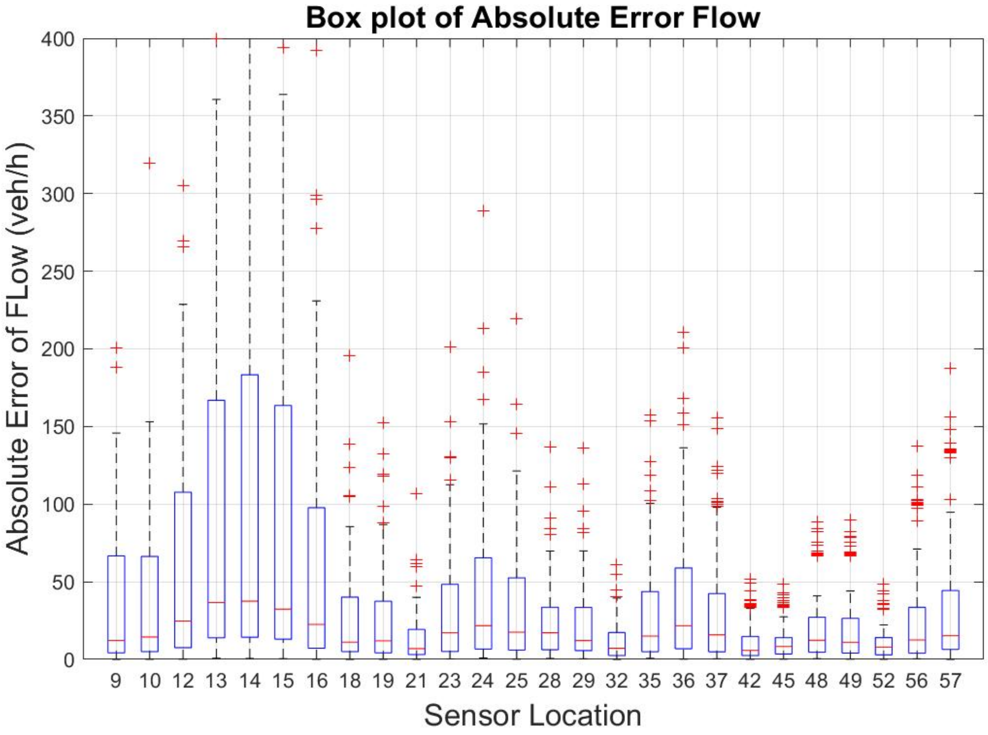

Figure 4.8 Box Plot of Flow over locations.

4.5 Summary

This chapter presented a traffic estimation for a large road network with different missing data ratios. The computational overhead of the large network was addressed by using a method called reduced measurement space proposed by [181] to select the most influential and information-rich segments in the road network. These are subsequently used in the particle filter measurement update step. Missing data in the selected segments are imputed using Kriging. A 1000-segment road network was simulated using SUMO. Different missing data ratios ranging from 10% to 70% were tested for different sizes of road network ranging from 100 to 1000 segments.

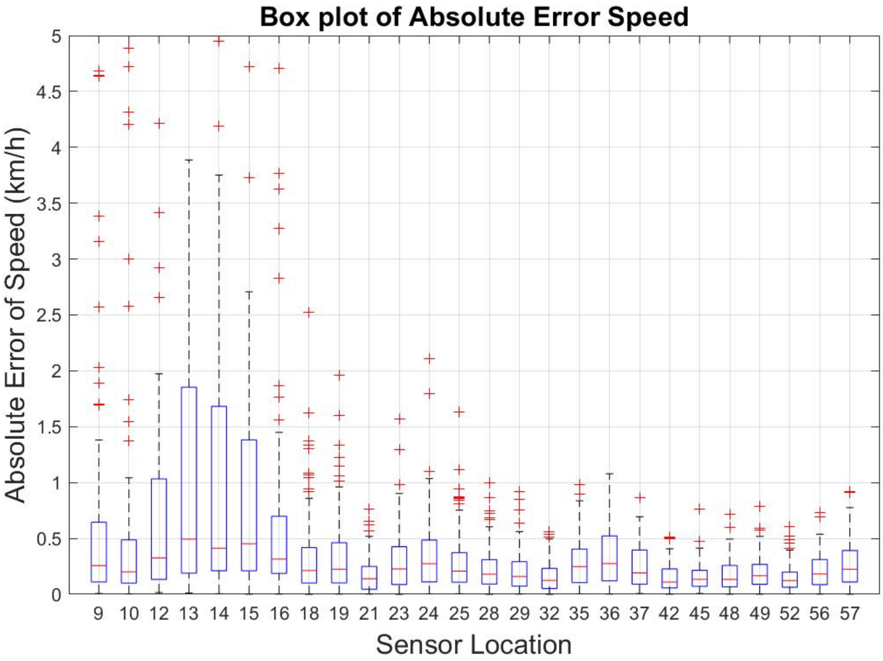

Figure 4.9 Box Plot of speed over locations.

The results indicate that considering a larger number of segments would reduce the overall estimation error even when the missing data ration is high. From the foregoing results and discussion, it is recommended that the best estimation accuracy would be obtained when the entire road network is considered at once. The effects of computational overhead could further be reduced by using a distributed approach with a central control unit.

What happens when road incidents result in lane closures and or increased demand beyond road capacity leading to congestion? A multi-model Bayesian Kriging approach is developed in the following Chapter 5, to capture the traffic dynamics, especially the effects of congestion caused by lane closures or excessive demand above road capacity.

Chapter 5

Bayesian Kriging with Local Covariance Functions for Urban Traffic State Prediction

5.1 Introduction

Intelligent Transportation System (ITS) requires knowledge of past, present and future traffic states for effective traffic monitoring and control. The past traffic state is readily available via measurements made with inductance loops, magnetic loops, video cameras, and social media data sources. The present and future state could be estimated and predicted respectively using past measurements and some computer simulation and traffic modelling techniques.

Kriging has been applied in the field of traffic engineering for state estimation [128, 19, 21, 23, 180], missing data imputation [185, 175, 186] and crash detection [129]. Braxmeier et al. [128, 19] employed Kriging with a moving neighbourhood to estimate the spatial location of traffic flows using data collected by vehicles equipped with Global Positioning System (GPS) signal receivers. Manepalli and Bham [129] compared Kriging and empirical Bayes in estimating road crashes and reported that Kriging performed better when the prediction term is less than three years. Wang and Kockelman [21] applied Ordinary Kriging with exponential model function to estimate the annual average daily traffic and reported an error of 31%. In [180] Kriging method and particle filtering were employed for traffic state estimation. Missing values were interpolated using Kriging, and the results were used in the weighting of the particle filter likelihood.

The Kriging method can also handle spatio-temporal prediction. For instance, [130] employed the modified Taylor Kriging to predict wind speed and reported an increment of 18.6% in performance compared to ARIMA. Other authors [132–134] have applied a modified version of Kriging to achieve spatio-temporal forecasting. A similar approach to [180] was applied in [187] to a large-scale road network with a high missing data ratio. In the work, column-based matrix decomposition (CBMD) was employed to select a subset of the segments that are more influential in improving computational overhead of processing large networks.

Most of the Kriging methods for traffic estimation assume a stationary covariance function with constant mean. In practice, traffic data is heteroscedastic and anisotropic in nature. Modelling the covariance function as constant over the entire region can produce local bias [150]. In [21] and [155], it was shown that the local estimation errors by using the ordinary Kriging weights are affected by data values. Interpolation variance was used in [156] to correct the smoothing effect of the ordinary Kriging variance. A modified-nugget effect was proposed in [155] to account for location-dependent non-constant variances.

The classical Kriging approach adopts a generative model of the covariance function to estimate the weights. This approach is not dependent on the data or site but on the joint probability of the locations. Bayesian Local Kriging (BLK) as proposed in [150] used a discriminative model of the covariance function, which is conditioned on the data to estimate the weights. This approach works well for both stationary and non-stationary systems or observations. Their model addresses two assumptions usually made in Kriging: stationarity and homoscedasticity. The first problem was addressed by using a discriminative covariance function conditioned on the data and the second by using a set of L local covariance functions. They considered a case where the regression part of the function is the same for all L local models.

Traffic data is periodic and seasonal with inter-day and intra-day dependence. Using fixed, spatial dependent covariance functions for prediction results in wrong results at some point and inaccurate at another. A spatio-temporal algorithm that is adaptive to the seasonal variations during days of the week, time and location are proposed. It is based on the approaches proposed in [150], [188, 145, 189, 190] and [191]. This work builds on the existing methods by employing a discriminative covariance model conditioned on the observations at each location. Thus, the proposed method can account for congested regions and interactions in the upstream and downstream of the congestion. Usually, the covariance function is only dependent on the separation distance irrespective of the traffic situation at the locations. This makes it difficult for the model to capture traffic dynamics and transitions from free-flow to congested states, congested states to free-flow, etc. This proposed approach can capture these dynamics and model them into the covariance matrix.

The contribution of this chapter is threefold: Traffic estimation using discriminative covariance functions conditioned on the data at each location which can capture the stochastic nature of the traffic; Robust and adaptive to both stationary and non-stationary traffic data; and Multi-model Gaussian traffic data analysis. Generally, a given dataset could be represented by different models. Traditional Kriging makes use of the “best” model that explains the whole dataset. This often leads to over-fitting or under-fitting within different scenarios. Using a weighted average of all the models has been shown to outperform a single model [192].

The rest of this chapter is structured as follows. Section 5.2 gives a brief overview of the related works. Section 5.3 presents the formulation of the problem of interest. Section 5.4 presents the experimental setup and performance evaluation. Finally, Section 5.5 concludes this work.

5.2 Related Work

In this section, the state-of-the-art approaches in traffic estimation are highlighted to put the current work in context. Braxmeier [128] takes into consideration the inhomogeneous and anisotropic nature of time-series data in estimating road traffic parameters. The method considered a rush-hour data set of only 30 minutes duration. The use of Euclidean distance in computing separation between locations fails to describe the spatial distance in a road network accurately. Zou [23] introduced road network distance called approximate road network distance (ARND), based on the isometric embedding theory, to describe the spatial distance between road links. The method addresses the problem of invalid spatial covariance function in Kriging caused by the non-Euclidean distance metric.

The Kriging weight is only a function of the distance between the points. It does not incorporate the heteroscedastic/stochastic nature of the data. In [21] and [155], it was shown that the local accuracy of estimation using ordinary Kriging weights is affected by data values. The use of interpolation variance was used in [156] to correct the smoothing effect of the ordinary Kriging variance. A modified-nugget effect was proposed in [155] to account for location-dependent non-constant variances.

In [193] a similar approach called vicinity Gaussian process was adopted. The vicinity sensor data of the local sensor are processed to make traffic flow prediction when the local sensor malfunctions or communication fails. First, a weighted directed graph of the network is built up. Next, a dissimilarity matrix is derived and accounts for the selection of training subsets. A more similar approach was proposed in [194] where non-negative matrix factorisation (NMF) was used to cluster speed data into related local clusters. Spatio-temporal speed estimation is then carried out by using the GP models of the most probable correlated cluster.

The proposed approach differs from these methods in the following aspects. Firstly, whereas [193] and [194] use a deterministic model, the proposed method is Bayesian and uses a discriminative covariance function with recursively updated likelihood when new data becomes available. This is particularly useful in traffic estimation, which is dynamic and stochastic in nature. Thus the proposed approach could be applied in diverse road network situations. Secondly, this approach combines Bayesian Model Averaging, which assigns different weight scales to each of the natural clusters identified using NMF. This is particularly beneficial because road segments are correlated, and clusters mutually affect each other.

5.3 Model Formulation

Kriging [126] is an estimation technique which relies on exploiting the spatial correlation of the observations $z(\mathbf{r}_i)$, $i = 1, ..., m$ at the locations $\mathbf{r}_i \in R$ to predict or interpolate the values at unobserved location r_u. Let $\mathbf{z}_m = \mathbf{z}(\mathbf{r}) = [z(\mathbf{r}_1), z(\mathbf{r}_2), \ldots, z(\mathbf{r}_m)]^T$, with $(.)^T$ the transpose operation, be a non-stationary set of m measurements, in the proposed approach either the average vehicle speeds $\mathbf{v}(\mathbf{r})$ or vehicle counts $\mathbf{N}(\mathbf{r})$, observed in segments $i = 1, \ldots, m$ at the locations $\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_m)$, where $\mathbf{r}_i = [r_i^x, r_i^y]^T$. Note that the location of each sensor is uniquely defined by the segment topology. Then, the random set of variables $\mathbf{z}(\mathbf{r})$ can be approximated by a Gaussian process

$$\mathbf{z}(\mathbf{r}) \sim \mathcal{N}\big(\boldsymbol{\mu}(\mathbf{r}), \mathbf{C}(\mathbf{r})\big), \tag{5.1}$$

which is uniquely defined by the mean $\boldsymbol{\mu}(\mathbf{r}) = E[\mathbf{z}(\mathbf{r})]$, and the corresponding covariance $\mathbf{C}(\mathbf{r}^T, \mathbf{r}) = \big(\mathbf{z}(\mathbf{r}) - \boldsymbol{\mu}(\mathbf{r})\big)^T \big(\mathbf{z}(\mathbf{r}) - \boldsymbol{\mu}(\mathbf{r})\big)$, where $E(.)$ is the mathematical expectation operation. The process under study could be modelled as a second order random field of the form:

$$\mathbf{z}(\mathbf{r}) = f(\cdot)^T \boldsymbol{\mu} + w(\mathbf{r}) + \varepsilon(\mathbf{r}), \tag{5.2}$$

where $f(\cdot)$ is the regression function, $\boldsymbol{\mu}$ is the vector of the regression coefficients, $w(\mathbf{r})$ is a spatial process modelling the spatial correlation, and $\varepsilon(\mathbf{r})$ is an independent process measurement noise, which is also known as the nugget-effect (see, e.g., [195]). The process $w(\cdot)$ is mostly specified as a zero-mean GP with covariance function $\mathbf{C}(\mathbf{r},\mathbf{r}^{'}) = \sigma^2\rho(\mathbf{r},\mathbf{r}^{'};\theta)$ where σ^2 is the initial process variance, $\rho(\cdot,\cdot;\theta)$ is a correlation function and θ is a vector of the correlation parameters.

5.3.1 Traditional Kriging

The underlying idea of Kriging is to obtain the response $z(\mathbf{r}_u)$, interpreted as a random variable positioned at the location $\mathbf{r}_u$, by a weighted linear combination of the observations $\mathbf{z}(\mathbf{r}) = [z(\mathbf{r}_1), z(\mathbf{r}_2), \ldots, z(\mathbf{r}_m)]^T$, based on a covariance model of the spatial dependence between the observed and interpolation locations using (5.1), i.e. observations $z(\mathbf{r}_i)$, $i = 1,\ldots,m$ at locations $\mathbf{r}_i$.

$$\hat{z}(\mathbf{r}_u) = \sum_{i=1}^{m} w_i(\mathbf{r}_u)z(\mathbf{r}_i) = \mathbf{w}^T(\mathbf{r}_u)\mathbf{z}(\mathbf{r}). \tag{5.3}$$

The Kriging weights, $\mathbf{w}(\mathbf{r}_u)$ are computed from the spatial correlation or covariance between the observed locations $\mathbf{C}(\mathbf{r}_i,\mathbf{r}_j)$, and interpolation location $(\mathbf{r}_i,\mathbf{r}_j)$, based on the separation distance between them. It does not depend on the observations and mostly assumes a second order stationarity. Detailed discussion about Kriging is already presented in Section 3.3.

5.3.2 Multi-Model Bayesian Kriging

Kriging is among the best linear unbiased predictors because it uses the best model that describes the entire dataset to predict the values at the unobserved locations. The accuracy of the prediction decreases as we move farther away from the observation location. Also, there is the issue of under/overfitting for the locations where the model is not the best fit. Given that a dataset could be represented by different models that best describe the underlying process generating the data, a Kriging model that uses a weighted sum of the different models is proposed. The entire region is divided into $\mathscr{K}$ GP models, each with two separate stationary

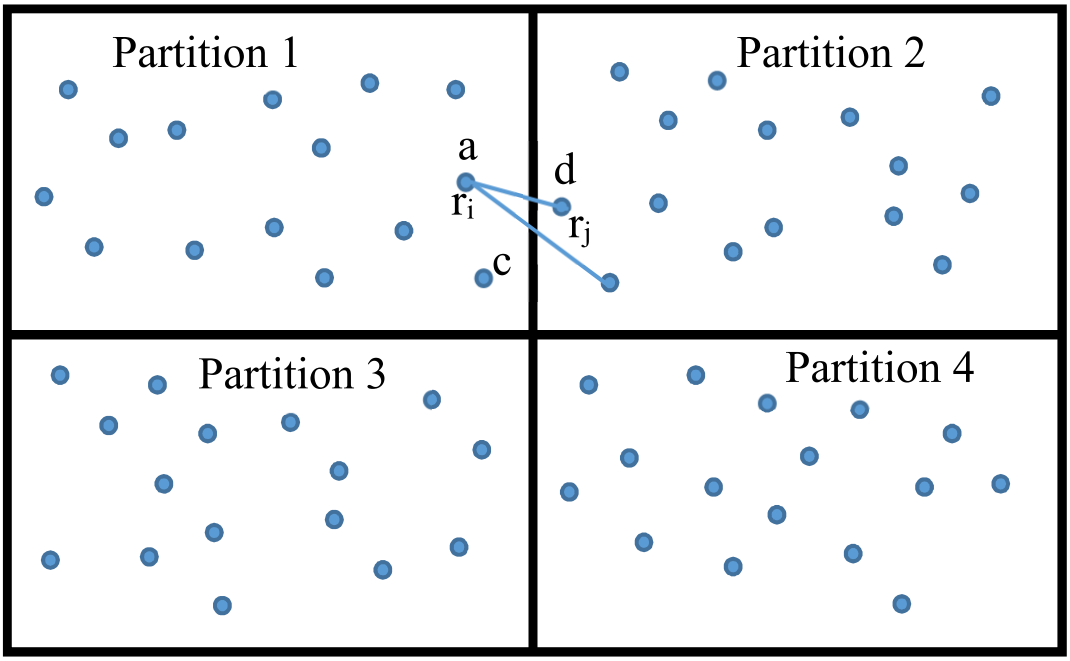

Figure 5.1 A rectangular grid network of 4 partitions

covariance structure $c_{i,k}(\cdot)$, $i = 0,1; k = 1,...,\mathscr{K}$ as in Section 5.4.

$$C_{k|r}(r_i, r_j) = \frac{c_{0,k(r_i - r_j)}}{c_{1,k}^{1/2}(r_i - t)c_{1,k}^{1/2}(r_j - t)}. \tag{5.4}$$

Here, t is the prediction site, $c_{1,k}$ is the local covariance kernel that models the stationary part and allows each model to maintain its distinct smoothness properties while $c_{0,k}$ is the global covariance function that models the non-stationary part and cross-correlation across the models.

Normally, the prediction at site **a** is computed with only the model parameters of the partition it falls under, in this case partition 1. This will either lead to over or under fitting. With the proposed MMBK, the contribution of each partition to the prediction at the site **a** is factored in by taking their weighted sum into consideration. For instance, the farther away point **d** is from point **a**, the less the value it will contribute to the prediction at the site **a**. Note that prediction points **a** and **d** could be in the same or different partition. Thus the nearer $\mathbf{r}_j$ is to **a** or **d** the larger the covariance and vice versa.

The partition could be either hard or soft partition. Hard partition is done with the spatial location of the points using directed graphs while the soft partition is done using the correlation of measurements. The approach adopted in this work is the soft partition.

For each unobserved location $\mathbf{r}_u$, each local Kriging model, k is used, to predict the response variable as sensed from that location. Without causing misunderstanding, r will be used rather than $\mathbf{r}_u$ for simplification. However, bear in mind that r is still a vector. The same as for the prediction site t, which is also a vector indicating the location of the prediction site in 2D space. Each GP model is defined by its mean $\boldsymbol{\mu}(\mathbf{r})$, covariance matrix $\mathbf{C}(\mathbf{r})$ and prior process variance, σ^2.

The likelihood of the prediction given the location is then evaluated according to (5.5).

$$p(\mathbf{z}_n|k,\sigma^2,r) = \frac{exp\left[-\frac{1}{2\sigma^2}\mathbf{z}_n^T\mathbf{C}_n^{-1}(k|r)\mathbf{z}_n\right]}{\left[\sigma^n(2\pi)^{n/2}||\mathbf{C}_n(k|r)||^{1/2}\right]}, \tag{5.5}$$

where $||\cdot||$ is the determinant, and $\mathbf{C}_n(k|r)$ is a covariance matrix of all the n observations given the prediction site r. With the choice of a uniform prior and inverse chi-square for σ^2, the expected likelihood becomes

$$\begin{aligned}\mathscr{L}(\mathbf{z}_n|k,t) \quad = \quad & \frac{1}{(2\pi)^{(n-p)/2}||\mathbf{C}_n(k|t)||^{1/2}\,||\mathbf{g}_n^\top\mathbf{C}_n^{-1}\mathbf{g}_n||^{1/2}} \times \\ & \frac{(\sigma_0^2\nu_0/2)^{\nu_0/2}}{\Gamma(\nu_0/2)}\frac{\Gamma(\nu_n/2)}{(\sigma^2_{n|k,t}\nu_n/2)^{\nu_n/2}},\end{aligned} \tag{5.6}$$

where Γ is the gamma function and $\mathbf{g}_n$ is an n-dimensional vector of the interpolating point and observation locations, with $\nu_n = \nu_0 + n - p$ and

$$\sigma^2_{n|k,t} = \frac{\nu_0\sigma_0^2 + (n-p)\hat{\sigma}_n^2(k|t)}{\nu_0 + n - p}, \tag{5.7}$$

The restricted maximum likelihood estimator (MLE) $\hat{\sigma}_n^2(k|t)$ of the prior σ^2 is given by,

$$\hat{\sigma}_n^2(k|t) = \frac{(\mathbf{z}_n - \mathbf{g}_n\hat{\beta}_n(k|t))^\top\mathbf{C}_n^{-1}(k|t)(\mathbf{z}_n - \mathbf{g}_n\hat{\beta}_n(k|t))}{n-p}, \tag{5.8}$$

and the MLE of the mean is given by,

$$\hat{\beta}_n(k|t) = (\mathbf{g}_n^\top\mathbf{C}_n^{-1}(k|t)\mathbf{g}_n)^{-1}\mathbf{g}_n^\top\mathbf{C}_n^{-1}(k|t)\mathbf{z}_n. \tag{5.9}$$

After collecting a set of measurements $\mathbf{Z}_n^t$ = $\{\mathbf{z}_{1:n}^t\}$, up to time t_n, the probability density function (PDF) of the traffic state $p(\mathbf{x}_n^t|\mathbf{Z}_n^t)$ at time t_n, is computed using Bayes' rule as:

$$p(\mathbf{x}_n^t|\mathbf{Z}_n^t) = \frac{p(\mathbf{z}_n^t|\mathbf{x}_n^t)p(\mathbf{x}_n^t|\mathbf{Z}_{n-1}^t)}{p(\mathbf{z}_n^t|\mathbf{Z}_{n-1}^t)}, \tag{5.10}$$

where the likelihood, $p(\mathbf{z}_n^t|\mathbf{x}_n^t)$ is given by (5.5), and $p(\mathbf{z}_n^t|\mathbf{Z}_{n-1}^t)$ is a normalizing constant. The value of traffic state at unobserved location r after n observation is then given by the weighted sum of GP models as,

$$\hat{\mathbf{x}}_n(\mathbf{r}_u) = \sum_{k=1}^{\mathscr{K}} \omega_n^k(r)\mathbf{b}_n^T(\mathbf{r}_u|k,t)\mathbf{z}_n, \tag{5.11}$$

where $\omega_n^k(r)$ is the conditional weight of each GP model given the location of the response variable computed by (5.12),

$$\omega_n^k(r) = \frac{\omega_0^k p(\mathbf{z}_n|k,\sigma^2,r)}{\sum_{k=1}^{\mathscr{K}} \omega_0^k p(\mathbf{z}_n|k,\sigma^2,r)}, \tag{5.12}$$

and $\mathbf{b}_n(r|k,r)$ is equivalent to the normal Kriging weight and is given by,

$$\mathbf{b}_n(r_u|k,t) = \mathbf{C}_n^{-1}(k|r,)\mathbf{c}_n(r_u,k|t). \tag{5.13}$$

Here $\mathbf{C}_n$ is the covariance matrix of interpolating points. and $\mathbf{c}_n$ is the covariance vector of the interpolating points and prediction point of all the mixture models. These could be computed using any kernel from the family of kernels in equations (5.14) - Matérn, (5.15) - spherical, (5.16) - Gaussian, or (5.17) - exponential. A series of experiments is performed (see Section 5.4.2) to determine the most accurate kernel.

$$\mathbf{C}_{n,\ell}(\mathbf{r}) = (1+\frac{\sqrt{3}}{\theta}\mathbf{r})exp(-\frac{\sqrt{3}}{\theta}\mathbf{r}), \tag{5.14}$$

$$\mathbf{C}_{n,\ell}(\mathbf{r}) = 1-\frac{3\mathbf{r}}{2\theta}+0.5(\frac{\mathbf{r}}{\theta})^3, \tag{5.15}$$

$$\mathbf{C}_{n,\ell}(\mathbf{r}) = exp(-\frac{\mathbf{r}^2}{\theta}), \tag{5.16}$$

$$\mathbf{C}_{n,\ell}(\mathbf{r}) = exp(-\frac{1}{\theta}\mathbf{r}). \tag{5.17}$$

In equations (5.14) to (5.17), $\mathbf{r} = ||\mathbf{r}_i - \mathbf{r}_j||_2$ is the ℓ_2 norm and θ is the learning/scale parameter. The procedure is summarized in Algorithm 6.

Algorithm 6 Multi-Model Bayesian Kriging Algorithm with Nonstationary Covariance Functions

```
Input: Road network map, Sensor Locations, Flow and Speed Measurements
Output: Predicted Traffic State $\mathbf{x}(\mathbf{r}_u)$
  1. Clustering and Model Length Selection
  <Cluster the road network using NMF to determine model number $\mathscr{K}$>
  2. Model Parameter Estimation
  <Estimate the Non-stationary Covariance Parameters: characteristic
  length scale, model means and inverse chi-2 prior>
  for <Each Prediction Site> do
      for $k \leftarrow 1$ to $\mathscr{K}$ do
          Compute $\mathbf{C}_n(k|t)$, $\mathbf{c}_n(k|t)$ using (5.14) to (5.17)
          Compute $\hat{\boldsymbol{\beta}}_n(k|t)$ using equation (5.9)
          Compute $\hat{\mathbf{z}}_n(k|t)$
  1. Bayesian Inference
          Compute the Expected Likelihood $\mathscr{L}(\mathbf{z}_n|k,t)$ using equation (5.6)
          Update the Posterior using equation (5.10)
      end for
      Compute the traffic state $\mathbf{x}_n(\mathbf{r}_u)$ using equation (5.11)
  end for
```

5.3.3 Estimation of Optimal Mixture Length

The parameters to be estimated include the mean, covariance matrices and the optimal number of mixture models. As the choice of the prior weight is critical for weighted model averaging, they were chosen according to the methods recommended by [196]. The following properties were provided as a guide in choosing the weights, namely:

- Dilution property - highly correlated models should be given lower weights while less correlated models should be given higher weights.

- Strong dilution property - When a new model is to be added, and they have a similar weight to the already existing model(s), the weight of the existing identical model(s) should be divided among the newly added model while the weights of other models remain the same.

- Monotonicity property - The weights of existing models may change but should not increase when new models are added.

5.3.4 Non-Negative Matrix Factorisation

To determine the optimal mixture length $\mathscr{K}$ and hence the number of mixture models to use, NMF was used to reduce the dimension of the data. NMF is Linear dimensionality reduction (LDR) often applied in situations where the underlying factors could be expressed as non-negative. In LDR, the goal is to reduce a high dimensional matrix $V \in \mathbb{R}^{m \times n}$, into two lower-dimensional matrices $W \in \mathbb{R}^{m \times \mathscr{K}}$ and $H \in \mathbb{R}^{\mathscr{K} \times n}$ as,

$$V \approx WH \tag{5.18}$$

Each row in V, say, v_i is computed as a weighted sum of some basis elements where each row in W is the weight and each row in H is the basis according to,

$$v_i = \sum_{j=1}^{n} w_{ij} h_j. \tag{5.19}$$

When there is no constraint on the values of W and H such that they could be either positive or negative, LDR is equivalent to PCA and truncated SVD. When columns of W are assumed independent, it becomes equivalent to Independent component analysis (ICA) [197]. When either W or H or both are assumed sparse, the approximation becomes equivalent to spare PCA [198]. When W or H are constrained to be positive (non-negative), it becomes equivalent to NMF. There are different ways of estimating the reconstruction error. The most common method is the use of the Frobenius norm because it is easy to efficiently compute an optimal approximation using truncated SVD, and it also assumes that the reconstruction error is Gaussian.

$$min|X - WH|_F^2. \tag{5.20}$$

The choice of NMF is predicated on the fact that road traffic is flow can be expressed with non-negative functions. For a non-negative matrix $V \in \mathbb{R}^{m \times n}$, the aim of NMF is to factorise the matrix into two non-negative matrices $W \in \mathbb{R}^{m \times \mathscr{K}}$ and $H \in \mathbb{R}^{\mathscr{K} \times n}$ such that the reconstruction error $||V - WH||^2$ is minimised. The parameter $\mathscr{K}$, is chosen such that it is less than the minimum of m and n, $\mathscr{K} < \min(m,n)$. It could be used to represent the number of correlated clusters in the dataset. NMF is computed by formulating an objective function which is then iteratively optimised. Different optimisations methods are employed such as alternating least squares (ALS) [199], Alternating nonnegative least squares (ANLS) [200–202], multiplicative update [203], hierarchical alternating least squares (HALS) [204], and initialisation [205].

One major challenge in NMF is the choice of optimal value for $\mathscr{K}$. In this work, the method of the Cophenetic Correlation Coefficient (CCC) is adopted as used in [206]. The choice of this approach for the optimal cluster length is predicated on the fact that the cophenetic correlation coefficient computes the similarities between observations in a hierarchical clustering tree. NMF is computed several times, say, 100 or 200 with different values of $\mathscr{K}$ each iteration. For each run, the CCC is computed and the value with the maximum number of occurrences chosen as the optimal rank.

5.4 Performance Evaluation

The algorithm is tested with a real dataset from Santander, Spain, which was obtained from the EU SETA Project [207].

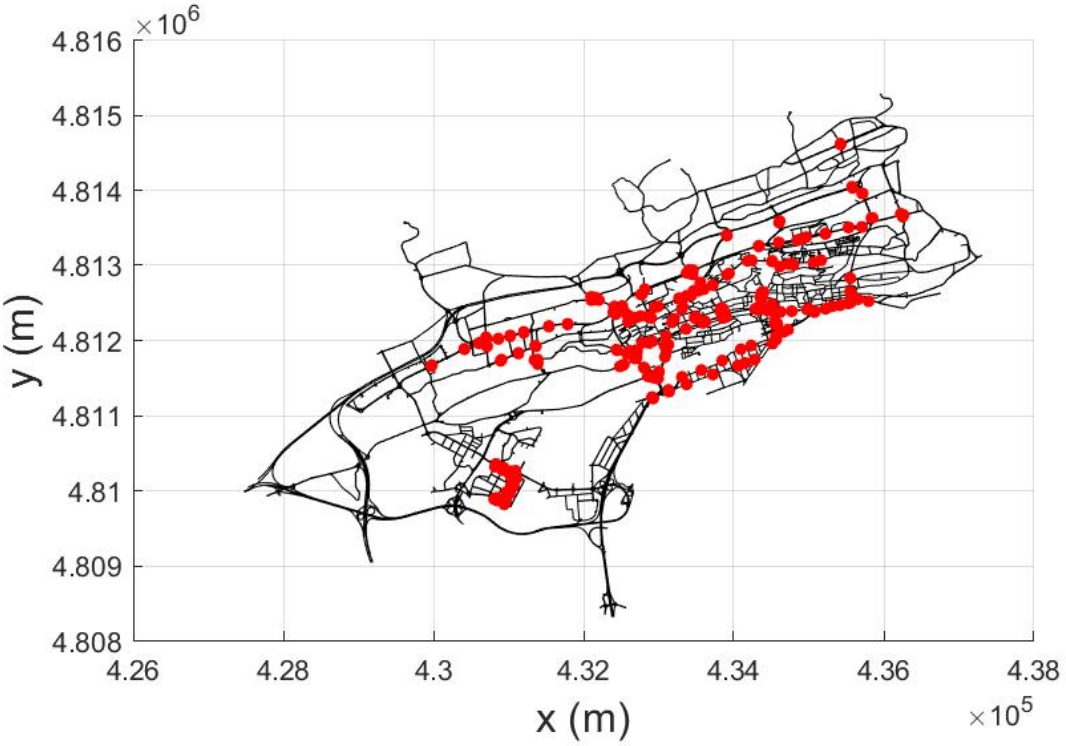

Figure 5.2 Santander road network with sensor locations

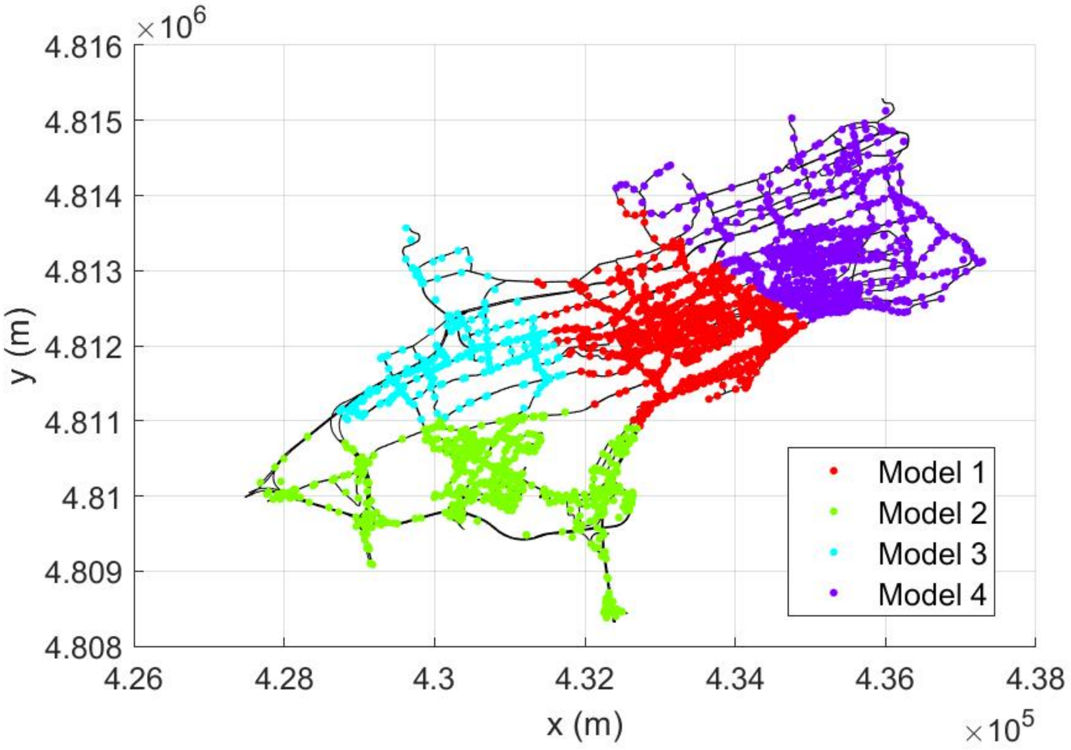

Figure 5.3 Santander road network with sensor and interpolated locations

The dataset consists of physical sensor measurements at 296 locations in the city (Fig. 5.2). Sensors were installed at these 296 locations as shown in the figure and measurements at an additional 3810 (total 4106) locations (Fig. 5.3), were simulated using Advanced Interactive Microscopic Simulator for Urban Networks (AIMSUN) [208]. These AIMSUN measurements were used as ground truth for performance evaluation of the algorithm. The physical sensor measurements were used as the training set, and those at the other locations were interpolated using the proposed method. The root mean squared error (RMSE) (2.9) of the traffic flow was computed and benchmarked with the traditional Kriging approach. The normalised (relative) root mean squared error (NRMSE) (2.10) was also computed to access the accuracy of the proposed method relative to the actual measurement.

5.4.1 Simulation Design

Figure 5.3 shows the city map partitioned into a total of 4106 segments. Magnetic loop

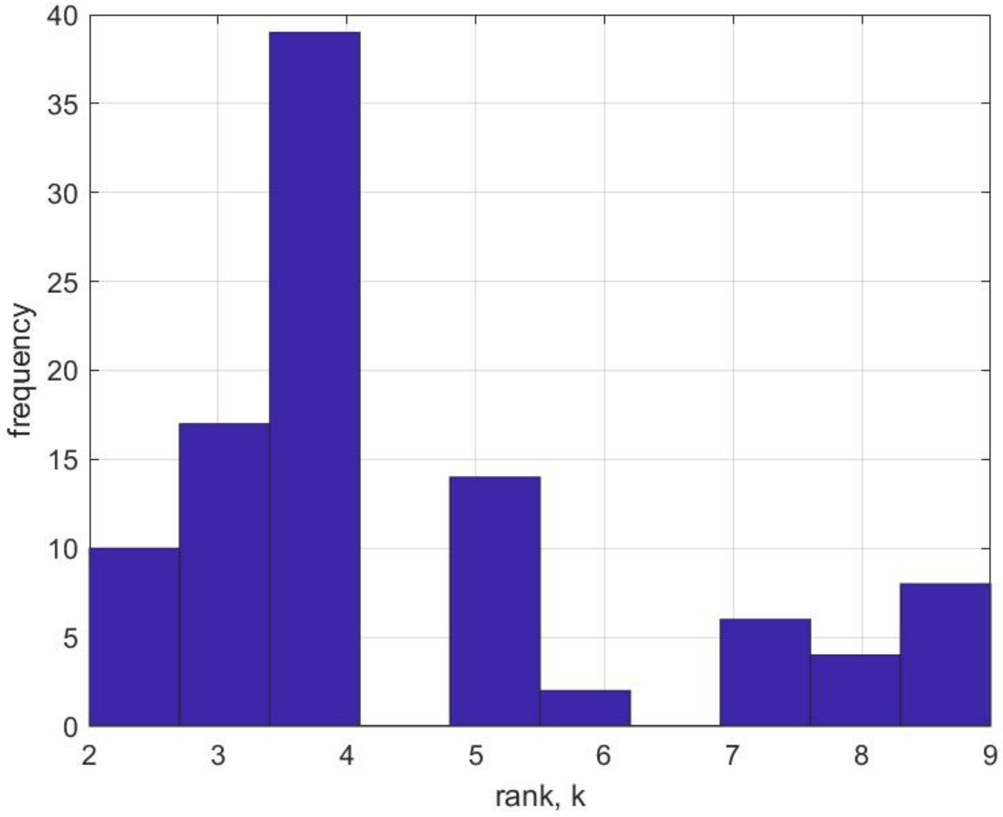

Figure 5.4 Plot showing how to compute optimal mixture length

detectors were installed in 296 of these segments as shown in Figure 5.2 and forms the measurement **z**. The objective of this study is to predict traffic flow at the other locations using the available observations. Measurements at those other locations are simulated using AIMSUN to serve as a Ground Truth for performance evaluation.

To determine the correlations between the road segments, NMF was used to decompose the network into different related clusters. The method discussed in Section 5.3.3 was used. NMF was computed 100 times with different values of k ranging from 2 to 10 each iteration. For each run, the CCC was computed, and the value with the maximum number of occurrences was chosen as the optimal rank and hence the mixture model length $\mathscr{K}$.

The result for the 100 runs of NMF is shown in Fig. 5.4. This shows the optimal mixture length $\mathscr{K}$ to be 4 (Fig. 5.2). The non-stationary covariance parameter of (5.4) was determined using minimax distance criterion [209].

5.4.2 Results and Discussion

Investigation with Different Kernels

The effect of different kernels on the prediction error is investigated for a fixed value of $\mathscr{K}$.

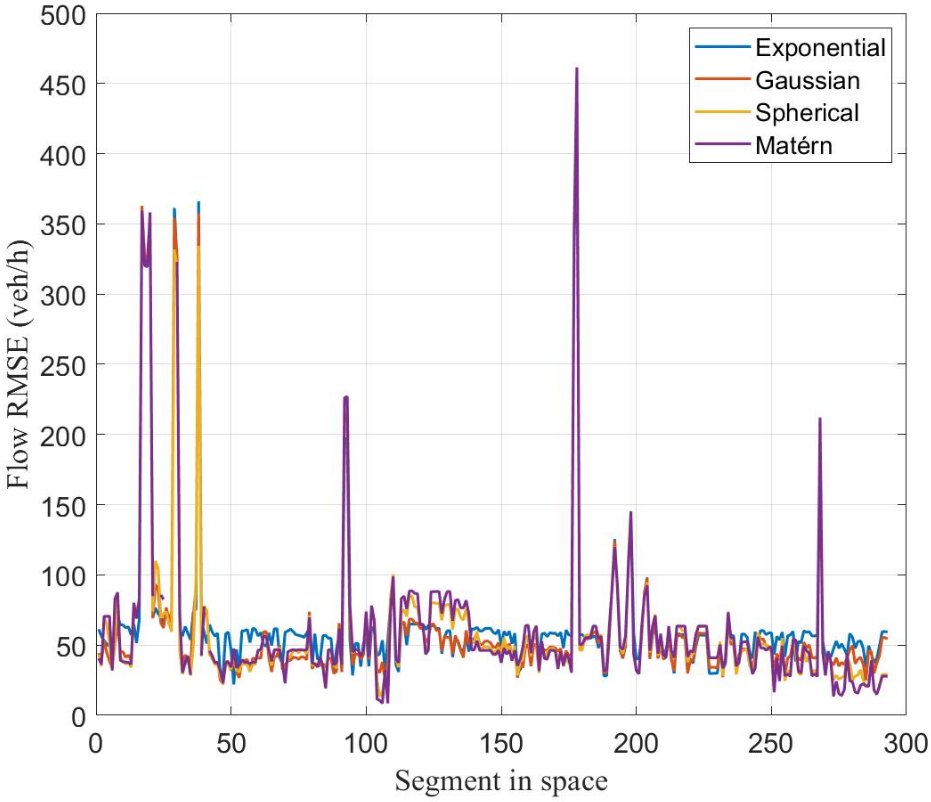

Figure 5.5 RMSE and NRMSE of flow across the fourth cluster at a given time.

Figure 5.5 shows the root mean squared error for different kernels for a section of the road segment where sensors are installed. The mean of the flow for the different kernels is 62 veh/h, 64 veh/h, 130 veh/h and 106 veh/h respectively for the exponential, Gaussian, spherical and Matérn kernels. It indicates that the best kernel for the given dataset is the Matérn kernel. It is observed from Figure 5.5 that there are some locations where the other kernels perform better than the exponential kernel. However, the overall performance of the exponential kernel is better than the others. Hence, the exponential kernel is used in the performance comparison with the traditional Kriging.

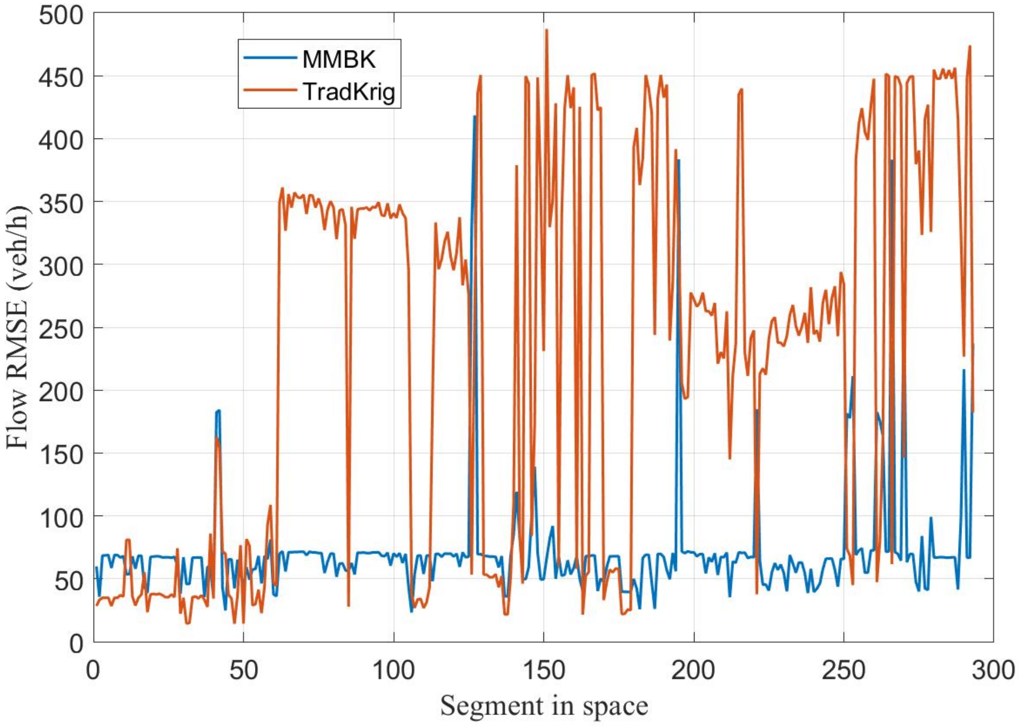

Figure 5.6 The RMSE of flow across the sensor locations at a given time.

Comparison with Traditional Kriging

Figures 5.6 shows the RMSE of the flow for sections of the road segment where sensors are installed. The minimum, maximum and mean of the RMSE is shown in Table 5.1. Whereas the mean RMSE for the Bayesian Kriging is 72 veh/h that of the traditional Kriging is 230 veh/h representing 65% improvement. The minimum RMSE for the Bayesian Kriging is 23 veh/h while that of traditional Kriging is 14 veh/h. This represents a decrease in the minimum accuracy. This is expected as the MMBK takes the average of all the different Kriging models. The maximum RMSE for the MMBK is and 416 veh/h, while that of the traditional Kriging is 487 veh/h, representing 14.3% improvement. Overall, there is an improvement in the average RMSE, thereby ensuring that the predicted traffic state is close to the actual state for the most part of the prediction horizon.

Table 5.1 Minimum, maximum, mean and percentage improvement

	min	max	mean
RMSE MMBK (veh/h)	23	416	72
RMSE TradKrig (veh/h)	14	487	230

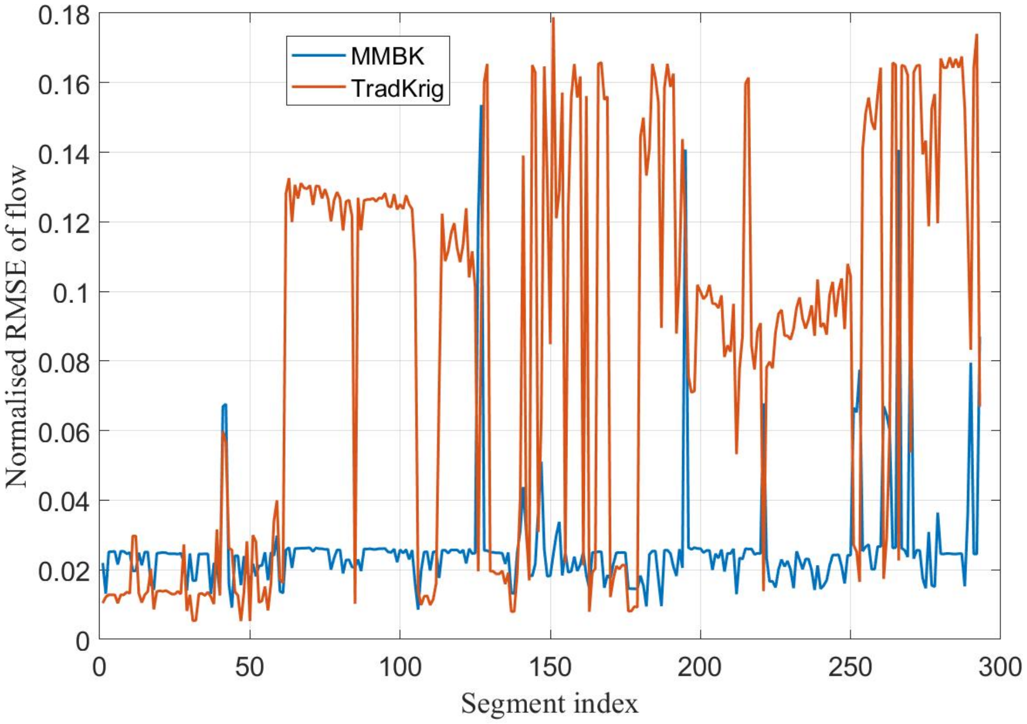

Figure 5.7 NRMSE of flow across the sensor locations at a given time.

Figure 5.7 shows the normalised RMSE (NRMSE) of the traffic flow for a section of the road network where measurements are available. It would be observed that the accuracy of the prediction is more stable for the MMBK approach compared to the traditional Kriging. However, there are some sections of the road where the performance of the traditional Kriging is better than the proposed approach. This is expected as mentioned earlier since the MMBK is a weighted sum of the different Kriging models which attempts to reduce the overall variance of the estimated traffic state from the actual traffic state. The normalised RMSE also shows that the percentage of RMSE is well below 5% for the majority of the cases with a few outliers. It can see that by aggregating Bayesian local Kriging results, the proposed approach has achieved improvement in terms of prediction accuracy.

5.5 Summary

In this work, a multi-model Bayesian Kriging approach with a discriminative covariance function conditioned on the observation at each location is proposed for the computation of traffic state. Information from the surroundings of the current segment is weighted to find each traffic state. The method possesses the potentiality to account for congested regions and interactions in the upstream and downstream of the congestion.

In the commonly used Kriging approaches, the covariance function depends only on the separation distance irrespective of the traffic at the considered locations. A key limitation of such an approach is its inability to capture well the traffic dynamics and transitions between different states. This paper proposes a multi-model Bayesian Kriging approach for the prediction of urban traffic. The approach can capture the dynamics and fluctuations in traffic flow between different states by modelling the changes via a covariance matrix.

The main novelty consists in representing both stationary and non-stationary changes in traffic flows by a discriminative covariance function conditioned on the observation at each location. A local covariance function captures the model behaviour at a given prediction site while the global covariance function models the interaction among the different models. The use of a weighted sum of the different Kriging models ensures that the overall estimation variance is minimised.

In this work, the regressor is assumed to be the same for all the models. The work also considered the same correlation lengths for all the $\mathscr{K}$ concurrent models. A possible further area of research would be to investigate the effect of using different trends and covariance structures. This could potentially accommodate more uncertainty in the process trend.

Uniform prior to the weights is used in the present work. In the present work, the optimal cluster size is selected using NMF. The predictions in and around the cluster boundaries were computed by using a weighted sum of each cluster's prediction. The use of other methods would be explored in future work to improve the prediction results for clusters with strong local optima.

In future work, an informative prior that could consider some road attributes like the number of lanes, intersections, parks and seasonal variations of traffic will be explored. Soft partitioning is used in the present work. In future work, hard partitioning using a directed graph will be investigated together with soft partitioning using historical data.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

This thesis proposed different approaches and algorithms for road traffic state estimation and prediction. It builds on existing approaches by proposing the use of a discriminative covariance model conditioned on the observation at each location. The proposed approach can account for congested regions and interactions in the upstream and downstream of the congestion. A summary of the key contributions of each chapter of the thesis is presented below.

In Chapter 2, the theoretical background of intelligent transportation systems, traffic estimation approaches and review of related works was presented. The traffic flow models discussed are microscopic flow models, macroscopic traffic flow models, and mesoscopic flow models. Different traffic state estimation approaches such as neural networks, deep learning, Kalman filters, principal component analysis, and support vector machine/regression were discussed. The chapter concluded with a review of related works as it relates to this study with special emphasis on Kriging and PF state-of-the-art.

Chapter 3 presented a novel approach to tackle the problem of missing and sparse data in traffic estimation. This approach entails interpolating the missing values using Kriging with a level of confidence assigned to the Kriged values by computing their interpolation error variance. This level of confidence is then used to compute the weight to be assigned during the computation of innovation terms used in PF. An expression of the likelihood function is derived for the case when the missing value is calculated based on Kriging interpolation. With the Kriging interpolation, the missing values of the measurements are predicted, which are subsequently used in the computation of likelihood terms in the particle filter algorithm.

This was tested using simulated and real data by assigning fixed test-values to the weighting factor. From the results presented benefit of lowering the weighting of interpolated values as compared to actual measurements has offered an improvement. The results show 23% to 36.34% improvement in RMSE values for the synthetic data used. A multi-step ahead traffic estimation approach that captures the dynamic and stochastic nature of traffic using discriminative covariance functions conditioned on the data at each location is developed.

Chapter 4 presented a traffic estimation for a large road network with different missing data ratios. The computational overhead of the large network was addressed by using a method called reduced measurement space proposed in [181] to select the most influential and information-rich segments in the road network. These are subsequently used in the particle filter measurement update step. Missing data in the selected segments are imputed using Kriging. A 1000-segment road network was simulated using SUMO. Different missing data ratios ranging from 10% to 70% were tested for different sizes of road network ranging from 100 to 1000 segments.

The results indicate that considering a larger number of segments would reduce the overall estimation error even when the missing data ration is high. From the foregoing results and discussion, it is recommended that the best estimation accuracy would be obtained when the entire road network is considered at once. The effects of computational overhead could further be reduced by using a distributed approach with a central control unit.

In Chapter 5, a multi-model Bayesian Kriging approach with discriminative covariance function conditioned on the observation at each location is proposed for the computation of traffic state. This entails dividing the entire road network into different partitions with similar distribution using a carefully chosen clustering method. Predictions at required locations in space and time are then performed using each of the models as sensed from that location. Generally, a given dataset could be represented by different models. Traditional Kriging makes use of the “best” model that explains the whole dataset. This often leads to over-fitting and underfitting with different scenarios. Using a weighted average of all the models has been shown to outperform a single model. A weighted sum of the models is then computed to get the estimated value at the location of interest. Normally, the covariance function is only dependent on the separation distance irrespective of the traffic situation at the locations. This makes it impossible for the model to capture traffic dynamics and transitions from free-flow to congested state, congested state to free-flow, etc. Our proposed approach can capture these dynamics and model it into the covariance matrix. The prior used are uniformly weighted with an inverse chi-square distribution. The proposed method is able to account for congested regions and interactions in the upstream and downstream of the congestion.

6.2 Future Work

In future work, the algorithm will be validated further by empirically computing the weighting factor β of the Kriging estimate of the missing measurements with real data from a larger road network. In addition, the use of different methods in calculating the Kriging variance would be investigated.

Uniform prior weights are used in the present work. In future work, an informative prior that could consider some road attributes like the number of lanes, intersections, parks and seasonal variations of traffic would be explored. Some additional road features like parks, shopping centres would be investigated in future work. Soft partitioning is used in the present work. In future work, hard partitioning using a directed graph will be investigated together with soft partitioning using historical data.

The clustering of the road network is achieved using non-negative matrix factorisation. It was noted that the determination of the optimal cluster size using NMF poses serious challenges. Although the cophenetic correlation coefficient is used to address this challenge, it would be observed that this approach is not entirely accurate consistently. For instance, in the 100 simulations performed to choose the optimal cluster size, only about 40 per cent of the results gave the optimal cluster length as 4, the other 60 per cent were distributed across cluster size of 2, 3 5 6 7 8 and 9. With this, one can never conclude with all certainty that using a different cluster size would not produce a better result. It would be worth trying other clustering methods in future work or building a model that would use adaptive cluster size.

The experiments for the large urban road network were performed using simulated data which spans one thousand segments. The simulation also involved only two types of vehicles with a fixed driver behaviour modelling which is not possible in a real-life scenario. Future work would consider the use of real data. Also, the effects of other road features and whether the information could be incorporated into the system for contextual and improved performance.

This work assumed the same regressor function for all the models. It also considered the same correlation lengths for all the $\mathscr{K}$ concurrent models. In order to allow for more uncertainty in the process trend, future research could be directed towards investigating the effect of using different trends and covariance structures.

6.3 Personal Reflections 2016 to 2020

The PhD journey starts with the selection of a research topic. This is often preceded by a brief or extensive literature survey in one's area of interest to identify gaps in the existing body of knowledge. Thus I began this stage with an open mind. My supervisor was very helpful in pointing me to different areas, and I eventually settled for Sensor Data Fusion for Improving traffic Mobility in Smart Cities. Before I started, I had expected an easy sail. However, the numerous failed attempts, codes not working as expected and being told that some of my "results are incorrect and that I have not made much progress over the past one year..." made me feel depressed and began to question my ability to carry through.

The take away from this whole experience is persistence, focus and determination. There are times when the experimental results are not giving expected results, making me spend days, weeks and some times months without any hope in sight. Having a supportive supervisor helped me to overcome the temptation of quitting at those periods. Determination kept me going whenever I submit a report with the hope that I will be commended for doing something "great" only for the supervisory team to condemn the whole report. There was always that feeling of "I am not good at this and can never be of any good". The assurance of my wife telling me that she believes in me and reminding me how I have overcome challenges in the past kept me going and focussed on completing the program.

Bibliography

[1] R. Boel and L. Mihaylova, “A compositional stochastic model for real-time freeway traffic simulation.” *Transportation Research Part B: Methodological*, vol. 40, no. 4, pp. 319–334, 2006.

[2] L. Mihaylova, R. Boel, and A. Hegyi, “Freeway traffic estimation within particle filtering framework,” *Automatica*, vol. 43, no. 2, pp. 290–300, 2007.

[3] “Chris Nash with contributions from partner: UNITE (UNIfication of accounts and marginal costs for Transport Efficiency) Final Report for Publication, Funded by 5th Framework RTD Programme,” Tech. Rep., 2003.

[4] S. Alonze, “Traffic Congestion to Cost the UK Economy More Than £300 Billion Over the Next 16 Years,” *INRIX*, 2014.

[5] A. de Palma and R. Lindsey, “Traffic congestion pricing methodologies and technologies,” *Transp. Res. Part C Emerg. Technol.*, vol. 19, no. 6, pp. 1377–1399, 2011.

[6] D. Schrank, T. Lomax, and B. Eisele, “The 2009 Urban Mobility Report. College Station: Texas Transportation Institute, Texas A&M University.” Tech. Rep., 2009.

[7] T. Lomax, D. Schrank, and B. Eisele, “2015 Urban Mobility Scorecard: Texas Transportation Institute, Texas A&M University.” Texas, Tech. Rep., 2015.

[8] G. Cookson and B. Pishue, “INRIX Global Traffic Scorecard,” Tech. Rep. February, 2017.

[9] B. V. Arem, H. R. Kirby, M. J. M. Van Der Vlist, and J. C. Whittaker, “Recent advances and applications in the field of short-term traffic forecasting,” *International Journal of Forecasting*, vol. 13, no. 1, pp. 1–12, 1997.

[10] Y. Zhang and Y. Zhang, “A Comparative Study of Three Multivariate Short-Term Freeway Traffic Flow Forecasting Methods With Missing Data,” *J. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 205–218, 2016.

[11] S. P. Hoogendoorn and P. H. L. Bovy, “Generic gas-kinetic traffic systems modeling with applications to vehicular traffic flow,” *Transp. Res. Part B*, vol. 35, no. 4, pp. 317 – 336, 2001.

[12] M. Papageorgiou, M. Ben-Akiva, J. Bottom, P. H. L. Bovy, S. P. Hoogendoorn, N. B. Hounsell, A. Kotsialos, and M. Mcdonald, *Chapter 11, ITS and Traffic Management in: Handbooks in Operations Research and Management Science.* Amsterdam: Elsevier, 2007.

[13] L. A. Pipes, "An Operational Analysis of Traffic Dynamics," *J. Appl. Phys.*, vol. 24, no. 3, pp. 274–281, 1953.

[14] P. G. Gipps, "A Behavioural Car-Following Model for Computer Simulation," *Transp Res Part B Methodol*, vol. 15, no. 2, pp. 105–111, 1981.

[15] A. Messmer and M. Papageorgiou, "METANET: A macroscopic simulation program for motorway networks," *Traffic Eng. Control*, vol. 31, no. 8-9, pp. 466 – 470, 1990.

[16] C. F. Daganzo, "The Cell Transmission Model: A Dynamic Representation of Highway Traffic Consistent with the Hydrodynamic Theory," *Transpn. Res.-B*, vol. 28, no. 4, pp. 269–287, 1994.

[17] T. Seo, A. M. Bayen, T. Kusakabe, and Y. Asakura, "Traffic state estimation on highway: A comprehensive survey," *Annual Reviews in Control*, vol. 43, no. 2017, pp. 128–151, 2017.

[18] H.-p. Lu, Z.-y. Sun, and W.-c. Qu, "Big Data-Driven Based Real-Time Traffic Flow State Identification and Prediction," *Discrete Dynamics in Nature and Society*, vol. 2015, no. 1, pp. 1–11, 2015.

[19] H. Braxmeier, V. Schmidt, and E. Spodarev, "Kriged Road-Traffic Maps," in *Interfacing Geostatistics and GIS.*, J. Pilz, Ed. Heidelberg: Springer, 2009, ch. 9, pp. 105–119.

[20] M. Lowry, "Spatial interpolation of traffic counts based on origin-destination centrality," *Journal of Transport Geography*, vol. 36, no. 2014, pp. 98–105, 2014.

[21] X. Wang and K. M. Kockelman, "Forecasting Network Data: Spatial Interpolation of Traffic Counts Using Texas Data," *Transp. Res. Rec.*, vol. 2105, pp. 100–108, 2009.

[22] M. Gastaldi, G. Gecchele, and R. Rossi, "Estimation of Annual Average Daily Traffic from one-week traffic counts. A combined ANN-Fuzzy approach," *Transportation Research Part C: Emerging Technologies*, vol. 47, no. 2014, pp. 86–99, 2014.

[23] H. Zou, Y. Yue, Q. Li, and A. G. Yeh, "An improved distance metric for the interpolation of link-based traffic data using kriging: a case study of a large-scale urban road network," *Int. J. Geogr. Inf. Sci.*, vol. 26, no. 4, pp. 667–689, 2012.

[24] Y. Song, X. Wang, G. Wright, D. Thatcher, P. Wu, and P. Felix, "Traffic Volume Prediction With Segment-Based Regression Kriging and its Implementation in Assessing the Impact of Heavy Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. Preprint, no. Preprint, pp. 1–12, 2018.

[25] H. Yang, J. Yang, L. D. Han, X. Liu, L. Pu, S.-m. Chin, and H.-l. Hwang, "A Kriging based spatiotemporal approach for traffic volume data imputation," *PLoS ONE*, vol. 13, no. 4, pp. 1–11, 2018.

[26] N.-E. El Faouzi and L. A. Klein, "Data Fusion for ITS: Techniques and Research Needs," in *Transp. Res. Procedia*, vol. 15, no. June 2016. Elsevier B.V., 2016, pp. 495–512.

[27] H. B. Mitchell, *Data fusion: Concepts and ideas*, 2nd ed. Berlin: Springer Verlag, 2012.

[28] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: Where we are and where we're going," *Transp. Res. Part C Emerg. Technol.*, vol. 43, no. 3, Part 2, pp. 3–19, 2014.

[29] J. W. C. Van Lint and S. P. Hoogendoorn, "A Robust and Efficient Method for Fusing Heterogeneous Data from Traffic Sensors on Freeways," *Comput. Civ. Infrastruct. Eng.*, vol. 25, no. 8, pp. 596–612, 2010.

[30] L. Ambühl and M. Menendez, "Data fusion algorithm for macroscopic fundamental diagram estimation," *Transp. Res. Part C*, vol. 71, no. 1, pp. 184 – 197, 2016.

[31] C. S. Regazzoni and A. Tesei, "Distributed data fusion for real-time crowding estimation," *Signal Processing*, vol. 53, no. 1, pp. 47–63, 1996.

[32] E. Huang, C. Antoniou, Y. Wen, M. Ben-Akiva, J. Lopes, and J. Bento, "Real-time multi-sensor multi-source network data fusion using dynamic traffic assignment models," in *Proc. 2009 12th Int. IEEE Conf. Intell. Transp. Syst.*, 2009, pp. 1–6.

[33] C. Bachmann, B. Abdulhai, M. J. Roorda, and B. Moshiri, "A comparative assessment of multi-sensor data fusion techniques for freeway traffic speed estimation using microsimulation modeling," *Transp. Res. Part C*, vol. 26, no. 1, pp. 33–48, 2013.

[34] X. Zhou, "A Multi-Sensor Data Fusion Approach for Real-Time Lane-Based Traffic Estimation," Ph.D. dissertation, Arizona State University, 2015.

[35] A. Ladino, A. Y. Kibangou, H. Fourati, and C. Canudas de Wit, "Travel time forecasting from clustered time series via optimal fusion strategy," in *Proc. Eur. Control Conf. 2016*, Aalborg, Denmark, 2016.

[36] J. Chen, K. H. Low, Y. Yao, and P. Jaillet, "Gaussian Process Decentralized Data Fusion and Active Sensing for Spatiotemporal Traffic Modeling and Prediction in Mobility-on-Demand Systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 3, pp. 901–921, 2015.

[37] Y. Zheng, "Methodologies for Cross-Domain Data Fusion: An Overview," *IEEE Trans. Big Data*, vol. 1, no. 1, pp. 16–34, 2015.

[38] E. Lovisari, C. C. De Wit, and A. Kibangou, “A Fusion Algorithm for Traffic Density Estimation using Sensors and Floating Car Data,” in *Proc. 22nd ITS World Congr.*, Bordeaux, France, 2015, pp. 1 – 12.

[39] W. Elmenreich, “A review on system architectures for sensor fusion applications,” *Softw. Technol. Embed. Ubiquitous Syst.*, pp. 547–559, 2007.

[40] B. Khaleghi, A. Khamis, F. O. Karray, and S. N. Razavi, “Multisensor data fusion: A review of the state-of-the-art,” *Inf. Fusion*, vol. 14, no. 1, pp. 28–44, 2013.

[41] B. D. Greenshields, “The photographic method of studying traffic behavior,” in *Proc. 13th Annu. Meet. Highw. Res. Board.* Washington, D.C.,: Highway Research Board, 1934, pp. 382–399.

[42] C. F. Daganzo, *Traffic flow theory, in Carlos F. Daganzo (ed.) Fundamentals of Transportation and Traffic Operations*, C. F. Daganzo, Ed. Bingley, UK: Emerald Group Publishing, 1997.

[43] M. Saifuzzaman and Z. Zheng, “Incorporating human-factors in car-following models: A review of recent developments and research needs,” *Transportation Research Part C: Emerging Technologies*, vol. 48, no. 2014, pp. 379–403, 2014.

[44] M. Cremer and J. Ludwig, “A fast simulation model for traffic flow on the basis of boolean operations,” *Math. Comput. Simul.*, vol. 28, no. 4, pp. 297 – 303, 1986.

[45] M. Schreckenberg, A. Schadschneider, K. Nagel, and N. Ito, “Discrete stochastic models for traffic flow,” *Phys. Rev. E*, vol. 51, no. 4, pp. 2939–2949, 1995.

[46] K. Nagel and Schreckenberg, “A cellular Automaton Model for Freeway Traffic,” *J. Phys*, vol. 2, no. 12, pp. 2221–2229, 1992.

[47] K. Nagel, “Particle hopping models and traffic flow theory,” *Phys. Rev. E*, vol. 3, no. 6, pp. 4655–4672, 1996.

[48] K. Nagel, C. Barrett, and M. Rickert, “Parallel Traffic Micro-Simulation by Cellular Automata and Application for large scale Transportation Modeling,” Los Alamos New Mexico, Tech. Rep., 1996.

[49] D. Helbing, “Traffic and related self-driven many-particle systems,” *Rev. Mod. Phys.*, vol. 73, no. 4, pp. 1067–1141, 2001.

[50] S. Gurupackiam and S. Lee Jones, “Empirical Study of Accepted Gap and Lane Change Duration Within Arterial Traffic Under Recurrent and Non-Recurrent Congestion,” *International Journal for Traffic and Transport Engineering*, vol. 2, no. 4, pp. 306–322, 2012.

[51] T. Toledo, “Driving behaviour: Models and challenges,” *Transport Reviews*, vol. 27, no. 1, pp. 65–84, 2007.

[52] S. Moridpour, M. Sarvi, and G. Rose, “Lane changing models: A critical review,” *Transportation Letters*, vol. 2, no. 3, pp. 157–173, 2010.

[53] P. G. Gipps, “A model for the structure of lane-changing decisions,” *Transportation Research Part B*, vol. 20, no. 5, pp. 403–414, 1986.

[54] P. Hidas and K. Behbahanizadeh, “Microscopic simulation of lane changing under incident conditions,” *Proc., 14th International Symposium on the Theory of Traffic Flow and Transportation*, pp. 53–69, 1999.

[55] P. Hidas, “Modelling lane changing and merging in microscopic traffic simulation,” *Transportation Research Part C: Emerging Technologies*, vol. 10, no. 5, pp. 351–371, 2002.

[56] Q. Yang and H. N. Koutsopoulos, “A microscopic traffic simulator for evaluation of dynamic traffic management systems,” *Transportation Research Part C: Emerging Technologies*, vol. 4, no. 3 PART C, pp. 113–129, 1996.

[57] K. I. Ahmed, “Modeling drivers acceleration and lane changing behavior,” Ph.D. dissertation, Massachusetts Institute of Technology.

[58] M. Rahman, M. Chowdhury, Y. Xie, and Y. He, “Review of microscopic lane-changing models and future research opportunities,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1942–1956, 2013.

[59] M. J. Lighthill, F. R. S. And, and G. B. Whitham, “On kinematic waves II. A theory of traffic flow on long crowded roads,” in *Proc. R. Soc. London. Ser. A. Math. Phys. Sci.*, 1955, pp. 317 – 345.

[60] P. I. Richards, “Shock Waves on the Highway,” *Oper. Res.*, vol. 4, no. 1, pp. 42–51, 1956.

[61] G. F. Newell, “Comments on Traffic Dynamics.” *Transp. Res. Part B Methodol.*, vol. 23, no. 5, pp. 386 – 389, 1989.

[62] S. A. H. Algadhi and H. S. Mahmassani, “Modelling Crowd Behavior and Movement: Application to Makkah Pilgrimage,” in *Proc. 11th Int. Symp. Transp. Traffic Theory, Yokohama, Japan.*, M. Koshi, Ed. New York: Elsevier, 1990, pp. 59–78.

[63] Y. Sheffl, H. Mahmassani, and W. B. Powell, “A Transportation Network Evacuation Model.” *Transp. Res. Part A Gen.*, vol. 16, no. 3, pp. 209 – 218, 1982.

[64] C. Canudas-de Wit and A. Ferrara, “A variable-length Cell Transmission Model for road traffic systems,” *Transportation Research Part C: Emerging Technologies*, vol. 97, no. November, pp. 428–455, 2018.

[65] C. F. Daganzo, “The cell transmission model, part II: Network traffic,” *Transp. Res. Part B Methodol.*, vol. 29, no. 2, pp. 79–93, 1995.

[66] R. Boel and L. Mihaylova, "Modelling freeway networks by hybrid stochastic models," in *Proc. 2004 Intell. Veh. Symp.*, Perma, 2004, pp. 182–187.

[67] I. Prigogine and F. C. Andrews, "A Boltzmann-Like Approach for Traffic Flow," *Oper. Res.*, vol. 8, no. 6, pp. 789–797, 1960.

[68] S. L. Paveri-Fontana, "On Boltzmann-Like Treatments for Traffic Flow: A Critical Review of the Basic Model and an Alternative Proposal for Dilute Traffic Analysis," *Trans Res*, vol. 9, no. 4, pp. 225–235, 1975.

[69] D. Helbing, "Modeling multi-lane traffic flow with queuing effects," *Physica A*, vol. 242, no. 1-2, pp. 175–194, 1997.

[70] M. Ben-Akiva, M. Bierlaire, D. Burton, H. N. Koutsopoulos, and R. Mishalani, "Network State Estimation and Prediction for Real-Time Traffic Management," *Networks Spat. Econ.*, vol. 1, no. 3, pp. 293–318, 2001.

[71] R. Jayakrishnan and H. S. Mahmassani, "An evaluation tool for advanced traffic information and management systems in urban networks," *Transp. Res. Part C Emerg. Technol.*, vol. 2, no. 3, pp. 129–147, 1994.

[72] A. Jamshidnejad, I. Papamichail, M. Papageorgiou, and B. De Schutter, "A mesoscopic integrated urban traffic flow-emission model," *Transp. Res. Part C Emerg. Technol.*, vol. 75, no. 1, pp. 45–83, 2017.

[73] M. S. Ahmed and A. R. Cook, "Analysis of Freeway Traffic Time-Series Data by Using Box-Jenkins Techniques," *Transp. Res. Rec.*, no. 722, pp. 1–9, 1979.

[74] E. I. Vlahogianni, J. C. Golias, and M. G. Karlaftis, "Short term traffic forecasting: Overview of objectives and methods," *Transp. Rev.*, vol. 24, no. 5, pp. 533–557, 2004.

[75] H. J. VanLint and C. C. VanHinsbergen, "Short term traffic and travel time prediction models," *Artif. Intell. Appl. to Crit. Transp. issues. Chowdhury, R., Sade. S. (Eds.), Transp. Res. Circ. Natl. Acad. Press. Washingt. DC*, vol. E-C168, pp. 22 – 41, 2012.

[76] B. Williams, P. Durvasula, and D. Brown, "Urban Freeway Traffic Flow Prediction: Application of Seasonal Autoregressive Integrated Moving Average and Exponential Smoothing Models," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 1644, no. 98, pp. 132–141, 1998.

[77] T. Mai, B. Ghosh, and S. Wilson, "Short-Term Traffic Flow Forecasting Using Dynamic Linear Models," in *Proc. ITRN2011*, 2011.

[78] L. Chu, "Adaptive Kalman Filter Based Freeway Travel time Estimation," *Trb*, pp. 1–21, 2005.

[79] L. L. Ojeda, A. Y. Kibangou, and C. de Wit, "Adaptive Kalman Filtering for Multi-Step ahead Traffic Flow Prediction," in *Proc. 2013 Am. Control Conf.* Washington, DC: IEEE, 2013, pp. 4724–4729.

[80] E. A. Wan and R. Van Der Merwe, "The Unscented Kalman Filter for Nonlinear Estimation," in *Proc. IEEE 2000 Adapt. Syst. Signal Process. Commun. Control Symp. (Cat. No.00EX373)*, Lake Louise, AB, 2000, pp. 153–158.

[81] H. Chen and H. A. Rakha, "Real-time travel time prediction using particle filtering with a non-explicit state-transition model," *Transp. Res. Part C Emerg. Technol.*, vol. 43, no. 1, pp. 112–126, 2014.

[82] G. Foerster, "Traffic state estimation using hierarchical clustering and principal components analysis: a practical approach," in *Young Res. Semin.*, Brno, 2007.

[83] Y. Bin, Y. Zhongzhen, and Y. Baozhen, "Bus Arrival Time Prediction Using Support Vector Machines," *J. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 151–158, 2006.

[84] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, "Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions," *Expert Syst. Appl.*, vol. 36, no. 3, part 2, pp. 6164–6173, 2009.

[85] X. Zhang and J. A. Rice, "Short-term travel time prediction," *Transp. Res. Part C*, vol. 11, no. 3–4, pp. 187–210, 2003.

[86] J. Rice and E. Van Zwet, "A simple and effective method for predicting travel times on freeways," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 3, pp. 200–207, 2004.

[87] J. W. C. Van Lint, S. P. Hoogendoorn, and H. J. Van Zuylen, "Accurate freeway travel time prediction with state-space neural networks under missing data," *Transp. Res. Part C*, vol. 13, no. 5-6, pp. 347–369, 2005.

[88] J. W. C. Van Lint, "Online learning solutions for freeway travel time prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 38–47, 2008.

[89] Y. Zhang and A. Haghani, "A gradient boosting method to improve travel time prediction," *Transp. Res. Part C*, vol. 58, no. 2, pp. 308–324, 2015.

[90] M. Dougherty, "A Review of Neural Networks Applied to Transport," *Transp. Res. Part C*, vol. 3, no. 4, pp. 247–260, 1995.

[91] M. S. Dougherty and M. R. Cobbett, "Short-term inter-urban traffic forecasts using neural networks," *Int. J. Forecast.*, vol. 13, no. 1, pp. 21–31, 1997.

[92] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach," *Transp. Res. Part C Emerg. Technol.*, vol. 13, no. 3, pp. 211–234, 2005.

[93] W. Zheng, D.-H. Lee, M. Asce, and Q. Shi, "Short-Term Freeway Traffic Flow Prediction: Bayesian Combined Neural Network Approach," *J. Transp. Eng.*, vol. 132, no. 2, pp. 114–121, 2006.

[94] H. Van Lint, S. P. Hoogendoorn, and H. J. Van Zuylen, "State Space Neural Networks for Freeway Travel Time Prediction," in *Artif. Neural Networks — ICANN 2002. ICANN 2002. Lect. Notes Comput. Sci.*, J. Dorronsoro, Ed. Berlin, Heidelberg: Springer, 2002, vol. 2415, pp. 1043–1048.

[95] J. G. Hunt and G. D. Lyons, "Modelling dual carriageway lane changing using neural networks," *Transportation Research Part C*, vol. 2, no. 4, pp. 231–245, 1994.

[96] L. Chen and C. L. P. Chen, "Ensemble Learning Approach for Freeway Short-Term Traffic Flow Prediction," in *2007 IEEE International Conference on System of Systems Engineering*. IEEE, apr 2007, pp. 1–6.

[97] X. Chen, Z. He, and J. Wang, "Spatial-temporal traffic speed patterns discovery and incomplete data recovery via SVD-combined tensor decomposition," *Transportation Research Part C*, vol. 86, no. 1, pp. 59–77, 2017.

[98] S. Liao, J. Chen, J. Hou, Q. Xiong, and J. Wen, "Deep Convolutional Neural Networks with Random Subspace Learning for Short-term Traffic Flow Prediction with Incomplete Data," in *Proceedings of the International Joint Conference on Neural Networks*. IEEE, 2018, pp. 1–6.

[99] N. G. Polson and V. O. Sokolov, "Deep learning for short-term traffic flow prediction," *Transp. Res. Part C J.*, vol. 79, no. 1, pp. 1–17, 2017.

[100] G. Hinton, *Deep Belief Nets*. Boston, MA: Springer US, 2010, pp. 267–269. [Online]. Available: https://doi.org/10.1007/978-0-387-30164-8_208

[101] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 1701–1708.

[102] R. Collobert and J. Weston, "A unified architecture for natural language processing," pp. 160–167, 2008.

[103] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic Flow Prediction With Big Data : A Deep Learning Approach," *Intell. Transp. Syst. IEEE Trans.*, vol. 16, no. 2, pp. 1–9, 2014.

[104] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems 1," *Trans. ASME - J. Basic Eng.*, vol. 82, no. Series D, pp. 35–45, 1960.

[105] S. J. Julier and J. K. Uhlmann, "A New Extension of the Kalman Filter to Nonlinear Systems," in *Proc. 11th Int. Symp. Aerospace/Defence Sensing, Simul. Control.*, 1997.

[106] S. Julier, J. Ulhmann, and H. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. Automat. Contr.*, vol. 45, no. 3, pp. 477–482, 2000.

[107] I. T. Jolliffe, *Principal Component Analysis, Second Edition*, 2nd ed. New York: Springer-Verlag, 2002.

[108] M. E. Tipping and C. M. Bishop, “Probabilistic principal component analysis,” *J. R. Stat. Soc. Ser. B Stat. Methodol.*, vol. 61, no. 3, pp. 611–622, 1999.

[109] T. Tsekeris and A. Stathopoulos, “Measuring Variability in Urban Traffic Flow by Use of Principal Component Analysis,” *J. Transp. Stat.*, vol. 9, no. 1, pp. 49–62, 2006.

[110] Y. Zhu, Z. Li, H. Zhu, M. Li, and Q. Zhang, “A compressive sensing approach to urban traffic estimation with probe vehicles,” *IEEE Trans. Mob. Comput.*, vol. 12, no. 11, pp. 2289–2302, 2013.

[111] X. Jin, Y. Zhang, L. Li, and J. Hu, “Robust PCA-Based Abnormal Traffic Flow Pattern Isolation and Loop Detector Fault Detection,” *Tsinghua Sci. Technol.*, vol. 13, no. 6, pp. 829–835, 2008.

[112] E. J. Candès, X. Li, Y. Ma, and J. Wright, “Robust Principal Component Analysis?” *J. ACM*, vol. 58, no. 11, 2009.

[113] X. Xing, X. Zhou, H. Hong, W. Huang, K. Bian, and K. Xie, “Traffic Flow Decomposition and Prediction Based on Robust Principal Component Analysis,” in *Proc. 2015 IEEE 18th Int. Conf. Intell. Transp. Syst.* IEEE, 2015, pp. 2219–2224.

[114] B. E. Boser, I. M. Guyon, and V. N. Vapnik, “A Training Algorithm for Optimal Margin Classifiers,” in *Proc. Fifth Annu. Work. Comput. Learn. Theory – COLT ’92*, Pittsburgh, Pennsylvania, 1992, pp. 144–152.

[115] C. Cortes and V. Vapnik, “Support-Vector Networks,” *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[116] M. Yu, A. Rhuma, S. M. Naqvi, and J. Chambers, “Fall detection for the elderly in a smart room by using an enhanced one class support vector machine,” *17th DSP 2011 International Conference on Digital Signal Processing, Proceedings*, pp. 1–6, 2011.

[117] C. C. J. C. Burges, “A Tutorial on Support Vector Machines for Pattern Recognition,” *Data Min. Knowl. Discov.*, vol. 2, no. 2, pp. 121–167, 1998.

[118] C.-W. Hsu, C.-C. Chang, and L. Chih-Jen, “A Practical Guide to Support Vector Classification,” *BJU Int.*, vol. 101, no. 1, pp. 1396–400, 2010.

[119] K. R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, “An introduction to kernel-based learning algorithms,” *IEEE Trans. Neural Networks*, vol. 12, no. 2, pp. 181–201, 2001.

[120] A. Ding, X. Zhao, and J. Licheng, “Traffic Flow Time Series Prediction based on Statistics,” in *Proc. IEEE 5th Int. Conf. Intell. Transp. Syst.*, Singapore, 2002, pp. 727–730.

[121] C. H. Wu, J. M. Ho, and D. T. Lee, "Travel-time prediction with support vector regression," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 276–281, 2004.

[122] Y.-S. Jeong, Y.-J. Byon, M. M. Castro-Neto, and S. M. Easa, "Supervised Weighting-Online Learning Algorithm for Short-Term Traffic Flow Prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1700–1707, 2013.

[123] M. Lippi, M. Bertini, and P. Frasconi, "Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning," *Intell. Transp. Syst. IEEE Trans.*, vol. 14, no. 2, pp. 871–882, 2013.

[124] Y. Cong, J. Wang, and X. Li, "Traffic Flow Forecasting by a Least Squares Support Vector Machine with a Fruit Fly Optimization Algorithm," *Procedia Eng.*, vol. 137, no. 1, pp. 59–68, 2016.

[125] D. Den Hertog, J. Kleijnen, and A. Siem, "The correct Kriging variance estimated by bootstrapping," *Journal of the Operational Research Society*, vol. 57, no. 4, pp. 400–409, 2006.

[126] D. G. Krige, "A statistical approach to some basic mine problems on the Witwatersrand," *Journal Chem. Metall. Min. Soc. South Africa*, vol. 52, no. 6, pp. 119–139, 1951.

[127] G. Matheron, "Principles of Geostatistics." *Econ. Geol.*, vol. 58, no. 2, pp. 1246–1266, 1963.

[128] H. Braxmeier, V. Schmidt, and E. Spodarev, "Spatial Extrapolation of Anisotropic Road Traffic Data," *Image Anal. Stereol.*, vol. 23, no. 3, pp. 185–198, 2004.

[129] U. R. R. Manepalli and G. H. Bham, "Crash Prediction: Evaluation of Empirical Bayes and Kriging Methods," in *Proc. 3rd Int. Conf. Road Saf. Simul.*, Indianapolis, 2011.

[130] H. Liu, J. Shi, and E. Erdem, "Prediction of wind speed time series using modified Taylor Kriging method," *Energy*, vol. 35, no. 12, pp. 4870–4879, 2010.

[131] T. Hengl, G. B. M. Heuvelink, M. P. Tadić, and E. J. Pebesma, "Spatio-temporal prediction of daily temperatures using time-series of MODIS LST images," *Theor. Appl. Climatol.*, vol. 107, no. 1-2, pp. 265–277, 2012.

[132] D. Yang, C. Gu, Z. Dong, P. Jirutitijaroen, N. Chen, and W. M. Walsh, "Solar irradiance forecasting using spatial-temporal covariance structures and time-forward kriging," *Renew. Energy*, vol. 60, no. 1, pp. 235–245, 2013.

[133] D. Yang, Z. Dong, T. Reindl, P. Jirutitijaroen, and W. M. Walsh, "Solar irradiance forecasting using spatio-temporal empirical kriging and vector autoregressive models with parameter shrinkage," *Sol. Energy*, vol. 103, no. 1, pp. 550–562, 2014.

[134] S. Bhattacharjee and S. K. Ghosh, “Time-series augmentation of semantic kriging for the prediction of meteorological parameters,” in *Proc. 2015 IEEE Int. Geosci. Remote Sens. Symp.* Milan: IEEE, 2015, pp. 4562–4565.

[135] L. Pásztor, K. Z. Szabó, G. Szatmári, A. Laborczi, and Á. Horváth, “Mapping geogenic radon potential by regression kriging,” *Science of the Total Environment*, vol. 544, pp. 883–891, 2016.

[136] S. Biswas, S. Chakraborty, S. Chandra, and I. Ghosh, “Kriging based approach for estimation of vehicular speed and passenger car units on an urban arterial,” *Journal of Transportation Engineering, Part A: Systems*, vol. 143, no. 3, pp. 1–11, 2016.

[137] E. Mehdad and J. P. C. Kleijnen, “Classic Kriging versus Kriging with bootstrapping or conditional simulation: classic Kriging’s robust confidence intervals and optimization,” *Journal of the Operational Research Society*, vol. 66, no. 11, pp. 1804–1814, 2015.

[138] M. Franco-Villoria and R. Ignaccolo, “Bootstrap based uncertainty bands for prediction in functional kriging,” *Spatial Statistics*, vol. 21, no. 1, pp. 130–148, 2017.

[139] E. Tapoglou, G. P. Karatzas, I. C. Trichakis, and E. A. Varouchakis, “A spatio-temporal hybrid neural network-Kriging model for groundwater level simulation,” *Journal of Hydrology*, vol. 519, no. 2014, pp. 3193–3203, 2014.

[140] D. Liang and N. Kumar, “Time-space Kriging to address the spatiotemporal misalignment in the large datasets,” *Atmospheric Environment*, vol. 72, pp. 60–69, 2013.

[141] J. Berger, V. D. Oliveira, and B. Sansó, “Objective Bayesian analysis of spatially correlated data,” *Journal of the American Statistical Association*, vol. 96, no. 456, pp. 1361–1374, 2001.

[142] F. Fouedjio, N. Desassis, and J. Rivoirard, “A generalized convolution model and estimation for non-stationary random functions,” *Spatial Statistics*, vol. 16, no. 2016, pp. 35–52, 2016.

[143] C. G. Kaufman, M. J. Schervish, and D. W. Nychka, “Covariance tapering for likelihood-based estimation in large spatial data sets,” *Journal of the American Statistical Association*, vol. 103, no. 484, pp. 1545–1555, 2008.

[144] N. Cressie, G. Johannesson, S. Journal, R. Statistical, S. Series, and B. S. Methodology, “Fixed Rank Kriging for Very Large Spatial Data Sets,” *Journal of the Royal Statistical Society, Series B*, vol. 70, no. 1, pp. 209–226, 2008.

[145] M. Katzfuss, “Bayesian nonstationary spatial modeling for very large datasets,” *Environmetrics*, vol. 24, no. 3, pp. 189–200, 2013.

[146] S. Banerjee, A. E. Gelfand, and A. O. Finley, “Gaussian predictive process models for large spatial,” *Journal of Royal Statistical Society B*, vol. 70, no. 4, pp. 825–848, 2008.

[147] D. Nychka, S. Bandyopadhyay, D. Hammerling, and F. Lindgren, "A Multiresolution Gaussian Process Model for the Analysis of Large Spatial Datasets." *Journal of Computational and Graphical Statistics*, vol. 24, no. 2, pp. 579–599, 2015.

[148] T. Romary, "Incomplete Cholesky decomposition for the Kriging of large datasets," *Spatial Statistics*, vol. 5, no. 2013, pp. 85–99, 2013.

[149] J. M. Tadic, X. Qiu, V. Yadav, and A. M. Michalak, "Mapping of satellite Earth observations using moving window block kriging," *Geosci. Model Dev. Discuss*, vol. 8, pp. 3311–3319, 2015.

[150] L. Pronzato and M. J. Rendas, "Bayesian Local Kriging," *Technometrics*, vol. 59, no. 3, pp. 293–304, 2017.

[151] C. J. Paciorek and M. J. Schervish, "Spatial modelling using a new class of nonstationary covariance functions," *Environmetrics*, vol. 17, no. 5, pp. 483–506, 2006.

[152] P. Ma and E. L. Kang, "Fused Gaussian Process for Very Large Spatial Data," pp. 1–33, 2017.

[153] M. Katzfuss and D. Hammerling, "Parallel inference for massive distributed spatial data using low-rank models," *Statistics and Computing*, vol. 27, no. 2, pp. 363–375, 2017.

[154] M. Katzfuss, "A Multi-Resolution Approximation for Massive Spatial Datasets," *Journal of the American Statistical Association*, vol. 112, no. 517, pp. 201–214, 2017.

[155] J. Yin, S. H. Ng, and K. M. Ng, "Kriging metamodel with modified nugget-effect: The heteroscedastic variance case," *Computers and Industrial Engineering*, vol. 61, no. 3, pp. 760–777, 2011.

[156] J. K. Yamamoto, "Correcting the smoothing effect of ordinary kriging estimates," *Mathematical Geology*, vol. 37, no. 1, pp. 69–94, 2005.

[157] C. V. Deutsch, "Correcting for negative weights in ordinary kriging," *Computers and Geosciences*, vol. 22, no. 7, pp. 765–773, 1996.

[158] S. Bhattacharjee, P. Mitra, and S. K. Ghosh, "Spatial interpolation to predict missing attributes in GIS using semantic kriging," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 4771–4780, 2014.

[159] S. Bhattacharjee and S. K. Ghosh, "Performance Evaluation of Semantic Kriging: A Euclidean Vector Analysis Approach," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 6, pp. 1185–1189, 2015.

[160] L. Mihaylova, A. Hegyi, A. Gning, and R. K. Boel, "Parallelized particle and Gaussian sum particle filters for large-scale freeway traffic systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 36–48, 2012.

[161] C. Hermes, C. Wohler, K. Schenk, and F. Kummert, "Long-term vehicle motion prediction," in *Proc. 2009 IEEE Intelligent Vehicles Symposium, Xi'an*, 2009, pp. 652–657.

[162] C. Hermes, J. Einhaus, M. Hahn, C. Wöhler, and F. Kummert, "Vehicle tracking and motion prediction in complex urban scenarios," in *Proc. 2010 IEEE Intelligent Vehicles Symposium, San Diego, CA*, 2010, pp. 26–33.

[163] R. Herring, A. Hofleitner, P. Abbeel, and A. Bayen, "Estimating arterial traffic conditions using sparse probe data," in *Proc. 13th International IEEE Conference on Intelligent Transportation Systems*, 2010, pp. 929–936.

[164] J. Sau, N.-e. E. Faouzi, A. B. Aissa, and O. D. Mouzon, "Particle filter-based real-time estimation and prediction of traffic conditions Modeling framework," in *Proc. Appl. Stoch. Model. Data Anal. Int. Soc.*, Chania, Crete, 2007.

[165] H. Chen, H. A. Rakha, and S. Sadek, "Real-time freeway traffic state prediction: A particle filter approach," in *Proc. 2011 14th Int. IEEE Conf. Intell. Transp. Syst.*, Washington, DC, oct 2011, pp. 626–631.

[166] P. Cheng, Z. Qiu, and B. Ran, "Particle filter based traffic state estimation using cell phone network data," *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pp. 1047–1052, 2006.

[167] A. Hofleitner, R. Herring, and A. Bayen, "Arterial travel time forecast with streaming data: A hybrid approach of flow modeling and machine learning," *Transportation Research Part B: Methodological*, vol. 46, no. 9, pp. 1097–1122, 2012.

[168] A. Hofleitner, R. Herring, P. Abbeel, and A. Bayen, "Learning the dynamics of arterial traffic from probe data using a dynamic bayesian network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1679–1693, 2012.

[169] R. Wang, D. B. Work, and R. Sowers, "Multiple model particle filter for traffic estimation and incident detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3461–3470, 2016.

[170] R. Wang, S. Fan, and D. B. Work, "Efficient multiple model particle filtering for joint traffic state estimation and incident detection," *Transportation Research Part C: Emerging Technologies*, vol. 71, no. 1, pp. 521–537, 2016.

[171] H. Chen and H. A. Rakha, "Real-time travel time prediction using particle filtering with a non-explicit state-transition model," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 112–126, 2014.

[172] S. P. Hoogendoorn and P. H. L. Bovy, "State-of-the-art of vehicular traffic flow modelling," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 215, no. 4, pp. 283–303, 2001.

[173] T. T. Dang, H. Y. Ngan, and W. Liu, "Distance-based k-nearest neighbors outlier detection method in large-scale traffic data," in *International Conference on Digital Signal Processing, DSP.* IEEE, 2015, pp. 507–510.

[174] M. Hawes, H. M. Amer, and L. Mihaylova, "Traffic State Estimation via a Particle Filter with Compressive Sensing and Historical Traffic Data," in *Proc. 2016 19th Int. Conf. Inf. Fusion*, 2016, pp. 735–742.

[175] Y. Li, Z. Li, and L. Li, "Missing traffic data: comparison of imputation methods," *IET Intelligent Transport Systems*, vol. 8, no. 1, pp. 51–57, feb 2014.

[176] N. Cressie, "The Origins of Kriging," *Mathematical Geology*, vol. 22, no. 3, pp. 239–252, 1990.

[177] M. Oliver and R. Webster, "A tutorial guide to geostatistics: Computing and modelling variograms and kriging," *CATENA*, vol. 113, no. 2014, pp. 56–69, feb 2014.

[178] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[179] Z. Chen, "Bayesian Filtering : From Kalman Filters to Particle Filters , and Beyond," McMaster University, Canada, Tech. Rep., 2003.

[180] K. J. Offor, M. Hawes, and L. S. Mihaylova, "Short Term Traffic Flow Prediction with Particle Methods in the Presence of Sparse Data," in *Proc. 2018 21st International Conference on Information Fusion (FUSION)*, Cambridge, 2018, pp. 1205–1212.

[181] M. Hawes, H. M. Amer, and L. Mihaylova, "Traffic state estimation via a particle filter over a reduced measurement space," in *Proceedings on 20th International Conference on Information Fusion -.* Xi'an, China: IEEE, 2017.

[182] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerospace and Electronic Systems Magazine*, vol. 25, no. 7 PART 2, pp. 53–81, 2010.

[183] N. Mitrovic, M. T. Asif, J. Dauwels, and P. Jaillet, "Low-Dimensional Models for Compressed Sensing and Prediction of Large-Scale Traffic Data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2949–2954, 2015.

[184] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO - Simulation of Urban MObility - an Overview," in *Proceedings of the 3rd International Conference on Advances in System Simulation (SIMUL'11)*, Barcelona, Spain, 2011, pp. 63–68.

[185] L. Li, Y. Li, and Z. Li, "Efficient missing data imputing for traffic flow by considering temporal and spatial dependence," *Transportation Research Part C*, vol. 34, no. 2013, pp. 108–120, 2013.

[186] B. Bae, H. Kim, H. Lim, Y. Liu, L. D. Han, and P. B. Freeze, “Missing data imputation for traffic flow speed using spatio-temporal cokriging,” *Transportation Research Part C: Emerging Technologies*, vol. 88, no. March, pp. 124–139, 2018.

[187] K. J. Offor, L. Vaci, and L. S. Mihaylova, “Traffic Estimation for Large Urban Road Network with High Missing Data Ratio,” *Sensors*, vol. 19, no. 12, p. 2813, 2019.

[188] F. Crawford, D. Watling, and R. Connors, “A statistical method for estimating predictable differences between daily traffic flow profiles,” *Transportation Research Part B: Methodological*, vol. 95, no. November, pp. 196–213, 2017.

[189] B. A. Konomi, H. Sang, and B. K. Mallick, “Adaptive Bayesian Nonstationary Modeling for Large Spatial Datasets Using Covariance Approximations,” *Journal of Computational and Graphical Statistics*, vol. 23, no. 3, pp. 802–829, 2014.

[190] J.-m. Chiou, Y.-c. Zhang, W.-h. Chen, and C.-w. Chang, “A functional data approach to missing value imputation and outlier detection for traffic flow data,” *Transportmetrica B : Transport Dynamics*, vol. 0566, no. 2, pp. 106–129, 2014.

[191] I. G. Guardiola, T. Leon, and F. Mallor, “A functional approach to monitor and recognize patterns of daily traffic profiles,” *Transportation Research Part B*, vol. 65, no. 2014, pp. 119–136, 2014.

[192] D. Ginsbourger, C. Helbert, and L. Carraro, “Discrete mixtures of kernels for Kriging-based optimization,” *Quality and Reliability Engineering International*, vol. 24, no. 6, pp. 681–691, 2008.

[193] P. Wang, Y. Kim, H. Yang, and L. Mihaylova, “Short-Term Traffic Prediction with Vicinity Gaussian Processes in the Presence of Missing Data,” in *in Proc. 2018 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, Bonn, 2018, pp. 1–6.

[194] T. V. Le, R. J. Oentaryo, S. Liu, and H. C. Lau, “Local Gaussian Processes for Efficient Fine-Grained Traffic Speed Prediction,” *IEEE Transactions on Big Data*, vol. 3, no. 2017, pp. 194–207, 2017.

[195] J.-P. Chilès and P. Delfiner, *Geostatistics : modeling spatial uncertainty*, 2nd ed., P. Delfiner, Ed. New York: John Wiley & Sons, 2012.

[196] P. H. Garthwaite and E. Mubwandarikwa, “Selection of weights for weighted model averaging,” *Australian and New Zealand Journal of Statistics*, vol. 52, no. 4, pp. 363–382, 2010.

[197] P. Comon, “Independent component analysis, A new concept?” *Signal Processing*, vol. 36, pp. 287–314, 1994.

[198] A. D’Aspremont, L. El Ghaoui, M. Jordan, and G. Lanckriet, “A Direct Formulation for Sparse PCA Using Semidefinite Programming,” *SIAM Review*, vol. 49, no. 3, pp. 434–448, 2007.

[199] A. Cichocki and A. H. Phan, "Fast local algorithms for large scale nonnegative matrix and tensor factorizations," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E92-A, no. 3, pp. 708–721, 2009.

[200] H. Kim and H. Park, "Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis," *Bioinformatics*, vol. 23, no. 12, pp. 1495–1502, 2007.

[201] ——, "Non-negative Matrix Factorization Based on Alternating Non-negativity Constrained Least Squares and Active Set Method," *SIAM J. Matrix Aanal. Appl*, vol. 30, no. 2, pp. 713–730, 2008.

[202] J. Kim, Y. He, and H. Park, "Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework," *Journal of Global Optimization*, vol. 58, no. 2, pp. 285–319, 2014.

[203] D. D. Lee and H. S. Seung, "Algorithms for Non-negative Matrix Factorization," *Nips*, no. 1, 2000.

[204] N. Gillis, "The Why and How of Nonnegative Matrix Factorization," *arXiv:1401.5226v2 [stat.ML]*, pp. 1–25, 2014. [Online]. Available: http://arxiv.org/abs/1401.5226

[205] C. Boutsidis and E. Gallopoulos, "SVD based initialization: A head start for nonnegative matrix factorization," *Pattern Recognition*, vol. 41, no. 4, pp. 1350–1362, 2008.

[206] J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Metagenes and molecular pattern discovery using matrix factorization," *Proceedings of the National Academy of Sciences*, vol. 101, no. 12, pp. 4164–4169, mar 2004.

[207] "EU SETA Project," 2018. [Online]. Available: http://setamobility.weebly.com/

[208] *Aimsun 8 Users' Manual*, TSS - Transport Simulation Systems, 2014. [Online]. Available: https://www.scribd.com/document/266245543/Aimsun-Users-Manual-v8

[209] L. Pronzato, "Minimax and maximin space-filling designs : some properties and methods for construction," *Journal de la Société Française de Statistique*, vol. 158, no. 1, pp. 7–36, 2017.