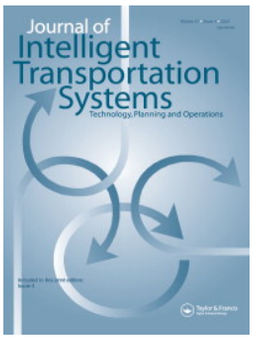


Taylor & Francis
Taylor & Francis Group

Journal of Intelligent Transportation Systems
Technology, Planning, and Operations

ISSN: 1547-2450 (Print) 1547-2442 (Online) Journal homepage: www.tandfonline.com/journals/gits20

Traffic state prediction using conditionally Gaussian observed Markov fuzzy switching model

Zied Bouyahia, Hedi Haddad, Stéphane Derrode & Wojciech Pieczynski

To cite this article: Zied Bouyahia, Hedi Haddad, Stéphane Derrode & Wojciech Pieczynski (2023) Traffic state prediction using conditionally Gaussian observed Markov fuzzy switching model, Journal of Intelligent Transportation Systems, 27:4, 503-522, DOI: 10.1080/15472450.2022.2069498

To link to this article: https://doi.org/10.1080/15472450.2022.2069498

JOURNAL OF INTELLIGENT TRANSPORTATION SYSTEMS
2023, VOL. 27, NO. 4, 503–522
https://doi.org/10.1080/15472450.2022.2069498

Traffic state prediction using conditionally Gaussian observed Markov fuzzy switching model

Zied Bouyahia[a], Hedi Haddad[a], Stéphane Derrode[b], and Wojciech Pieczynski[c]

[a]College of Arts and Applied Sciences, Dhofar University, Salalah, Oman; [b]Université de Lyon, CNRS, École Centrale de Lyon, LIRIS, CNRS UMR, Lyon, France; [c]Institut Polytechnique de Paris, Telecom SudParis, SAMOVAR, Paris, France

ABSTRACT

Efficient and accurate prediction of traffic state plays a major role in the implementation of effective intelligent transportation systems. Therefore, traffic forecasting has been attracting a significant interest over the last decades striving to efficiently achieve highest accuracy of reliable prediction algorithms. In this paper, we present a novel prediction algorithm based on the Conditionally Gaussian Observed Markov Fuzzy Switching Model (CGOMFSM). The proposed scheme relies on a triplet representation of traffic encompassing traffic flow, speed and a switch process in order to infer parameters that govern the dynamics of traffic. This work investigates the impact of explicit incorporation of fuzzy switching processes on the accuracy of traffic data prediction. It aims to overcome the shortcomings of several existing prediction schemes in which the crisp modeling of traffic dynamics is hindering the effectiveness of prediction. The experimental study shows that the proposed algorithm yields satisfactory results for a prediction horizon up to 60 minutes for data collected at regular 15-minutes intervals.

ARTICLE HISTORY
Received 16 September 2020
Revised 12 April 2022
Accepted 20 April 2022

KEYWORDS
Fuzzy switching linear state-space model; prediction; short-term traffic data

1. Introduction

Accurate and timely prediction of the short-term traffic state is of a paramount importance for both road users and traffic control authorities as it can help providing guidance for travelers and proactively managing traffic congestion. The term traffic state usually refers to the condition of traffic at a given time and location, and is usually described by a collection of variables or parameters such as flow, speed and density (Antoniou et al., 2013). Traffic state prediction is a challenging task because traffic is intrinsically a very complex phenomenon that emerges from the interactions between other complex phenomena and entities, such as drivers' behaviors, road users' mobility patterns, road infrastructures and settings, weather conditions and sporadic events (road accidents, social events, etc.). Because it is not possible to capture all these aspects into a unique model, hundreds of traffic state prediction approaches, models and algorithms have been proposed over the last four decades. The majority of the proposed traffic prediction solutions focus on very specific traffic aspects, such as traffic non-linearity, traffic seasonality, traffic spatio-temporal correlation, traffic fluctuation and traffic congestion propagation. For example, (i) non-parametric short-term traffic prediction models only capture traffic's non-linearity, (ii) spatio-temporal short-term traffic prediction models only address the spatio-temporal correlation between traffic conditions, (iii) time-series models only address traffic seasonality, etc. Predicting how traffic conditions change over time is one aspect that has attracted the attention of transportation researchers. The proposed models are inspired by previous works on the general problem of modeling and analyzing changing data, which has been studied in different research communities using different terminologies, such as concept drift in Machine Learning (Widmer & Kubat, 1996), covariate shift in pattern recognition (Moreno-Torres et al., 2012) and nonstationarity in signal processing (Haykin & Li, 1995). One of the most widely used models for capturing and analyzing changes in traffic state conditions are based on the concept of regime-switching in time series. Basically, the idea consists in grouping traffic states (variables) into regions called regimes that reflect homogeneous groups of variables with similar characteristics (Antoniou et al., 2013). From this standpoint, traffic is conceptually modeled as a

CONTACT Zied Bouyahia zbouyahia@du.edu.om College of Arts and Applied Sciences, Dhofar University, Salalah, P. Code. 211, Oman.

process that evolves from one regime to another, where every regime might correspond to intuitive information about traffic dynamics, such as "free flow," "congested," "severely congested," etc. Regime-switching models have been introduced in the traffic state of the art to depict how traffic conditions change from one regime to another. Different approaches have been used for modeling traffic regimes and the nonlinearity underlying their transitions, including clustering (Antoniou et al., 2013; L. Sun & Zhou, 2005), Smooth-Transition Regression (Kamarianakis et al., 2010), etc. Relatively few works have been interested to the prediction of traffic regime-switches (Cetin & Comert, 2006; Kamarianakis et al., 2012; Z. Li et al., 2014; Lu et al., 2020; Qi & Ishak, 2014; Yu & Zhang, 2004) and all of them concluded that regime-switching models may improve the accuracy of short-term traffic prediction (Pavlyuk, 2017). In this paper we study the effect of explicitly modeling gradual transitions between traffic regimes on the accuracy of traffic state prediction. Unlike previous works that added intermediate regimes between free-flow and congested traffic states (Lu et al., 2020), we explicitly model transitions between two traffic regimes as an interpolation of the hard states corresponding to the boundary traffic regimes. We propose a new fuzzy switching regime model to capture the gradual transitions of the traffic dynamics. We show that smoothing transitions using a fuzzy model considerably improves traffic prediction accuracy compared to hard transitions. To the best of our knowledge, the use of fuzzy regime switching models for traffic state prediction has not been previously explored in the state of the art.

The contributions of this paper are summarized as follows:

- We propose a new fuzzy regime-switching model for traffic flow prediction based on the conditionally Gaussian observed Markov fuzzy switching model (CGOMFSM, Bouyahia et al., 2020) and we devise a new EM-based algorithm to estimate the parameters of the CGOMFSM.
- Rather than using only one traffic feature to predict traffic state, we use a triplet representation involving: a pairwise process (traffic flow and average speed) as well as an auxiliary process referring to traffic state. Therefore, the traffic condition is modeled by a stochastic process which allows for an explicit representation of traffic state transitions.

The remainder of this paper is organized as follows. First, we present an overview of the relevant state-of-the-art approaches for traffic data prediction. Then, in Section 3, we give a brief review of the CGOFMSM and depict the novel prediction algorithm based on it. Section 4 details the semi-supervised model parameters' estimation scheme that we propose based on the Expectation-Maximization principle. Section 5 reports the experimental study conducted to assess the interest of the proposed CGOMFSM-predictor.

2. Related works

Beside the early naive approaches (Smith et al., 2002), traffic prediction models can be broadly classified as parametric, non-parametric, or hybrid. Main parametric models include the ARIMA family (Ahmed & Cook, 1979; Lee & Fambro, 1999) (including SARIMA (Kumar & Vanajakshi, 2015; Williams & Hoel, 2003), KARIMA (Van Der Voort et al., 1996), STARIMA (Duan et al., 2019; Min & Wynter, 2011), VARMA (Kamarianakis & Prastacos, 2003) and ARIMAX (Williams, 2001)) and Markov chains (Mei et al., 2015; Qi & Ishak, 2014) (including linear dynamical systems (Mei et al., 2015)). Other parametric methods are based on linear regression (H. Sun et al., 2003) or Gaussian processes (Xie et al., 2010; J. Zhao & Sun, 2016). The main advantages of parametric models are their relatively ease of implementation and low computational costs and complexity, their ability to provide explicit theoretical interpretability of the traffic dynamics, and to perform over limited-size datasets (Mei et al., 2013). However, they usually require high quality of accurate and stable datasets, which are not common characteristics of traffic data. Indeed, traffic dynamics are stochastic and unstable (Song et al., 2019), and parameters training is vulnerable to incomplete data (Mei et al., 2015). Moreover, parametric models usually fail to capture the non-linearity of traffic dynamics (Nagy & Simon, 2018). To address these limitations, non-parametric methods, also called data-driven, have been gaining increasing interest over parametric models, including artificial neural networks models (Hu et al., 2019), deep learning models (L. Li et al., 2019) and the support vector regression (SVR) models (Wei & Liu, 2013), to mention few. Their main advantage over parametric models is their ability to model dynamic and non-linear traffic aspects. However, they require huge training datasets in order to achieve good prediction performance, and they fail to predict "unseen" traffic incidents, because only incidents that are already observed in the training datasets can be predicted. Given that each prediction approach has its own advantages and limitations, and that there is no one model suitable for all situations,

hybrid methods that combine merits of different parametric and non-parametric approaches have been widely used in the recent years to improve the prediction accuracy. Hybrid models include Bayesian-neural networks (Zheng et al., 2006), fuzzy rule-based models (Zhang & Ye, 2008), chaos-wavelet analysis-support vector machine (J. Wang & Shi, 2013), ARFIMA-NAR neural networks (Xu et al., 2021), and wavelet transformation combined with artificial neural networks (Mousavizadeh Kashi & Akbarzadeh, 2019). Hybrid prediction models are generally better than pure prediction ones in terms of prediction accuracy (C. Wang & Ye, 2016).

Most of the proposed models traditionally address only the temporal change of the traffic state variables and ignore their spatial dynamics. Recently, spatio-temporal short traffic prediction models have been gaining an increasing interest in the research community to improve the accuracy of traffic prediction by taking into consideration spatial and temporal correlations between traffic conditions of neighbor road segments. Different parametric, non-parametric and hybrid models have been devised in this regard, including temporal graph convolutional neural networks (Zhao et al., 2019), Spatio-Temporal Random Effects models (Wu et al., 2016), multivariate models (Zhang & Zhang, 2016), etc. A recent review of spatio-temporal traffic prediction models can be found in Ermagun and Levinson (2018).

From a conceptual perspective, most of the above-mentioned models address the problem of traffic state prediction, where the state is described by a number of numerical and continuous variables/parameters, such as flow, density and speed, leading to an infinite number of traffic states (Antoniou et al., 2013). Other works have been interested in modeling and predicting how do short-term classes or categories of traffic conditions change, by attempting to describe the dynamics governing traffic conditions' evolution using high-level and discrete stages commonly called regimes. As stated above, traffic is switching from one regime to another, and thus the key problem is to identify regimes over time.

Let us stress that the regime-switching traffic models introduced in the state-of-the-art are issued from time-series switching models initially developed in machine learning (Bergmeir et al., 2012). In machine learning, switching models (Chiappa, 2014), also called piecewise linear models, capture the time-variability in a time-series model by splitting a sequence of time-stamped observations into non-overlapping segments or regions, each with fixed model parameters. In the literature, regime-switching models have been referred to using different terms, such as switching state-space models, change-point regression models and segmented regression models (Kamarianakis et al., 2010). But, in general, two complementary approaches have been used to analyze changes in time-series: regime-switching and change-points (Cabrieto et al., 2018). Regime switching models use an additional stochastic process taking its values in a finite set of switches, or regimes. The regime changes aim to capture changes in the underlying studied phenomenon through the observed time period (Lange & Rahbek, 2009). Markov switching (MS) are the prevalent regime switching models in the state of the art, such as MS ARCH and MS GARCH models (Lange & Rahbek, 2009). In contrast, change point detection algorithms aim to identify the boundaries of the identified segments (regimes), given the observed time series and an underlying predictive model from which the segments of observations are generated (Agudelo-España et al., 2019). Change points are sudden variations in time series data that may correspond to transitions between different regimes. Time series' change points detection is commonly used in a variety of applications such as speech and image analysis, human activity analysis and medical condition monitoring, to mention a few (Aminikhanghahi & Cook, 2017). Different change-point models and methods have been proposed in machine learning using both supervised (SVM: Support Vector Machine, Hidden Markov Model, Gaussian Mixture Model, etc.) and non-supervised approaches (Gaussian Process, CUSUM, Kernel-Based model, Bayesian models, etc.); a complete review can be found in Aminikhanghahi and Cook (2017). Let us mention that change-point and regime-switching models are complementary in the sense that change-point refers to the point in time where the change takes place and regime-switch refers to the occurrence of a different regime after the change point.

Relatively few hybrid change-point and regime-switching time series methods have been used in the transportation state-of-the-art for the prediction of traffic changes. Authors in Cetin and Comert (2006) combined expectation-maximization (EM) and cumulative summation (CUSUM) algorithms in an ARIMA model to update the estimated mean of the time series. A Markov-switching ARIMA model is used in Yu and Zhang (2004). Hidden Markov Models (HMMs) were used in Qi and Ishak (2014) to define traffic states using Mean and Contrast of speed observations. State transition probabilities are used to model the

stochastic evolution of traffic conditions. Five states of contrast speed values were identified to indicate small, moderate, and rapid traffic speed changes. The trained HMMs were used to estimate the most likely sequence of traffic states that corresponds to a given sequence of traffic speed observations. Similar approach was used in Li et al. (2014). Authors in Comert and Bezuglov (2013) combined a HMM with a change-point detection method in order to estimate ARIMA model's parameters. Threshold regressions regime-switching were used in Kamarianakis et al. (2012) for traffic speed prediction.

Even though these works have concluded that regime-switching/change-point models may improve the accuracy of short-term traffic prediction, capturing and predicting how traffic dynamics transit between different regimes has not received too much attention in the literature. Moreover, the existing models were mainly introduced to predict rapid change in traffic conditions caused by abnormal events, such as accidents, and consequently the transitions between regimes are supposed to happen abruptly. However, in reality, traffic dynamics can switch from one state/regime to another at different speeds. As mentioned above, traffic congestion may quickly happen after a road accident, but it will take more time to gradually transit from the congested to the free state. To the best of our knowledge, studying how capturing such a gradual regime-switching may improve traffic prediction accuracy has not been considered before. In this paper we propose a new prediction algorithm, based on the recent CGOFMSM model, that allows for capturing gradual transitions between traffic regimes.

3. Stationary CGOMSM and fuzzy prediction

For several decades, the large family of HMMs has demonstrated its interest in many applications related to the field of classification or filtering of time series of data. The huge success of HMMs has spurred innumerable research works attempting to improve the straightforward aspect of HMMs by extending them (see Yu, 2015). Since the Rabiner's paper (Rabiner, 1989), numerous extensions of Markov models have been derived, incorporating additional features to overcome both the linearity and the Gaussianity constraints of the initial modeling. We can also cite the extended Kalman filter, the unscented Kalman filter, the particle filter, etc. Another track that has appeared in the literature is to approximate a given non-linear non-Gaussian system by a switching Gaussian system. In the standard switching models, e.g., in jump Markov linear systems (JMLSs), there is no known fast exact optimal filtering algorithm (Doucet & Andrieu, 2001).

However, there are some recent switching models in which fast exact optimal filtering is computationally feasible, e.g., the conditionally Markov switching hidden linear model (CMSHLM (Pieczynski, 2011)) and the conditionally Gaussian observed Markov switching model (CGOMSM (Abbassi et al., 2015)). The ability of these models to approximate non-linear and non-Gaussian systems has been illustrated through experiments with stochastic volatility data in Gorynin et al. (2017a). However, these models only take into account salient and crisp transitions between switches, and, to the best of our knowledge, only a few body of research work has been devoted to proposing fuzzy variants of JMLSs. In this regard, the discrete-time Takagi-Sugano (T.-S.) approach to fuzzy filter design for Markov jump models has been gaining substantial attention over the last few years (Xie et al., 2017), especially in the fuzzy control and fault detection research community. Another fuzzy model, extending the CGOMSM and named "Conditionally Gaussian Observed Markov Fuzzy Switching Model" (CGOMFSM), has been recently proposed (Bouyahia et al., 2020). Unlike the T.-S. model, CGOMFSM assumes that, conditionally to switches, the model is pairwise linear, which enriches the modeling capacities of the classical linear models. In a previous work, we proposed a new traffic state estimation scheme based on the CGOMFSM, and the results were promising. The proposed scheme relies on assuming that there exist an a priori distribution of the switches process. In this paper we propose a novel traffic state predictor based on the CGOFMSM, in which the assumption of known a priori distribution of the switch process used in Bouyahia et al. (2021) is relaxed. Therefore, in this paper we propose to use new prediction and parameters estimation algorithms that are different from those used in Bouyahia et al. (2021). Indeed, the implicit fuzzy switching model presented in Bouyahia et al. (2021) was used for traffic state inference from recorded speed measures with an EM algorithm that assimilates the transitory states as mixtures of the boundary hard switches 0 and 1. The model we propose in this paper refers to intermediary switches using a discretization of the interval [0,1] with distinct parameters. As discussed later, the proposed fuzzy predictor allows for accurate calculations of the predicted variables, up to numerical approximations of some integrals.

3.1. Modeling traffic state as a stationary CGOMSM

Classically, a conditionally Gaussian observed Markov switching model is defined by a triplet $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N, \boldsymbol{R}_1^N)$.

- $\boldsymbol{X}_1^N = \{X_1, \ldots, X_N\}$ denotes a real-valued stochastic process, representing the (hidden) traffic flow during a period of N samples;
- $\boldsymbol{Y}_1^N = \{Y_1, \ldots, Y_N\}$ denotes a real-valued stochastic process, representing the (observed) time-average traffic speed during the same period of time.
- The discrete process $\boldsymbol{R}_1^N = \{R_1, \ldots, R_N\}$ represents the regime switching; for all $n \in [1, N]$, $R_n \in \Omega = \{1, \ldots, K\}$.

In the stochastic setting considered here, a prediction consists in estimating the mean and covariance of the density $p(x_{n+\mathcal{W}}|\boldsymbol{y}_1^n)$, where $\mathcal{W} \geq 1$ is the horizon of the prediction.

Let us denote $\boldsymbol{Z}_n = (X_n, Y_n)^\top$, $\boldsymbol{T}_n = (X_n, Y_n, R_n)^\top$, and assume the following:

$$\boldsymbol{T}_1^N = (\boldsymbol{T}_1, ..., \boldsymbol{T}_N) \text{ is Markov, with} \quad (1)$$

$$p(r_{n+1}|x_n, r_n, y_n) = p(r_{n+1}|r_n),\ n = 1, ..., N-1, \quad (2)$$
(which implies the Markovianity of $\boldsymbol{R}_1^N$);

$$\boldsymbol{Z}_1^N = (\boldsymbol{Z}_1, ..., \boldsymbol{Z}_N) \text{ is Gaussian conditional on } \boldsymbol{R}_1^N. \quad (3)$$

Such a model is called "Conditionally Gaussian Markov Switching Model" (CGMSM). Assuming that the model is stationary, i.e., margins and transition parameters do not depend on n, the CGMSM can be written

$$\boldsymbol{Z}_{n+1} = \boldsymbol{A}(\boldsymbol{r}_n^{n+1})\boldsymbol{Z}_n + \boldsymbol{B}(\boldsymbol{r}_n^{n+1})\boldsymbol{W}_{n+1} + \boldsymbol{N}(\boldsymbol{r}_1^2) \quad (4)$$

for all $n \in [1, ..., N-1]$, and where

- Conditionally to $R_1 = r_1$, the random vector $\boldsymbol{Z}_1$ is distributed according to a Gaussian distribution with assumed known parameters.
- Vectors $\boldsymbol{W}_n = (U_n, V_n)^\top$, with U_n and V_n denote mutually independent Gaussian random variable with zero mean and identity covariance matrices. $\boldsymbol{W}_n$ vectors are also assumed mutually independent.
- For all $r_1, r_2 \in \Omega$, vectors $\boldsymbol{N}(\boldsymbol{r}_1^2)$ are mean vectors given by

$$\boldsymbol{N}(\boldsymbol{r}_1^2) = \boldsymbol{M}(r_2) - \boldsymbol{A}(\boldsymbol{r}_1^2)\boldsymbol{M}(r_1)$$

where $\boldsymbol{M}(r_1) = \begin{bmatrix} M^X(r_1) \\ M^Y(r_1) \end{bmatrix}$. Note that we have $\mathbb{E}\left[\begin{pmatrix} X_n \\ Y_n \end{pmatrix} | R_n = r_n\right] = \boldsymbol{M}(r_n)$ for all $n \in [1, ..., N]$.

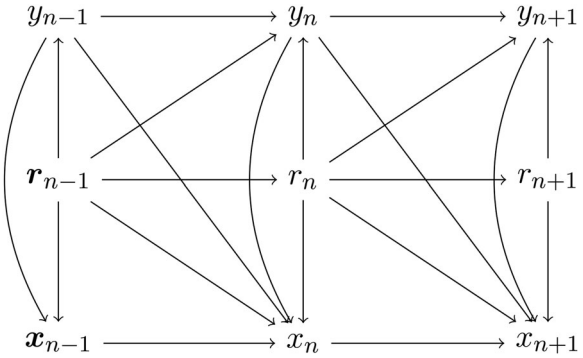


Figure 1. Dependency graph of CGOMFSM.

- For all $r_1, r_2 \in \Omega$, matrices $\boldsymbol{A}(\boldsymbol{r}_1^2)$ and $\boldsymbol{B}(\boldsymbol{r}_1^2)$ are denoted by

$$\boldsymbol{A}(\boldsymbol{r}_1^2) = \begin{bmatrix} a^1(\boldsymbol{r}_1^2) & a^2(\boldsymbol{r}_1^2) \\ a^3(\boldsymbol{r}_1^2) & a^4(\boldsymbol{r}_1^2) \end{bmatrix}$$
$$\boldsymbol{B}(\boldsymbol{r}_1^2) = \begin{bmatrix} b^1(\boldsymbol{r}_1^2) & b^2(\boldsymbol{r}_1^2) \\ b^3(\boldsymbol{r}_1^2) & b^4(\boldsymbol{r}_1^2) \end{bmatrix}$$

with

$$\boldsymbol{B}(\boldsymbol{r}_1^2)\boldsymbol{B}^\top(\boldsymbol{r}_1^2) = \begin{bmatrix} \gamma^1(\boldsymbol{r}_1^2) & \gamma^2(\boldsymbol{r}_1^2) \\ \gamma^3(\boldsymbol{r}_1^2) & \gamma^4(\boldsymbol{r}_1^2) \end{bmatrix}$$

The CGMSM assumes that, conditionally to switches, the model is pairwise linear, which enriches its traditional linear counterpart.

The CGOMSM is a particular CGMSM, obtained by setting

$$a^3(\boldsymbol{r}_1^2) = 0 \quad (5)$$

for each $\boldsymbol{r}_1^2 \in \Omega^2$. It has been shown that the CGOMSM allows for exact recursive equations for filtering (Gorynin et al., 2017a) and smoothing (Gorynin et al., 2017b). This is mainly due to the fact that in CGOMSM the pair $(\boldsymbol{R}_1^N, \boldsymbol{Y}_1^N)$ is Markov. Figure 1 illustrates the dependencies between the stochastic processes $\boldsymbol{X}_1^N$, $\boldsymbol{Y}_1^N$ and $\boldsymbol{R}_1^N$.

3.2. Fuzzy prediction

Formally, prediction consists in computing $\mathbb{E}[\boldsymbol{Z}_{n+1}|\boldsymbol{y}_1^n]$. This computation is specified in the general framework of CGOMSM in Gorynin et al. (2017b). In CGOMFSM, $\boldsymbol{R}_1^N$ is a fuzzy Markov chain. Each R_n takes its values in $[0, 1]$, with $R_n = 0$ standing for "fluid traffic," $R_n = 1$ for "congested traffic," and $R_n \in]0, 1[$ for a density of traffic between the two. The originality of CGOMFSM is based on the hybrid shape of the distribution of R_n, made up from two

Dirac masses at 0 and 1, and a continuous part on $]0,1[$ specified by a classic (i.e., Lebesgue) density. Then by noting $\mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Gamma}_0)$, $\mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Gamma}_1)$ the Gaussian distributions $p(\boldsymbol{z}_n|r_n = 0)$ and $p(\boldsymbol{z}_n|r_n = 1)$ respectively, $p(\boldsymbol{z}_n|r_n \in [0,1])$ is assumed to be also Gaussian with a distribution denoted by $\mathcal{N}(\boldsymbol{\mu}_{r_n}, \boldsymbol{\Gamma}_{r_n})$. The simplest way, but not the only one, is to specify the parameters of $p(\boldsymbol{z}_n|r_n \in [0,1])$ by linear interpolation of the parameters of the two initial distributions according to

$$\begin{aligned}\boldsymbol{\mu}_{r_n} &= (1 - r_n)\,\boldsymbol{\mu}_0 + r_n\,\boldsymbol{\mu}_1 \\ \boldsymbol{\Gamma}_{r_n} &= (1 - r_n)\,\boldsymbol{\Gamma}_0 + r_n\,\boldsymbol{\Gamma}_1\end{aligned}$$

However, while computing different quantities of interest, one arrives at integrals on $[0,1]$ which are impossible to compute exactly and, therefore, approximations are required. Given that parameter estimation is a hard problem in the general case, we simplify it by making the following approximation: we assume that $[0,1]$ is divided into $F+2$ "discrete fuzzy levels": $0, \frac{1}{F+1}, \ldots, \frac{F}{F+1}, 1$. Thus the distribution of R_n is a distribution on

$$\tilde{\Omega} = \left\{0, \frac{1}{F+1}, \ldots, \frac{F}{F+1}, 1\right\}$$

The approximated model will be denoted by CGOMFSM^F, where F is the number of fuzzy levels in the approximation. The impact of F on prediction accuracy is discussed in the experimental study. However, let us insist that the approximate CGOMFSM so obtained is not a classic CGOMSM. Indeed, there are two Gaussian distributions in CGOMFSM, while there would be $F+2$ Gaussian distributions in CGOMSM.

With respect to the traffic prediction problem, let us assume, as already indicated, that the process $\boldsymbol{X}_1^n$ models the traffic flow and the process $\boldsymbol{Y}_1^n$ models the traffic speed until time index n. We attempt to predict the traffic data (speed and flow) at time index $n+1$ from the historic speed data only. In the stochastic framework considered here, the prediction consists in computing $\mathbb{E}\left[\boldsymbol{Z}_{n+1}|\boldsymbol{y}_1^n\right]$, which is the expected values of the traffic speed Y_{n+1} and traffic flow X_{n+1} at time index $n+1$ from past observations $\boldsymbol{y}_1^n$. Formally, using the discrete scheme specified above, this stands for computing

$$\mathbb{E}\left[\boldsymbol{Z}_{n+1}|\boldsymbol{y}_1^n\right] = \sum_{\boldsymbol{r}_n^{n+1} \in \tilde{\Omega}^2} \mathbb{E}\left[\boldsymbol{Z}_{n+1}|\boldsymbol{r}_n^{n+1}, \boldsymbol{y}_1^n\right] p\left(\boldsymbol{r}_n^{n+1}|\boldsymbol{y}_1^n\right) \quad (6)$$

Let us specify how these quantities can be computed:

- The term $\mathbb{E}\left[\boldsymbol{Z}_{n+1}|\boldsymbol{r}_n^{n+1}, \boldsymbol{y}_1^n\right]$ can be calculated using (4) by

$$\mathbb{E}\left[\boldsymbol{Z}_{n+1}|\boldsymbol{r}_n^{n+1}, \boldsymbol{y}_1^n\right] = \boldsymbol{A}(\boldsymbol{r}_n^{n+1})\mathbb{E}\left[\boldsymbol{Z}_n|r_n, \boldsymbol{y}_1^n\right] + \boldsymbol{N}(\boldsymbol{r}_1^2) \quad (7)$$

  with

$$\mathbb{E}\left[\boldsymbol{Z}_n|\boldsymbol{r}_n^{n+1}, \boldsymbol{y}_1^n\right] = \begin{bmatrix}\mathbb{E}\left[X_n|r_n, \boldsymbol{y}_1^n\right] \\ y_n\end{bmatrix}$$

  since R_{n+1} and $\boldsymbol{Z}_n$ are independent conditionally on R_n. The term $\mathbb{E}\left[X_n|r_n, \boldsymbol{y}_1^n\right]$ can be computed recursively, following (Gorynin et al., 2017a).
- The second term can be rewritten as $p(\boldsymbol{r}_n^{n+1}|\boldsymbol{y}_1^n) = p(r_{n+1}|r_n)p(r_n|\boldsymbol{y}_1^n)$, where $p(r_n|\boldsymbol{y}_1^n)$ are the filtered posterior probabilities of jumps, whose recursive calculation is detailed in Gorynin et al. (2017b).

Hence, it is possible to predict both traffic flow $\mathbb{E}\left[X_{n+1}|\boldsymbol{y}_1^n\right]$ and traffic speed $\mathbb{E}\left[Y_{n+1}|\boldsymbol{y}_1^n\right]$ in a recursive way by taking margins of (6). In this paper, we will only focus on the prediction of the traffic flow. The main reason for choosing traffic flow over speed is that, unlike the latter, traffic volume shows less volatility and varies gradually over sufficiently long periods of time. Therefore, traffic flow demonstrates better the potential of the fuzzy switching model.

While not developed here, and with similar calculations as the ones above, it is also possible to compute the predicted covariance matrix $\mathrm{Cov}\left[\boldsymbol{Z}_{n+1}|\boldsymbol{y}_1^n\right]$, allowing to get a confidence measure on the two predictions.

To achieve prediction for higher horizons, we forecast the next horizon using the previously predicted values. More precisely, (7) implies

$$\mathbb{E}\left[\boldsymbol{Z}_{n+k+1}|\boldsymbol{r}_{n+k}^{n+k+1}, \boldsymbol{y}_1^n\right] = \boldsymbol{A}\left(\boldsymbol{r}_{n+k}^{n+k+1}\right)\mathbb{E}[\boldsymbol{Z}_{n+k}|r_{n+k}, \boldsymbol{y}_1^n]\; p(r_{n+k+1}|r_{n+k})\, p\left(r_{n+k}|\boldsymbol{y}_1^n\right)$$

which gives $\mathbb{E}\left[X_{n+k+1}|\boldsymbol{r}_{n+k}^{n+k+1}, \boldsymbol{y}_1^n\right]$.

4. EM-based semi-supervised parameter estimation

In order to make the best possible prediction, we need to learn the parameters of the model that best suit the data and the intended application. For this purpose, we next propose a semi-supervised learning method based on the EM principle and on a stochastic variant. In any case, the method remains valid for multidimensional variables, even if matrix calculations have still to be conducted.

The algorithm depicted below is referred to as "semi-supervised" because we consider that we have a partial training set of M data $\boldsymbol{z}_1^M = (\boldsymbol{x}_1^M, \boldsymbol{y}_1^M)$ available, but we do not know the corresponding fuzzy switches $\boldsymbol{r}_1^M$.

Instead of estimating the parameters in (4), it is simpler to use parameters from the following equivalent parametrization (see Gorynin et al., 2017a, 2017b).

$$\begin{aligned} X_{n+1} &= \mathcal{A}_{r_n}^{r_{n+1}} X_n + \mathcal{B}_{r_n}^{r_{n+1}} Y_n + \mathcal{C}_{r_n}^{r_{n+1}} Y_{n+1} + \mathcal{D}_{r_n}^{r_{n+1}} + \lambda_{r_n}^{r_{n+1}} U_{n+1} \\ Y_{n+1} &= \mathcal{F}_{r_n}^{r_{n+1}} Y_n + \mathcal{G}_{r_n}^{r_{n+1}} + \pi_{r_n}^{r_{n+1}} V_{n+1} \end{aligned} \tag{8}$$

where, for $\boldsymbol{r}_1^2 \in \tilde{\Omega}^2$

$$\begin{aligned} \mathcal{C}_{r_1}^{r_2} &= \gamma^2(\boldsymbol{r}_1^2)\left(\gamma^4(\boldsymbol{r}_1^2)\right)^{-1} \\ \mathcal{A}_{r_1}^{r_2} &= a^1(\boldsymbol{r}_1^2) \\ \mathcal{B}_{r_1}^{r_2} &= a^2(\boldsymbol{r}_1^2) - \mathcal{C}_{r_1}^{r_2} a^4(\boldsymbol{r}_1^2), \\ \mathcal{D}_{r_1}^{r_2} &= M^X(r_2) - \mathcal{A}_{r_1}^{r_2} M^X(r_1) - \mathcal{B}_{r_1}^{r_2} M^Y(r_1) - \mathcal{C}_{r_1}^{r_2} M^Y(r_2) \\ \left(\lambda_{r_1}^{r_2}\right)^2 &= \gamma^1(\boldsymbol{r}_1^2) - \mathcal{C}_{r_1}^{r_2} \gamma^3(\boldsymbol{r}_1^2) \\ \mathcal{F}_{r_1}^{r_2} &= a^4(\boldsymbol{r}_1^2) \\ \mathcal{G}_{r_1}^{r_2} &= M^Y(r_2) - a^4(\boldsymbol{r}_1^2) M^Y(r_1) \\ \left(\pi_{r_1}^{r_2}\right)^2 &= \gamma^4(\boldsymbol{r}_1^2) \end{aligned} \tag{9}$$

Let us denote by θ the set of all parameters to be estimated:

$$\theta = \{\mathcal{A}_k^l, \mathcal{B}_k^l, \mathcal{C}_k^l, \mathcal{D}_k^l, \mathcal{F}_k^l, \mathcal{G}_k^l, \pi_k^l, \lambda_k^l, \boldsymbol{\mu}_k, \boldsymbol{\Gamma}_k, p(k,l)\}_{k,l\in\tilde{\Omega}},$$

where $p(k,l)$ is the joint a priori density, while terms $\boldsymbol{\mu}_k$ and $\boldsymbol{\Gamma}_k$ denote the mean vectors and covariance matrices of the Gaussian distributions presented in Section 3.2.

Following the principle of EM, we assume that we have an estimation of the parameters at some iteration q (i.e., θ^q), and we search for the next estimation (i.e., θ^{q+1}), by maximizing an auxiliary function $\mathcal{Q}$. In our case, the auxiliary $\mathcal{Q}$ is defined as

$$\mathcal{Q}(\theta, \theta^q) = \mathbb{E}\left[\ln p_\theta\left(\boldsymbol{x}_1^M, \boldsymbol{y}_1^M, \boldsymbol{r}_1^M\right) | \boldsymbol{x}_1^M, \boldsymbol{y}_1^M, \theta^q\right] \tag{10}$$

This expectation step is followed by a maximization step to find the next estimation of parameters according to $\theta^{q+1} = \text{argmax}_{\theta_q} \mathcal{Q}(\theta, \theta^q)$.

To get into the calculation details, note first that, since $\boldsymbol{t}_1^M = (\boldsymbol{x}_1^M, \boldsymbol{y}_1^M, \boldsymbol{r}_1^M)$ is Markov (see (1)), then

$$p\left(\boldsymbol{t}_1^M\right) = p(\boldsymbol{t}_1) \prod_{n=1}^{M} p(\boldsymbol{t}_{n+1} | \boldsymbol{t}_n)$$

and, from (2) and (5), we can write

$$p(\boldsymbol{t}_{n+1} | \boldsymbol{t}_n) = p(r_{n+1} | r_n)\, p\left(y_{n+1} | y_n, \boldsymbol{r}_n^{n+1}\right) p\left(x_{n+1} | x_n, \boldsymbol{r}_n^{n+1}, \boldsymbol{y}_n^{n+1}\right)$$

Hence, the auxiliary function can be written

$$\begin{aligned} Q(\theta, \theta^q) \quad &= \mathbb{E}[\ln p_\theta(x_1, y_1, r_1) | \boldsymbol{z}_1^M, \theta^q] \\ &\quad + \underbrace{\sum_{n=1}^{M-1} \mathbb{E}[\ln p_\theta(r_{n+1} | r_n) | \boldsymbol{z}_1^M, \theta^q]}_{Q^1(\theta, \theta^q)} \\ &\quad + \underbrace{\sum_{n=1}^{M-1} \mathbb{E}[\ln p_\theta(x_{n+1} | x_n, \boldsymbol{y}_n^{n+1}, \boldsymbol{r}_n^{n+1}) | \boldsymbol{z}_1^M, \theta^q]}_{Q^2(\theta, \theta^q)} \\ &\quad + \underbrace{\sum_{n=1}^{M-1} \mathbb{E}[\ln p_\theta(y_{n+1} | y_n, \boldsymbol{r}_n^{n+1}) | \boldsymbol{z}_1^M, \theta^q]}_{Q^3(\theta, \theta^q)} \end{aligned}$$

The term $Q^1(\theta, \theta^q)$ is used to establish the re-estimation formulas for subset $\tau^1 = \{p(k,l), \boldsymbol{\mu}_k, \boldsymbol{\Gamma}_k\}_{k,l\in\tilde{\Omega}}$. According to (8), the term $Q^2(\theta, \theta^q)$ allows to infer re-estimation formulas for subset $\tau^2 = \{\mathcal{A}_k^l, \mathcal{B}_k^l, \mathcal{C}_k^l, \mathcal{D}_k^l, \lambda_k^l\}_{k,l\in\tilde{\Omega}}$, while the term $Q^3(\theta, \theta^q)$ is used to obtain re-estimation formulas for subset $\tau^3 = \{\mathcal{F}_k^l, \mathcal{G}_k^l, \pi_k^l\}_{k,l\in\tilde{\Omega}}$.

4.1. Re-estimation of the a priori density

Let us denote $\psi_n(k,l) = p(r_n = k, r_{n+1} = l | \boldsymbol{z}_1^M)$, and $\phi_n(k) = p(r_n = k | \boldsymbol{z}_1^M)$, for $k, l \in \tilde{\Omega}$. Using these notations, $Q^1(\theta, \theta^q)$ writes

$$Q^1(\theta, \theta^q) = \sum_{n=1}^{M-1} \sum_{k,l\in\tilde{\Omega}} \ln\left(p^q(l|k)\right) \psi_n(k,l) \tag{11}$$

Classically, we get

$$p^{q+1}(k,l) = \sum_{n=1}^{M-1} \psi_n(k,l)$$

and

$$p^{q+1}(l|k) = \frac{\sum_{n=1}^{M-1} \psi_n(k,l)}{\sum_{n=1}^{M-1} \phi_n(k)}$$

where

$$\phi_n(k) = p\left(r_n = k | \boldsymbol{z}_1^M\right) = \alpha_n(k)\, \beta_n(k) \tag{12}$$

and $\psi_n(k,l)$ is given by Eq. (13).

$$\psi_n(k,l) = \frac{\alpha_n(k)\, p(\boldsymbol{z}_{n+1}, r_{n+1} = l | \boldsymbol{z}_n, r_n = k)\, \beta_{n+1}(l)}{\sum_{k^\star, l^\star\in\tilde{\Omega}} \alpha_n(k^\star)\, p(\boldsymbol{z}_{n+1}, r_{n+1} = l^\star | \boldsymbol{z}_n, r_n = k^\star)\, \beta_{n+1}(l^\star)} \tag{13}$$

Hence, the joint and marginal a posteriori probabilities can be calculated using the so-called forward and backward probabilities (Rabiner, 1989) $\alpha_n(k) = p(r_n = k, \boldsymbol{z}_1^n)$ and $\beta_n(r_n = k) = p(\boldsymbol{z}_{n+1}^M | r_n = , \boldsymbol{z}_n)$, whose discrete recursive calculations are provided as

$$\begin{aligned} \alpha_1(l) &= p(r_1 = l, \boldsymbol{z}_1) = p(r_1 = l)\, p(\boldsymbol{z}_1 | r_1 = l) \\ \alpha_{n+1}(l) &= \sum_{k \in \tilde{\Omega}} \alpha_n(k)\, p(\boldsymbol{z}_{n+1}, r_{n+1} = l | \boldsymbol{z}_n, r_n = k) \end{aligned} \tag{14}$$

and

$$\begin{aligned} \beta_M(k) &= 1, \\ \beta_n(k) &= \sum_{l \in \tilde{\Omega}} \beta_{n+1}(l)\, p(\boldsymbol{z}_{n+1}, r_{n+1} = l | \boldsymbol{z}_n, r_n = k) \end{aligned} \tag{15}$$

for $n = 1, \ldots, M-1$.

4.2. Re-estimation of the Gaussian densities

According to the procedure described in Section 3.2, we first compute the mean vectors and covariances matrices for the two hard switches 0 and 1, using classical re-estimation formulas

$$\boldsymbol{\mu}_0 = \frac{\sum_{n=1}^M \phi_n(0)\, \boldsymbol{z}_n}{\sum_{n=1}^M \phi_n(0)} \quad \boldsymbol{\mu}_1 = \frac{\sum_{n=1}^M \phi_n(1)\, \boldsymbol{z}_n}{\sum_{n=1}^M \phi_n(1)}$$

$$\boldsymbol{\Gamma}_0 = \frac{\sum_{n=1}^M \phi_n(0)\, (\boldsymbol{z}_n - \boldsymbol{\mu}_0)(\boldsymbol{z}_n - \boldsymbol{\mu}_0)^\top}{\sum_{n=1}^M \phi_n(0)}$$

$$\boldsymbol{\Gamma}_1 = \frac{\sum_{n=1}^M \phi_n(1)\, (\boldsymbol{z}_n - \boldsymbol{\mu}_1)(\boldsymbol{z}_n - \boldsymbol{\mu}_1)^\top}{\sum_{n=1}^M \phi_n(1)}$$

and then deduce the mean vectors and covariances matrices for the fuzzy switches $\frac{1}{F+1}, \ldots, \frac{F}{F+1}$, by linear interpolation using Eq. (6).

4.3. Re-estimation of CGOMFSM coefficients

The method to estimate parameters in τ^2 and τ^3 are based on the same principle, whose calculations are reported in Appendix A.

4.3.1. Parameters in τ^2

Since the law of X_{n+1} conditionally to $X_n, \boldsymbol{Y}_n^{n+1}, \boldsymbol{R}_n^{n+1}$ is Gaussian, with mean $\mathcal{A}_{r_n}^{r_{n+1}} x_n + \mathcal{B}_{r_n}^{r_{n+1}} y_n + \mathcal{C}_{r_n}^{r_{n+1}} y_{n+1} + \mathcal{D}_{r_n}^{r_{n+1}}$ and covariance $(\lambda_{r_1}^{r_2})^2$, we can write

$$Q^2(\theta, \theta^q) = \sum_{k,l \in \tilde{\Omega}} Q_{k,l}^2(\theta, \theta^q),$$

with, for all $k, l \in \tilde{\Omega}$, and $\mathcal{M}_k^l = \left[\mathcal{A}_k^l, \mathcal{B}_k^l, \mathcal{C}_k^l, \mathcal{D}_k^l\right]$

$$Q_{k,l}^2(\theta, \theta^q) = -\frac{1}{2} \sum_{n=1}^{M-1} \psi_n(k,l) \times \left[\ln\left(2\pi(\lambda_k^l)^2\right) + \left(x_{n+1} - \mathcal{M}_k^l \begin{bmatrix} x_n \\ y_n \\ y_{n+1} \\ 1 \end{bmatrix} \right)^2 (\lambda_k^l)^{-2} \right]$$

We look for an estimation $\mathcal{M}_{\cdot k}^{\,l}$ of $\mathcal{M}_k^l$ which maximizes $Q^2(\theta, \theta^q)$. Since all the parameters to be estimated depend on k and l, the parameters that maximize $Q_{k,l}^2(\theta, \theta^q)$ will also maximize $Q^2(\theta, \theta^q)$, making the problem easier to solve.

To get $\mathcal{M}_{\cdot r_1}^{\,r_2}$, we make use of the result presented in Appendix A. Setting $\omega_n = \psi_n(k,l)$, $\nu_{n+1} = x_{n+1}$, $\boldsymbol{\xi}^\top = \mathcal{M}_k^l$, $\sigma = \lambda_k^l$, $\boldsymbol{u}_{n+1} = [x_n, y_n, y_{n+1}, 1]^\top$ and $d = 4$, we get from (1) the coordinates of the maximum $\mathcal{M}_{\cdot k}^{\,l}$ as

$$\begin{aligned} \mathcal{M}_{\cdot k}^{\,l} &= \left[\mathcal{A}_{\cdot k}^{\,l}, \mathcal{B}_{\cdot k}^{\,l}, \mathcal{C}_{\cdot k}^{\,l}, \mathcal{D}_{\cdot k}^{\,l}\right] \\ &= \frac{\sum_{n=1}^{M-1} \psi_n(k,l)\, x_{n+1} [x_n, y_n, y_{n+1}, 1]}{\sum_{n=1}^{M-1} \psi_n(k,l) \begin{bmatrix} \boldsymbol{z}_n \boldsymbol{z}_n^\top & y_{n+1} \boldsymbol{z}_n & \boldsymbol{z}_n \\ y_{n+1} \boldsymbol{z}_n^\top & y_{n+1}^2 & y_{n+1} \\ \boldsymbol{z}_n^\top & y_{n+1} & 1 \end{bmatrix}} \end{aligned} \tag{16}$$

The re-estimation formula of covariances $(\lambda_k^l)^2$ is given by Eq. (17), by applying Eq. (2) in Appendix A.

$$\left(\dot{\lambda}_k^l\right)^2 = \frac{\sum_{n=1}^{M-1} \psi_n(k,l) \left(x_{n+1} - \left(\mathcal{A}_{\cdot k}^{\,l} x_n + \mathcal{B}_{\cdot k}^{\,l} y_n + \mathcal{C}_{\cdot k}^{\,l} y_{n+1} + \mathcal{D}_{\cdot k}^{\,l} \right) \right)^2}{\sum_{n=1}^{M-1} \psi_n(k,l)} \tag{17}$$

4.3.2. Parameters in τ^3

The approach is essentially the same as that taken previously. Since the law of Y_{n+1} conditionally to $Y_n, \boldsymbol{R}_n^{n+1}$ is Gaussian, with mean $\mathcal{F}_{r_1}^{r_2} y_n + \mathcal{G}_{r_1}^{r_2}$ and covariance $(\pi_{r_1}^{r_2})^2$, we can write

$$Q^3(\theta, \theta^q) = \sum_{k,l \in \tilde{\Omega}} Q_{k,l}^3(\theta, \theta^q)$$

with, for all $k, l \in \Omega$, and $\mathcal{P}_k^l = [\mathcal{F}_k^l, \mathcal{G}_k^l]$

$$Q_{k,l}^3(\theta, \theta^q) = -\frac{1}{2} \sum_{n=1}^{M-1} \psi_n(k,l) \left[\ln\left(2\pi(\pi_k^l)^2\right) + \left(y_{n+1} - \mathcal{P}_k^l \begin{bmatrix} y_n \\ 1 \end{bmatrix} \right)^2 (\pi_k^l)^{-2} \right]$$

Setting $\omega_n = \psi_n(k,l)$, $\nu_{n+1} = y_{n+1}$, $\boldsymbol{\xi}^\top = \mathcal{P}_k^l$, $\sigma = \pi_k^l$, $\boldsymbol{u}_{n+1} = (y_n, 1)^\top$ and $d = 2$ in Eq. (1), we get the maximum $\mathcal{P}_{\cdot k}^{\,l}$ as

Table 1. Selected measurement sites from England highways data sources (MIDAS and TMU stations deployed over freeways A1, A2, A3).

Site name	GPS coordinates	Date & time measurements
A1: MIDAS site 714	Long: −1.56427 Lat: 54.90506	1/1/19 0:14- 1/31/19 23:59
A1: MIDAS site 4940	Long: −1.57134 Lat: 54.90748	1/1/19 0:14 – 1/31/19 23:59
A1: MIDAS site 3636	Long: −1.57905 Lat: 54.90999	1/1/19 0:14 – 1/31/19 23:59
A1: TMU site 6503/2	Long: −0.41472 Lat: 52.58595	1/1/19 0:14 – 1/31/19 23:59
A1: TMU site 6504/1	Long: −0.42681 Lat: 52.60344	1/1/19 0:14 – 1/31/19 23:59
A2: MIDAS site 2356	Long: 0.308704 Lat: 51.43043	1/1/19 0:14 – 1/31/19 23:59
A2: MIDAS site 3261	Long: 0.359875 Lat: 51.41557	2/1/19 0:14 – 2/28/19 23:59
A2: TMU site 5848/2	Long: 0.442598 Lat: 51.39957	3/1/19 0:14 – 3/31/19 23:59
A2: TMU site 6048/2	Long: 0.175526 Lat: 51.43775	4/1/19 0:14 – 4/30/19 23:59
A3: TMU site 5509/2	Long: −0.96399 Lat: 50.98383	1/1/18 0:14 – 1/31/18 23:59
A3: MIDAS site 14052	Long: −0.76243 Lat: 51.09454	1/1/18 0:14 – 1/31/18 23:59

$$\mathcal{P}_{\cdot k}^{\ l} = \left[\mathcal{F}_{\cdot k}^{\ l}, \mathcal{G}_{\cdot k}^{\ l}\right] = \frac{\sum_{n=1}^{M-1} \psi_n(k,l)\, y_{n+1}[y_n, 1]}{\sum_{n=1}^{M-1} \psi_n(k,l) \begin{bmatrix} {y_n}^2 & y_n \\ y_n & 1 \end{bmatrix}} \quad (18)$$

Still using Eq. (2), the re-estimation equation for the variances $(\pi_k^l)^2$ is given by

$$\left(\dot{\pi}_k^l\right)^2 = \frac{\sum_{n=1}^{M-1} \psi_n(k,l) \left(y_{n+1} - \left(y_n \mathcal{F}_{\cdot r_n}^{\ r_{n+1}} + \mathcal{G}_{\cdot r_n}^{\ r_{n+1}}\right)\right)^2}{\sum_{n=1}^{M-1} \psi_n(k,l)} \quad (19)$$

Let us conclude this section with some remarks about the implementation of the semi-supervised parameter estimation algorithm that we proposed:

1. As for τ^1, the parameters in τ^2 and τ^3 are recomputed at each iteration, iteration index $q+1$ has been omitted for keeping the writing as simple as possible.
2. As already mentioned, the parameters in τ^2 and τ^3 can not be calculated for $r_n, r_{n+1} \in [0,1]$. They are instead computed for a number $F+2$ of discrete fuzzy levels. For example, $\mathcal{A}_{\cdot r_n}^{\ r_{n+1}}$ is implemented as an array of size $(F+2) \times (F+2)$ corresponding to $\mathcal{A}_{\cdot k}^{\ l}$. The higher is F, the more precision we get for the fuzziness, but at the expense of more computing time. The impact of the choice of F for the concerned application is studied in the following section.
3. The principle of EM is based on the iterative re-estimation of parameters until we observe a convergence (or, practically, until a fixed number of iterations has been reached). The algorithm requires the parameters to be initialized, preferably not too far from the real parameters, to avoid trapping in local maxima. For initialization, we used a simple K-means algorithm with $F+2$ classes to obtain a realization of the jumps, and developed empirical estimators for all the parameters in θ, on the basis of the so-generated complete data.

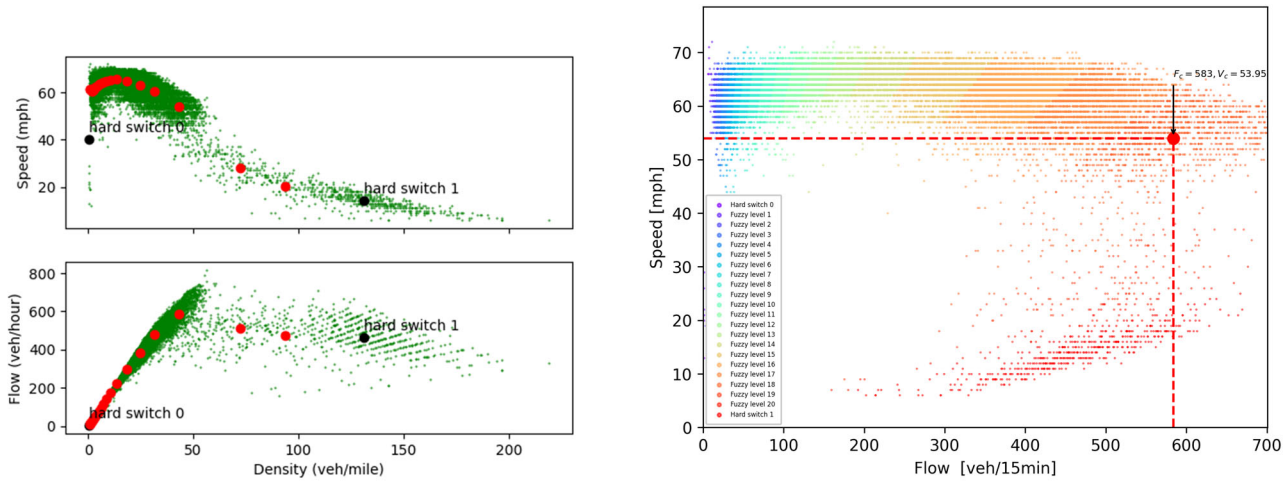
5. Experimental study

In this section, we present the results of our experimental study on the prediction of traffic data. The algorithms were implemented using C programming language on a machine with 2.2 Ghz Intel core i7 processor and 16 GB memory.

The data used in this study have been extracted from the Highways England's Motorway Incident Detection and Automatic Signaling (MIDAS) system and Traffic Monitoring Units (TMU) provided by the Network Information Services (NIS). Both systems are deployed across the United Kingdom's highway network relying mainly on inductive loops to collect traffic data approximately each 500 meters. The rationale behind choosing these datasets is that the measuring systems cover a variety of disparate geographical areas with a fairly low ratio of incomplete data. We selected traffic flow and speed data sources from the measurement sites detailed in Table 1. These sites have been selected in a way to consider datasets from both MIDAS and TMU that correspond to geographically dispersed highways with different patterns of traffic volume and speed limits. Each dataset contains one-month data collected at regular 15-minute intervals.

5.1. Experimental setup

Each data source is split into two sets: (i) a training set corresponding to the measurements of the first three weeks z_1^M and (ii) a test set of the remaining data points z_{M+1}^{M+T}. The CGOMFSM predictor's parameters are estimated using z_1^M. Afterwards, the prediction is iteratively carried out over the test set z_{M+1}^{M+T}. The training set corresponds to the measures acquired during three weeks which corresponds to roughly 2000 data points. We vary the prediction horizon denoted by $\mathcal{W}$ between 1 and 4, which corresponds



(a) Density-flow (down) and density-speed (up) diagrams

(b) Flow-speed diagram, The red dot corresponds to the capacity flow and capacity speed

Figure 2. Density-flow, density-speed and flow-speed diagrams corresponding to the monitoring site MIDAS 14052. (a) Density-flow (down) and density-speed (up) diagrams and (b) Flow-speed diagram, the red dot corresponds to the capacity flow and capacity speed.

respectively to one-step ahead (i.e., 15 minutes) and four-steps ahead (i.e., 60 minutes). In the considered data sources, the records corresponding to insignificant traffic were disregarded in the error computation as they might falsely impact the accuracy of the prediction scheme. Such records occur mainly overnight when the number of detected vehicles is significantly low. Typically, we consider data entries that correspond to a traffic flow of at least 50 vehicles. To illustrate the relevance of choosing a regime switching model, we depict in Figure 2 the traffic dynamics of the site MIDAS 14052 spanning over a period of one year. Speed-flow, speed-density and density-flow diagrams show two hard switches (plotted in black in Figure 2(a)) corresponding to free flow and congested traffic state respectively as well as the different fuzzy levels (plotted in red). Figure 2(b) also demonstrates the gradual transition between the different traffic states ranging from free-flowing traffic (corresponding to the level 0) to severely congested traffic (corresponding the level 1).

We conducted two series of experiments. In the first series we study the impact of the number of fuzzy levels F on the accuracy of the prediction. In the second series, we attempt to assess the impact of $\mathcal{W}$ on the forecasting precision.

The proposed prediction algorithm has been compared to three reference ones: (i) random walk (RW), a naive algorithm (ii) seasonal ARIMA, a parametric scheme widely used in time series forecasting, and (iii) LSTM, a non-parametric scheme based on long short-term memory. Given that traffic flow typically follows a cyclical pattern over the days of the week, we calibrated the RW and ARIMA algorithms to take into account these cyclical patterns. For the RW algorithm, we used the average flow of 3 weeks to predict the flow rate. We selected the seasonal ARIMA with order (1, 0, 1) and seasonal order (1, 1, 0, 672) (Guo et al., 2014) to consider the cyclical pattern of one-week flow rate instead of one-day flow rate. We considered each dataset independently from the others for training the ARIMA model. The error metrics used for comparison are the mean absolute percentage error (MAPE) and the root mean square error (RMSE) defined as follows:

$$\mathrm{MAPE} = \frac{1}{T}\sum_{i=M+1}^{M+T} \left|\frac{\hat{x}_i - x_i}{x_i}\right|$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{T}\sum_{i=M+1}^{M+T} (\hat{x}_i - x_i)^2}$$

Where x denote the observed traffic flow values and $\hat{x}$ denote the estimated ones.

5.2. Results

The performance results of the proposed CGOMFSM^F predictor for $F=0$ (crisp model, i.e., CGOMSM) and $F=5$, 10, 20 (fuzzy model, i.e., CGOFMSM) compared to LSTM, ARIMA and RW are detailed in Figure 3 (showing the MAPE results) and Figure 4 (showing the RMSE results). We notice that CGOMFSM^{20} provides the least error in terms of MAPE and RMSE and outperforms its LSTM

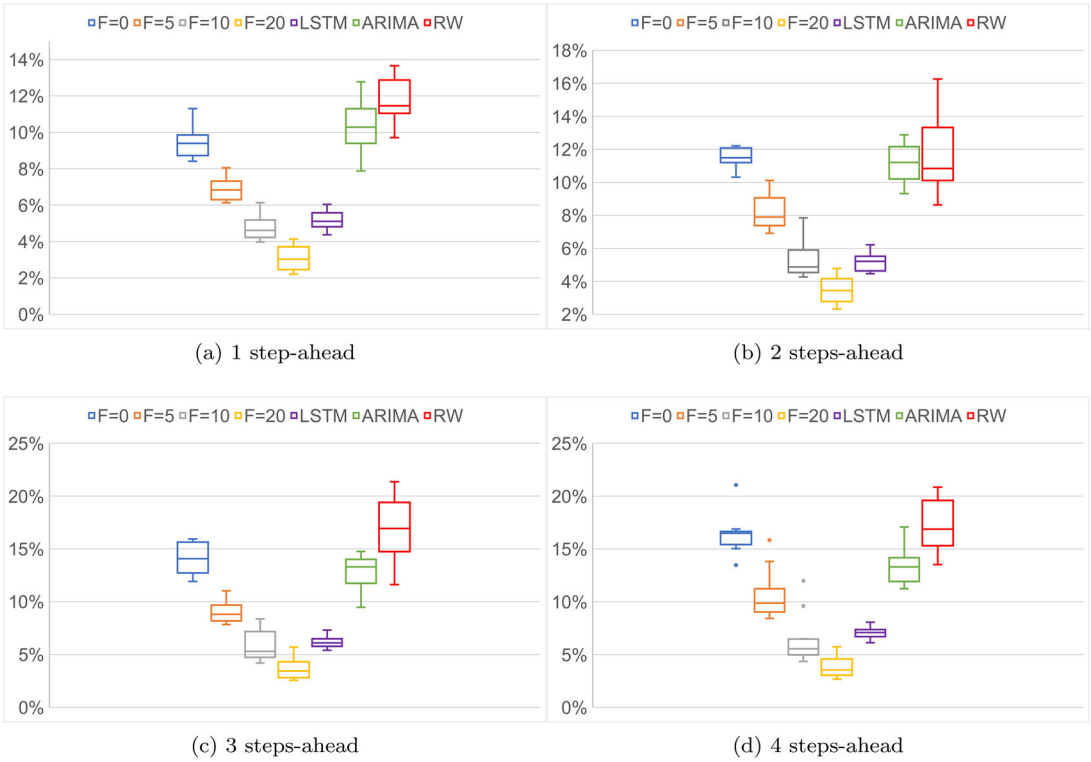


Figure 3. One step (a), two steps (b), three steps (c), and four steps (d) ahead MAPE prediction results for CGOMFSM with 0, 5, 10 and 20 fuzzy levels vs. baseline models. (a) 1 step-ahead, (b) 2 steps-ahead, (c) 3 steps-ahead, and (d) 4 steps-ahead.

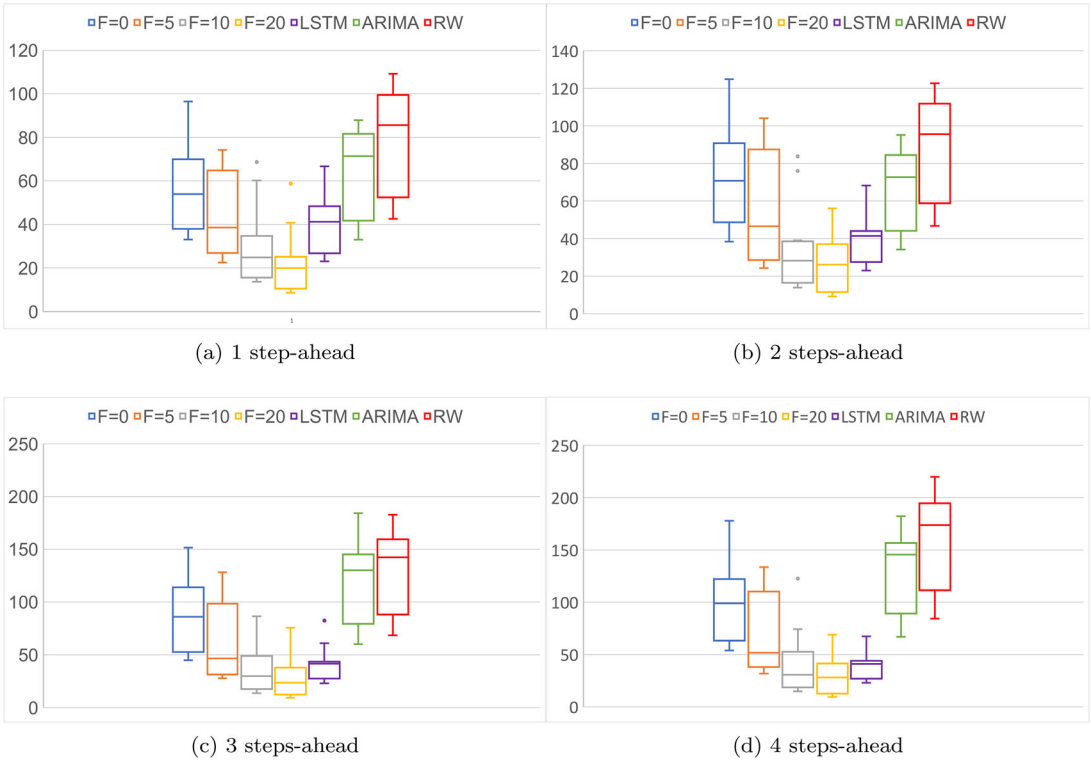
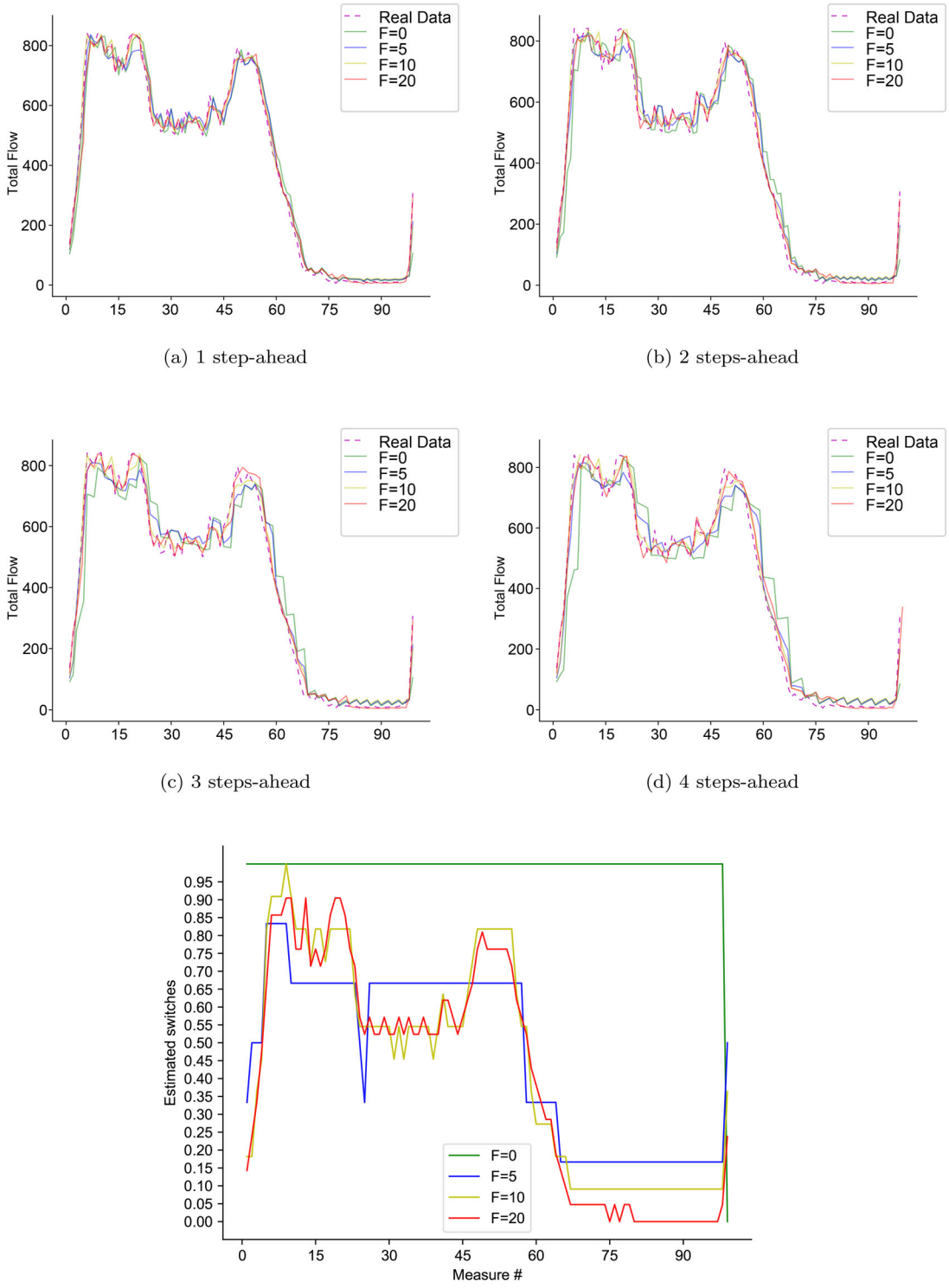


Figure 4. One step (a), two steps (b), three steps (c), and four steps (d) ahead RMSE prediction results for CGOMFSM with 0, 5, 10 and 20 fuzzy levels vs. baseline models. (a) 1 step-ahead, (b) 2 steps-ahead, (c) 3 steps-ahead, and (d) 4 steps-ahead.

competitor. However, when the number of fuzzy levels F is below 5, the LSTM predictor yields error rates better than those provided by our algorithm. The crisp model ($F = 0$) fails to match up to LSTM and



(a) 1 step-ahead

(b) 2 steps-ahead

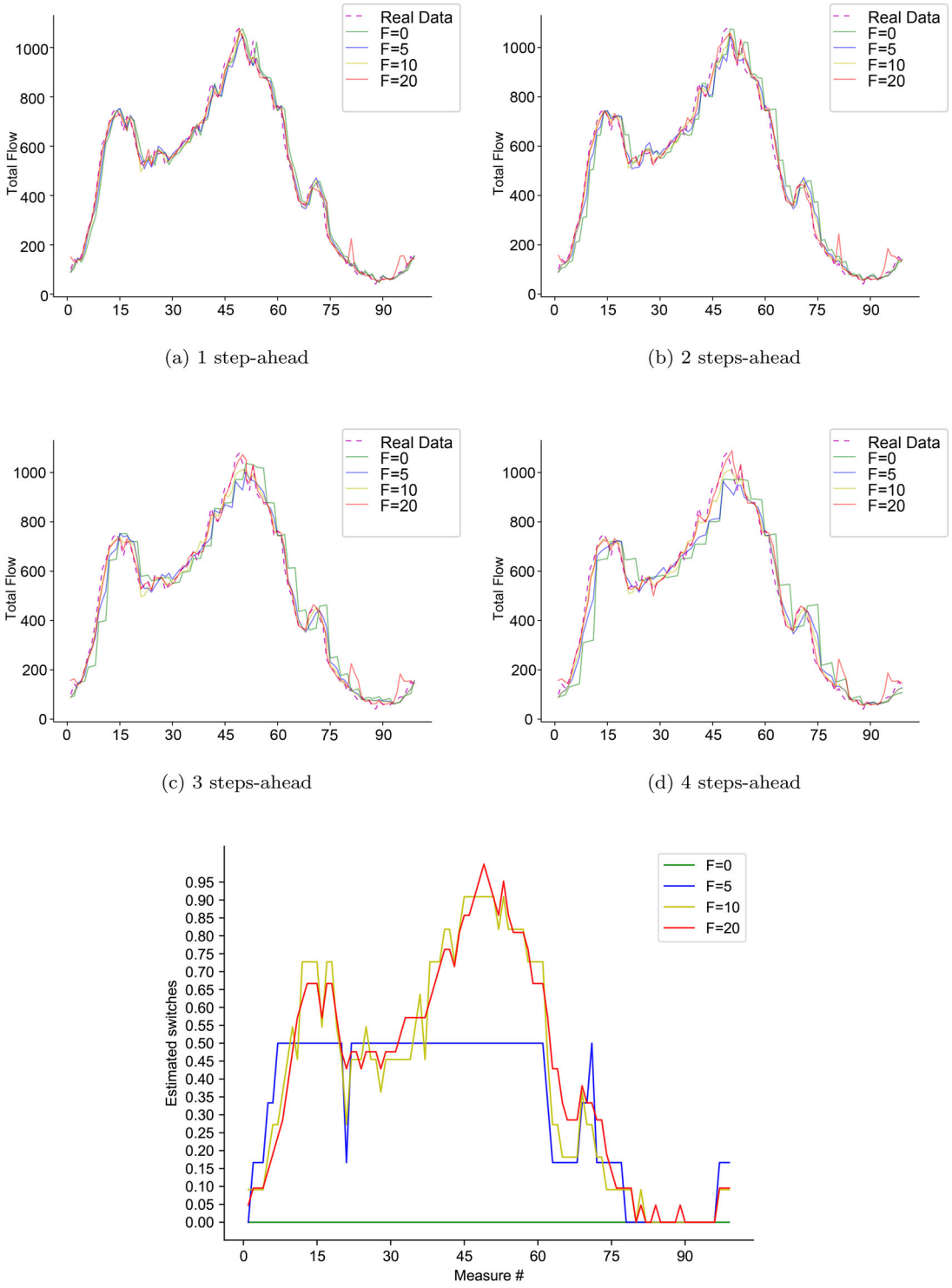
(c) 3 steps-ahead

(d) 4 steps-ahead

(e) Estimated switches (0 corresponds to free flow and 1 to traffic congestion)

Figure 5. Total carriageway flow (vehicles/15 min) prediction results for site TMU5848 and for (a) 15 minutes, (b) 30 minutes, (c) 45 minutes and (d) 60 minutes horizon. The x-axis corresponds to time steps. (a) 1 step-ahead, (b) 2 steps-ahead, (c) 3 steps-ahead, (d) 4 steps-ahead, and (e) Estimated switches (0 corresponds to free flow and 1 to traffic congestion).

remains within the error range of ARIMA specially for prediction horizons greater than 30 minutes. Figures 5–8 illustrate four examples of the prediction results obtained in different MIDAS and TMU sites. Overall, these results show that increasing the number of fuzzy levels improves the accuracy of CGOMFSM. This finding is corroborated by the figures representing the estimated switches. The plots showing the



(a) 1 step-ahead

(b) 2 steps-ahead

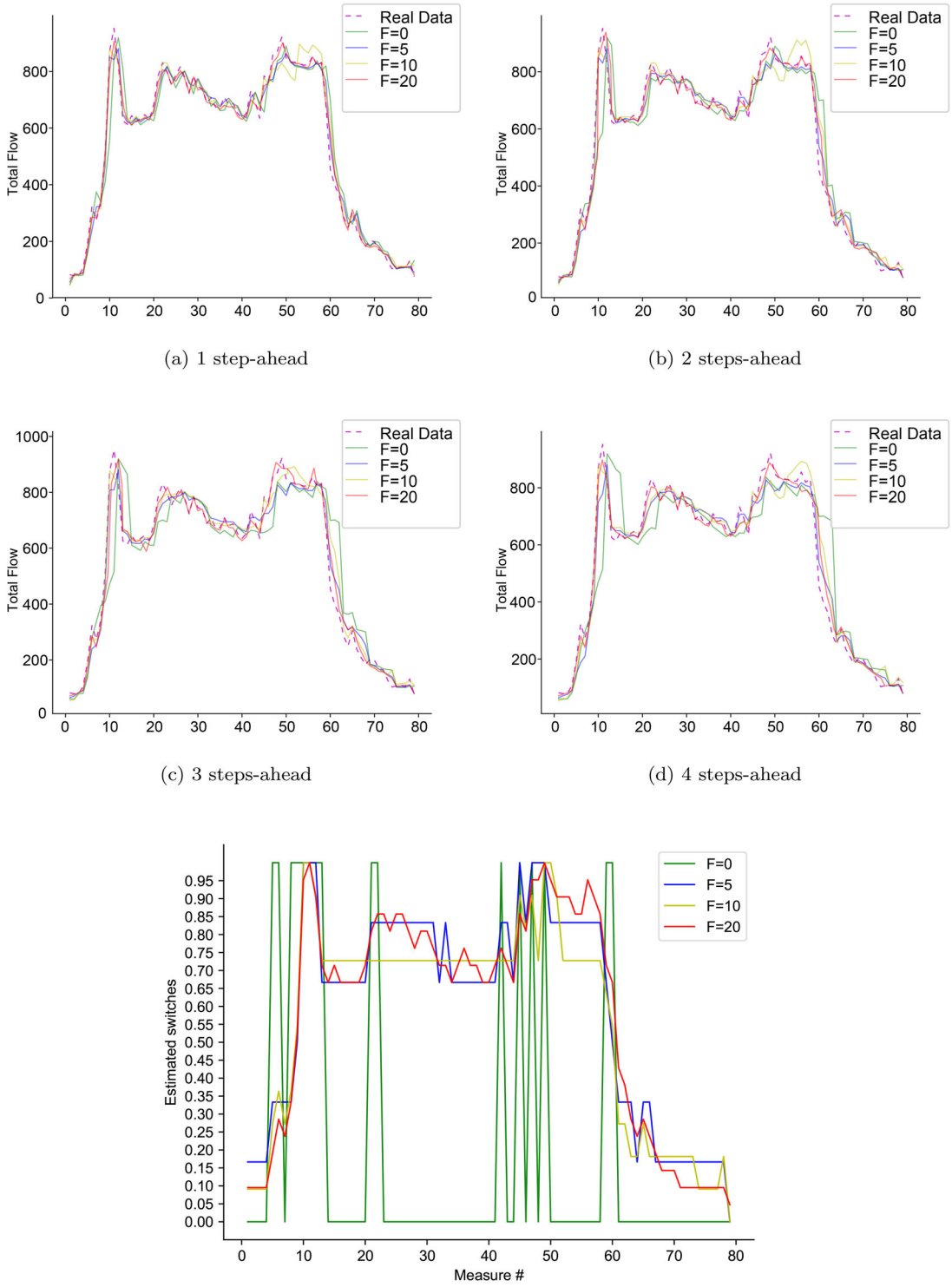
(c) 3 steps-ahead

(d) 4 steps-ahead

(e) Estimated switches (0 corresponds to free flow and 1 to traffic congestion)

Figure 6. Total carriageway flow prediction results for site TMU6048 and for (a) 15 minutes, (b) 30 minutes, (c) 45 minutes and (d) 60 minutes horizon. The x-axis corresponds to time steps. (a) 1 step-ahead, (b) 2 steps-ahead, (c) 3 steps-ahead, (d) 4 steps-ahead, and (e) Estimated switches (0 corresponds to free flow and 1 to traffic congestion).

switches process represents how the traffic condition evolves over time and exhibits the traffic state. From the plots of predicted data, we notice that the predicted flow diverges from the observed one specially when the traffic transits from one switch to another. For example, in Figure 5(e), the crisp model fails to predict the traffic flow compared to the CGOMFSM[20] in the interval 50-65. In the estimated switches plot,



(a) 1 step-ahead

(b) 2 steps-ahead

(c) 3 steps-ahead

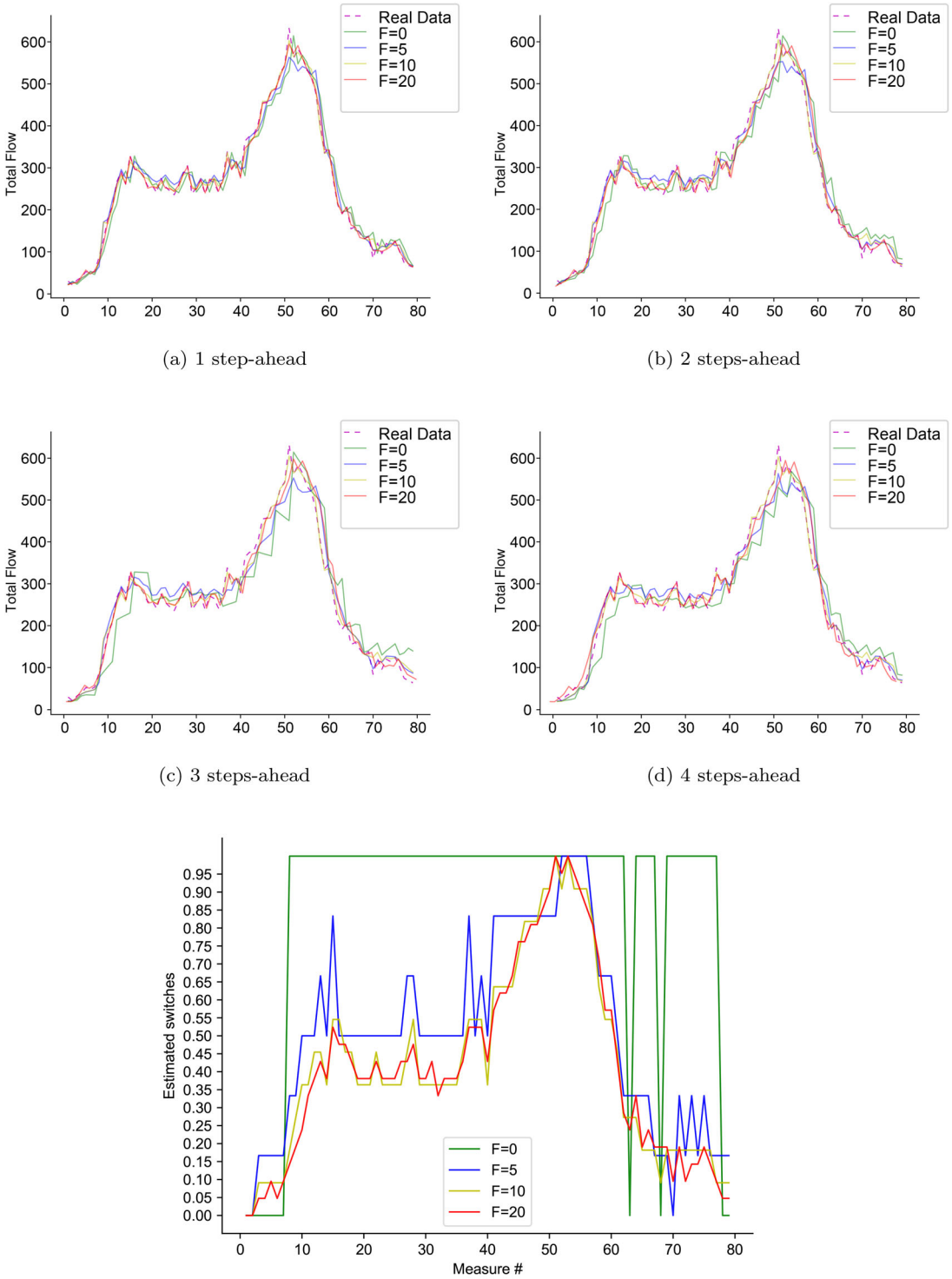
(d) 4 steps-ahead

(e) Estimated switches (0 corresponds to free flow and 1 to traffic congestion)

Figure 7. Total carriageway flow (vehicles/15 min) prediction results for site MIDAS3636 and for (a) 15 minutes, (b) 30 minutes, (c) 45 minutes and (d) 60 minutes horizon. The x-axis corresponds to time steps. (a) 1 step-ahead, (b) 2 steps-ahead, (c) 3 steps-ahead, (d) 4 steps-ahead, and (e) Estimated switches (0 corresponds to free flow and 1 to traffic congestion).

we notice that the slope of the plot is more transient with $F = 20$ compared to the other models where the transition is rather salient and therefore limits the performance of the predictor.

To better assess the performance of CGOMFSM for traffic flow prediction, we evaluated the percentage of missed and exact predictions. Missed predictions correspond to errors above 10% while exact ones are



(a) 1 step-ahead

(b) 2 steps-ahead

(c) 3 steps-ahead

(d) 4 steps-ahead

(e) Estimated switches (0 corresponds to free flow and 1 to traffic congestion)

Figure 8. Total carriageway flow prediction results for site MIDAS14052 and for (a) 15 minutes, (b) 30 minutes, (c) 45 minutes and (d) 60 minutes horizon. The x-axis corresponds to time steps. (a) 1 step-ahead, (b) 2 steps-ahead, (c) 3 steps-ahead, (d) 4 steps-ahead, and (e) Estimated switches (0 corresponds to free flow and 1 to traffic congestion).

with forecast error less than 1%. Figure 9 reports the percentage of missed and exact predictions. The results suggest that the higher the number of fuzzy switches the more accurate the precision of the CGOMFSM predictor is. These findings are consistent with the MAPE and RMSE results. It should be noted that for a short prediction horizon (typically 15-30 minutes), 10 fuzzy levels appear to be sufficient.

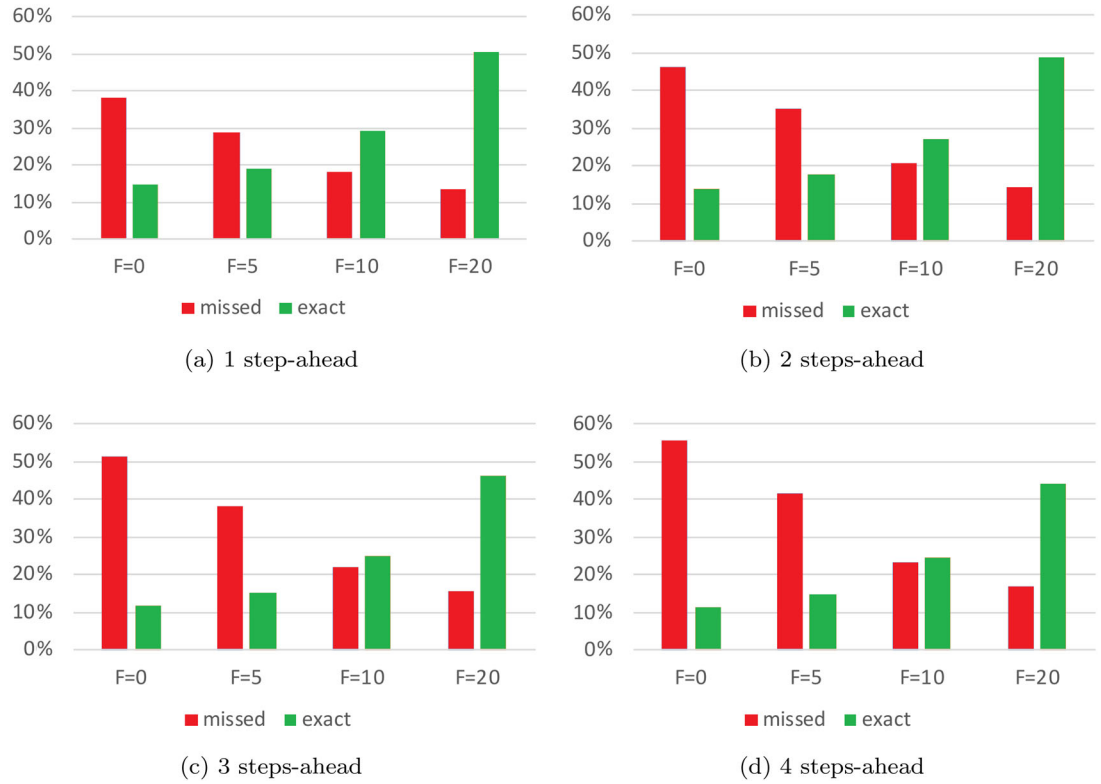


Figure 9. Missed and exact prediction percentages using CGOMFSM with $F = 0, 5, 10, 20$ fuzzy levels. Missed predictions correspond to errors above 10% while exact ones are with forecast error less than 1%. (a) 1 step-ahead, (b) 2 steps-ahead, (c) 3 steps-ahead, and (d) 4 steps-ahead.

Table 2. Running time as function of the number of fuzzy levels.

	$F=0$	$F=5$	$F=10$	$F=15$	$F=20$
Estimation	5 s	8 s	198 s	371 s	2552s
Prediction	0.105 s	0.190 s	0.213 s	0.303 s	1.401 s

The reported values are the averages of experiments conducted on the 12 sites. The average size of the training set is 2750 data points.

When the prediction horizon is relatively high (45 – 60 minutes) the CGOMFSM predictor with 20 fuzzy levels provides significantly more accurate results. However, the accuracy of the model with 20 fuzzy levels is achieved to the expense of computing time. Indeed, as reported in Table 2, the running time for the CGOMFSM parameter estimation increases from 5 seconds for the crisp model to 42 minutes when the number of fuzzy levels is as high as 20. It is worth mentioning that the average training time was 7 seconds for ARIMA and 13 seconds for LSTM. Regardless of the number of fuzzy levels, the required computation time for prediction remains below 1500 ms, which is compatible with the application context and allows for the development of real-world traffic prediction systems.

5.3. From traffic data to traffic information

Another paramount feature of the proposed prediction algorithm is that it allows to infer a user-friendly representation of traffic data. From a practical perspective, it is more appropriate to provide end users of intelligent transportation systems with an explicit representation of traffic state than a raw carriageway flow data or average speed data. The explicit introduction of switches as processes in the CGOMFSM helps achieving this goal. Figure 10 depicts excerpts over 20 hours of estimated traffic states using 0, 5, 10 and 20 fuzzy levels. We observe a more transient transition between traffic states (switches) for higher numbers of fuzzy levels. Unsurprisingly, the crisp model fails to detect distinct traffic states and assimilates the aggregated data to a single state (Figure 10(b) and (d)). Compared to the crisp model where only two states are predicted - free or congested-, we believe that distinguishing different transient traffic states in-between could be of interest to many road users who need to plan their trips in advance, and for whom the distinction between "moderately congested" and "severely congested" may make a big difference.

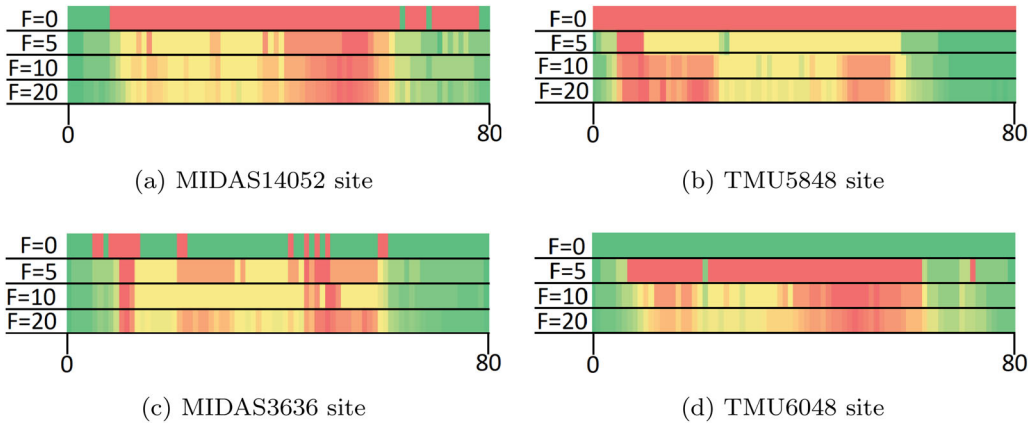


Figure 10. Estimated switches. Red color corresponds to switch 1 (congested traffic), whereas green color to switch 0 (free flow), yellow, orange and light green colors correspond to intermediary traffic states. (a) MIDAS14052 site, (b) TMU5848 site, (c) MIDAS3636 site, and (d) TMU6048 site.

6. Conclusion and future work

In this paper we proposed a novel short-term traffic state prediction algorithm based on a recent fuzzy variant (Bouyahia et al., 2020) of the conditionally Gaussian observed Markov switching models introduced in Abbassi et al. (2015). The main motivation behind the use of CGOMFSM is that it allows for a more realistic representation of the transitions between traffic switches, which allows to significantly reduce lags. CGOMFSM exploits the inherent correlation between traffic flow and average speed to infer a representational parametrization of the predictor. The idea behind this model is to extend the two-state binary representation into a more progressive representation, allowing to take into account an infinity of intermediate jumps, interpolated from the crisp model boundaries. In practice the number of intermediate states (called "fuzzy levels") is discrete but remains possibly very large, allowing a fine approximation close to a continuous transition, albeit at the expense of an increasing computation time. In this context, we developed an EM-based parameter estimation method for the model, to fit the parameters to the observed data for optimal prediction.

We showed through an experimental study that CGOMFSM yields promising prediction results when the number of fuzzy levels is high enough. Unlike the CGOMFSM proposed in Bouyahia et al. (2021), the model presented in this paper yields encouraging results for prediction horizons larger than 15 minutes. The prediction accuracy has been demonstrated for a time horizon up to 60 minutes and the experimental findings are very promising. The major contribution of CGOMFSM in the prediction of traffic states is its ability to transform discrete discontinuities into a continuous one allowing for a transient and smooth transition between the system regimes regardless of the transition lag. The "optimal" value of fuzzy levels has been inferred by an empirical study, yet we plan to further investigate a fair tradeoff between computation time and prediction accuracy. Since CGOMFSM supports multivariate prediction, one interesting direction for future research should address the inclusion of additional traffic data such as travel times in the prediction process, as it may help adjusting the model parametrization for better performances.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work is partially supported by The Research Council (TRC) of Oman, Grant BFP/RGP/ICT/18/119.

ORCID

Zied Bouyahia http://orcid.org/0000-0002-1103-2169
Hedi Haddad http://orcid.org/0000-0002-2070-4801
Stéphane Derrode http://orcid.org/0000-0002-2865-2057
Wojciech Pieczynski http://orcid.org/0000-0002-1371-2627

References

Abbassi, N., Benboudjema, D., Derrode, S., & Pieczynski, W. (2015). Optimal filter approximations in conditionally Gaussian pairwise Markov switching models. *IEEE Transactions on Automatic Control*, *60*(4), 1104–1109. https://doi.org/10.1109/TAC.2014.2340591

Agudelo-España, D., Gomez-Gonzalez, S., Bauer, S., Schölkopf, B., & Peters, J. (2019). Bayesian online

detection and prediction of change points. *arXiv preprint arXiv:1902.04524*.

Ahmed, M. S., & Cook, A. R. (1979). Analysis of freeway traffic time-series data by using box-jenkins techniques. *Transportation Research Record*, *722*, 1–9.

Aminikhanghahi, S., & Cook, D. (2017). A survey of methods for time series change point detection. *Knowledge and Information Systems*, *51*(2), 339–367.

Antoniou, C., Koutsopoulos, H., & Yannis, G. (2013). Dynamic data-driven local traffic state estimation and prediction. *Transportation Research Part C: Emerging Technologies*, *34*, 89–107. https://doi.org/10.1016/j.trc.2013.05.012

Bergmeir, C., Triguero, I., Molina, D., Aznarte, J., & Benitez, J. (2012). Time series modeling and forecasting using memetic algorithms for regime-switching models. *IEEE Transactions on Neural Networks and Learning Systems*, *23*(11), 1841–1847. https://doi.org/10.1109/TNNLS.2012.2216898

Bouyahia, Z., Derrode, S., & Pieczynski, W. (2020). Filtering in Gaussian linear systems with fuzzy switches. *IEEE Transactions on Fuzzy Systems*, *28*(8), 1760–1770. https://doi.org/10.1109/TFUZZ.2019.2921944

Bouyahia, Z., Haddad, H., Derrode, S., & Pieczynski, W. (2021). Toward a cost-effective motorway traffic state estimation from sparse speed and gps data. *IEEE Access*, *9*, 44631–44646. https://doi.org/10.1109/ACCESS.2021.3066422

Cabrieto, J., Adolf, J., Tuerlinckx, F., Kuppens, P., & Ceulemans, E. (2018). Detecting long-lived autodependency changes in a multivariate system via change point detection and regime switching models. *Scientific Reports*, *8*(1), 15637. https://doi.org/10.1038/s41598-018-33819-8

Cetin, M., & Comert, G. (2006). Short-term traffic flow prediction with regime switching models. *Transportation Research Record*, *1965*(1), 23–31. https://doi.org/10.1177/0361198106196500103

Chiappa, S. (2014). Explicit-duration Markov switching models. *Foundations and Trends® in Machine Learning*, *7*(6), 803–886. https://doi.org/10.1561/2200000054

Comert, G., & Bezuglov, A. (2013). An online change-point-based model for traffic parameter prediction. *IEEE Transactions on Intelligent Transportation Systems*, *14*(3), 1360–1369. https://doi.org/10.1109/TITS.2013.2260540

Doucet, A., & Andrieu, C. (2001). Iterative algorithms for state estimation of jump Markov linear systems. *IEEE Transactions on Signal Processing*, *49*(6), 1216–1227. https://doi.org/10.1109/78.923304

Duan, P., Mao, G., Liang, W., & Zhang, D. (2019). A unified spatio-temporal model for short-term traffic flow prediction. *IEEE Transactions on Intelligent Transportation Systems*, *20*(9), 3212–3223. https://doi.org/10.1109/TITS.2018.2873137

Ermagun, A., & Levinson, D. (2018). Spatio-temporal traffic forecasting: review and proposed directions. *Transport Reviews*, *38*(6), 786–814. https://doi.org/10.1080/01441647.2018.1442887

Gorynin, I., Derrode, S., Monfrini, E., & Pieczynski, W. (2017a). Fast filtering in switching approximations of nonlinear Markov systems with applications to stochastic volatility. *IEEE Transactions on Automatic Control*, *62*(2), 853–862. https://doi.org/10.1109/TAC.2016.2569417

Gorynin, I., Derrode, S., Monfrini, E., & Pieczynski, W. (2017b). Fast smoothing in switching approximations of non-linear and non-Gaussian models. *Computational Statistics & Data Analysis*, *114*, 38–46. https://doi.org/10.1016/j.csda.2017.04.007

Guo, J., Huang, W., & Williams, B. M. (2014). Adaptive kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification. *Transportation Research Part C: Emerging Technologies*, *43*, 50–64. https://doi.org/10.1016/j.trc.2014.02.006

Haykin, S., & Li, L. (1995). Nonlinear adaptive prediction of nonstationary signals. *IEEE Transactions on Signal Processing*, *43*(2), 526–535.

Hu, X., Xu, X., Xiao, Y., Chen, H., He, S., Qin, J., & Heng, P.-A. (2019). Sinet: A scale-insensitive convolutional neural network for fast vehicle detection. *IEEE Transactions on Intelligent Transportation Systems*, *20*(3), 1010–1019. https://doi.org/10.1109/TITS.2018.2838132

Kamarianakis, Y., Oliver Gao, H., & Prastacos, P. (2010). Characterizing regimes in daily cycles of urban traffic using smooth-transition regressions. *Transportation Research Part C: Emerging Technologies*, *18*(5), 821–840. https://doi.org/10.1016/j.trc.2009.11.001

Kamarianakis, Y., & Prastacos, P. (2003). Forecasting traffic flow conditions in an urban network: Comparison of multivariate and univariate approaches. *Transportation Research Record*, *1857*(1), 74–84. https://doi.org/10.3141/1857-09

Kamarianakis, Y., Shen, W., & Wynter, L. (2012). Real-time road traffic forecasting using regime-switching space-time models and adaptive LASSO. *Applied Stochastic Models in Business and Industry*, *28*(4), 297–315. https://doi.org/10.1002/asmb.1937

Kumar, S., & Vanajakshi, L. (2015). Short-term traffic flow prediction using seasonal ARIMA model with limited input data. *European Transport Research Review*, *7*(3), 1–9. https://doi.org/10.1007/s12544-015-0170-8

Lange, T., & Rahbek, A. (2009). An introduction to regime switching time series models. In T. Mikosch, J. P. Kreiß, R. Davis & T. Andersen (Eds), *Handbook of financial time series. Springer, Berlin* (pp 871–887). Heidelberg. https://doi.org/10.1007/978-3-540-71297-8_38.

Lee, S., & Fambro, D. B. (1999). Application of subset autoregressive integrated moving average model for short-term freeway traffic volume forecasting. *Transportation Research Record*, *1678*(1), 179–188. https://doi.org/10.3141/1678-22

Li, Z., Li, Y., & Li, L. (2014). A comparison of detrending models and multi-regime models for traffic flow prediction. *IEEE Intelligent Transportation Systems Magazine*, *6*(4), 34–44.

Li, L., Qu, X., Zhang, J., Wang, Y., & Ran, B. (2019). Traffic speed prediction for intelligent transportation system based on a deep feature fusion model. *Journal of Intelligent Transportation Systems*, *23*(6), 605–616. https://doi.org/10.1080/15472450.2019.1583965

Lu, Z., Xia, J., Wang, M., Nie, Q., & Ou, J. (2020). Short-term traffic flow forecasting via multi-regime modeling and ensemble learning. *Applied Sciences*, *10*(1), 356. https://doi.org/10.3390/app10010356

Mei, H., Ma, A., Poslad, S., & Oshin, T. (2015). Short-term traffic volume prediction for sustainable transportation in

an urban area. *Journal of Computing in Civil Engineering*, *29*(2), 04014036. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000316

Min, W., & Wynter, L. (2011). Real-time road traffic prediction with spatio-temporal correlations. *Transportation Research Part C: Emerging Technologies*, *19*(4), 606–616. https://doi.org/10.1016/j.trc.2010.10.002

Moreno-Torres, J. G., Raeder, T., Alaiz-Rodríguez, R., Chawla, N. V., & Herrera, F. (2012). A unifying view on dataset shift in classification. *Pattern Recognition*, *45*(1), 521–530. https://doi.org/10.1016/j.patcog.2011.06.019

Mousavizadeh Kashi, S. O., & Akbarzadeh, M. (2019). A framework for short-term traffic flow forecasting using the combination of wavelet transformation and artificial neural networks. *Journal of Intelligent Transportation Systems*, *23*(1), 60–71. https://doi.org/10.1080/15472450.2018.1493929

Nagy, A., & Simon, V. (2018). Survey on traffic prediction in smart cities. *Pervasive and Mobile Computing*, *50*, 148–163. https://doi.org/10.1016/j.pmcj.2018.07.004

Pavlyuk, D. (2017). On application of regime-switching models for short-term traffic flow forecasting. In *Advances in dependability engineering of complex systems* (pp. 340–349). Springer.

Pieczynski, W. (2011). Exact filtering in conditionally Markov switching hidden linear models. *Comptes Rendus Mathematique*, *349*(9–10), 587–590. https://doi.org/10.1016/j.crma.2011.02.007

Qi, Y., & Ishak, S. (2014). A hidden Markov model for short term prediction of traffic conditions on freeways. *Transportation Research Part C: Emerging Technologies*, *43*(1), 95–111. https://doi.org/10.1016/j.trc.2014.02.007

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257–286. https://doi.org/10.1109/5.18626

Smith, B. L., Williams, B. M., & Oswald, R. K. (2002). Comparison of parametric and nonparametric models for traffic flow forecasting. *Transportation Research Part C: Emerging Technologies*, *10*(4), 303–321. https://doi.org/10.1016/S0968-090X(02)00009-8

Song, Z., Guo, Y., Wu, Y., & Ma, J. (2019). Short-term traffic speed prediction under different data collection time intervals using a SARIMA-SDGM hybrid prediction model. *PLoS One*, *14*(6), e0218626. https://doi.org/10.1371/journal.pone.0218626

Sun, H., Liu, H., Xiao, H., He, R., & Ran, B. (2003). Use of local linear regression model for short-term traffic forecasting. *Transportation Research Record*, *1836*(1), 143–150. https://doi.org/10.3141/1836-18

Sun, L., & Zhou, J. (2005). Development of multiregime speed-density relationships by cluster analysis. *Transportation Research Record*, *1934*(1), 64–71. https://doi.org/10.1177/0361198105193400107

Van Der Voort, M., Dougherty, M., & Watson, S. (1996). Combining Kohonen maps with ARIMA time series models to forecast traffic flow. *Transportation Research Part C: Emerging Technologies*, *4*(5), 307–318. https://doi.org/10.1016/S0968-090X(97)82903-8

Wang, J., & Shi, Q. (2013). Short-term traffic speed forecasting hybrid model based on chaos-wavelet analysis-support vector machine theory. *Transportation Research Part C: Emerging Technologies*, *27*(2), 219–232. https://doi.org/10.1016/j.trc.2012.08.004

Wang, C., & Ye, Z. (2016). Traffic flow forecasting based on a hybrid model. *Journal of Intelligent Transportation Systems*, *20*(5), 428–437. https://doi.org/10.1080/15472450.2015.1091735

Wei, D., & Liu, H. (2013). An adaptive-margin support vector regression for short-term traffic flow forecast. *Journal of Intelligent Transportation Systems*, *17*(4), 317–327. https://doi.org/10.1080/15472450.2013.771107

Widmer, G., & Kubat, M. (1996). Learning in the presence of concept drift and hidden contexts. *Machine Learning*, *23*(1), 69–101. https://doi.org/10.1007/BF00116900

Williams, B. M. (2001). Multivariate vehicular traffic flow prediction: evaluation of ARIMAX modeling. *Transportation Research Record*, *1776*(1), 194–200. https://doi.org/10.3141/1776-25

Williams, B. M., & Hoel, L. A. (2003). Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results. *Journal of Transportation Engineering*, *129*(6), 664–672. https://doi.org/10.1061/(ASCE)0733-947X(2003)129:6(664)

Wu, Y.-J., Chen, F., Lu, C.-T., & Yang, S. (2016). Urban traffic flow prediction using a spatio-temporal random effects model. *Journal of Intelligent Transportation Systems*, *20*(3), 282–293. https://doi.org/10.1080/15472450.2015.1072050

Xie, X., Yue, D., & Peng, C. (2017). Multi-instant observer design of discrete-time fuzzy systems: a ranking-based switching approach. *IEEE Transactions on Fuzzy Systems*, *25*(5), 1281–1292. https://doi.org/10.1109/TFUZZ.2016.2612260

Xie, Y., Zhao, K., Sun, Y., & Chen, D. (2010). Gaussian processes for short-term traffic volume forecasting. *Transportation Research Record*, *2165*(1), 69–78. https://doi.org/10.3141/2165-08

Xu, X., Jin, X., Xiao, D., Ma, C., & Wong, S. (2021). A hybrid autoregressive fractionally integrated moving average and nonlinear autoregressive neural network model for short-term traffic flow prediction. *Journal of Intelligent Transportation Systems*, 1–18. https://doi.org/10.1080/15472450.2021.1977639

Yu, S.-Z. (2015). *Hidden semi-Markov models: Theory, algorithms and applications* (1st ed.). Elsevier Science Publishers B. V.

Yu, G., & Zhang, C. (2004). *Switching ARIMA model based forecasting for traffic flow* [Paper presentation]. In 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing, May (Vol. 2, p. ii–429).

Zhang, Y., & Ye, Z. (2008). Short-term traffic flow forecasting using fuzzy logic system methods. *Journal of Intelligent Transportation Systems*, *12*(3), 102–112. https://doi.org/10.1080/15472450802262281

Zhang, Y., & Zhang, Y. (2016). A comparative study of three multivariate short-term freeway traffic flow forecasting methods with missing data. *Journal of Intelligent Transportation Systems*, *20*(3), 205–218. https://doi.org/10.1080/15472450.2016.1147813

Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., & Li, H. (2019). T-GCN: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, *21*(9), 3848–3858.

Zhao, J., & Sun, S. (2016). High-order Gaussian process dynamical models for traffic flow prediction. *IEEE*

Transactions on Intelligent Transportation Systems, *17*(7), 2014–2019. https://doi.org/10.1109/TITS.2016.2515105

Zheng, W., Lee, D., & Shi, Q. (2006). Short-term freeway traffic flow prediction: Bayesian combined neural network approach. *Journal of Transportation Engineering*, *132*(2), 114–121. https://doi.org/10.1061/(ASCE)0733-947X(2006)132:2(114)

Appendix A. Calculation of the maximum gradient matrices

Let us consider the following function

$$f(\boldsymbol{\xi},\sigma) = -\frac{1}{2}\sum_{n=1}^{M-1}\omega_n\left(\ln\left(2\pi\,\sigma^2\right) + \frac{\left(v_{n+1} - \boldsymbol{\xi}^\top \boldsymbol{u}_{n+1}\right)^2}{\sigma^2}\right)$$

Where $v_{n+1} \in \mathbb{R}$, $\boldsymbol{u}_{n+1}, \boldsymbol{\xi} \in \mathbb{R}^d$, and $\forall n \in [1, M-1]$, $0 \leq \omega_n \leq 1$.

Calculating the derivative of f *w.r.t.* $\boldsymbol{\xi}$ gives

$$\frac{\partial f(\boldsymbol{\xi},\sigma)}{\partial \boldsymbol{\xi}} = \frac{1}{\sigma^2}\sum_{n=1}^{M-1}\omega_n\left(v_{n+1} - \boldsymbol{\xi}^\top \boldsymbol{u}_{n+1}\right)\boldsymbol{u}_{n+1}^\top$$

Setting the result to zero yields

$$\boldsymbol{\xi}.^\top = \left[\sum_{n=1}^{M-1}\omega_n\, v_{n+1}\,\boldsymbol{u}_{n+1}^\top\right]\cdot\left[\sum_{n=1}^{M-1}\omega_n\,\boldsymbol{u}_{n+1}\boldsymbol{u}_{n+1}^\top\right]^{-1} \tag{A1}$$

Similarly, calculating the derivative of f *w.r.t.* σ gives

$$\frac{\partial f(\boldsymbol{\xi},\sigma)}{\partial \sigma} = -\frac{1}{\sigma}\sum_{n=1}^{M-1}\omega_n\left(1 - \frac{\left(v_{n+1} - \boldsymbol{\xi}^\top \boldsymbol{u}_{n+1}\right)^2}{\sigma^2}\right)$$

and, setting the result to zero yields

$$\dot{\sigma}^2 = \frac{\sum_{n=1}^{M-1}\omega_n\left(v_{n+1} - \boldsymbol{\xi}^\top \boldsymbol{u}_{n+1}\right)^2}{\sum_{n=1}^{M-1}\omega_n}$$

Thus, especially for $\boldsymbol{\xi} = \boldsymbol{\xi}.$, we get

$$\dot{\sigma}^2 = \frac{\sum_{n=1}^{M-1}\omega_n\left(v_{n+1} - \boldsymbol{\xi}.\,\boldsymbol{u}_{n+1}\right)^2}{\sum_{n=1}^{M-1}\omega_n} \tag{A2}$$

At this stage, we should check that the $(\boldsymbol{\xi}., \dot{\sigma})$-point that cancels the gradient corresponds to a maximum. This verification is beyond the scope of this paper, and we have only verified numerically that this is the case.