



HAL
open science

P-ADRIP: A multi-agent-based system for traffic forecasting

Ha-Nhi Ngo, Elsy Kaddoum, Marie-Pierre Gleizes, Jonathan Bonnet,
Goursolle Anaïs

► **To cite this version:**

Ha-Nhi Ngo, Elsy Kaddoum, Marie-Pierre Gleizes, Jonathan Bonnet, Goursolle Anaïs. P-ADRIP: A multi-agent-based system for traffic forecasting. 14th ITS European Congress Toulouse: Smart and Sustainable Mobility for all (2022), ERTICO ITS Europe; Toulouse Métropole, May 2022, Toulouse, France. hal-03764691

HAL Id: hal-03764691

<https://hal.science/hal-03764691v1>

Submitted on 30 Aug 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Paper ID #140

P-ADRIP: A multi-agent-based system for traffic forecasting

Ha-Nhi Ngo^{1,2*}, Elsy Kaddoum¹, Marie-Pierre Gleizes¹, Anaïs Goursole², Jonathan Bonnet²

1. IRIT- Université de Toulouse, CNRS, Toulouse INP, UT3, UT1, UT2, Toulouse, firstname.lastname@irit.fr

2. Continental Digital Services France, Toulouse, firstname.lastname@continental.com

Abstract

Traffic forecasting has gained more and more interests in both academic and industrial researches. The time series-based models are firstly applied to deal with linear dependency but cannot describe the non-linear and complex properties of traffic data. Recently, many methods for traffic forecasting based on machine learning and deep learning approaches are proposed. However, these models always encounter the unsolved questions relating to the reliability and the feasibility. Indeed, traffic forecasting is a very challenging task due to the complex spatial correlations in road network, the high-level time dependency and the difficulty of long-term prediction. To address the mentioned challenges, we propose a novel system based on multi-agent systems approach called P-ADRIP (Prediction subsystem - Adaptive multi-agent system for DRIVING behaviors Prediction) that aims to provide dynamic and real-time traffic prediction. The conducted experiments demonstrate the outstanding performance of ADRIP comparing to the state-of-the-art prediction methods.

Keywords:

Traffic prediction; Multi-agent-based solution; Local Cooperation, Time series prediction methods

Introduction

Accurate and real-time traffic forecasting is nowadays essential for urban traffic control, safety and guidance functions of Intelligent Transportation System (ITS). It is widely applied for various transportation services to make better travel recommendations, alleviate traffic congestion and dangerous collision at jam queue, reduce the consumed energy and improve the traffic efficiency. Thus, the accurate prediction becomes indispensable for many ITS subsystems including Advanced Driver-Assistance Systems (ADAS), Advanced Traveler Information System (ATIS), *etc.*

This task is challenging due to the complex spatio-temporal dependencies and the difficulty of long-term forecasting. On one hand, traffic data shows that strong temporal dynamics lead to heavy dependency of predicted values on the historical traffic data *i.e.*, temporal dependency. Thus, efficient forecasting models must include the analysis of different time series properties such as the non-stationarity, the seasonality, the non-linearity, *etc.* On the other hand, traffic dynamic contains also the complex spatial correlation. This paper introduces a Multi-Agent System (MAS) [1] based system for predicting traffic dynamics called P-ADRIP (Prediction subsystem - Adaptive multi-agent system for DRIVING behaviors Prediction). P-ADRIP implements a real-time and long-term prediction strategy

guaranteeing that the prediction of traffic dynamics is provided and updated all the time.

State of the Art

For traffic prediction, existing researches focused on the fundamental parameters of traffic flow such as mean speed, volume or density. The observations of these parameters are collected by fixed or mobile sensors at a position over time. The prediction horizon has various ranges adapting to different applications: short-term (5-30 min), medium and long-term (over 30 min). With the development of ITS technologies and modeling, many approaches for traffic prediction have been developed and are mainly classified into two categories: parametric models and non-parametric models. For each category, we aim at presenting the most used methods following the literature review in [2].

The **parametric models** are based on the time series models which express future values as linear combination of different terms such as previous data, random noise or seasonal property. These models are mostly applied for univariate time series referring to a time series that consists of single (scalar) observation recorded sequentially over time. Some extended versions have been also developed for multivariate time series. In traffic prediction applications, the extended models for multivariate time series allow to integrate the spatio-temporal relations with neighboring road segments to predict future values of considered location. Among the parametric models, the ARMA-based models have been proposed as well suited for application to short-term traffic prediction [3]. The ARMA-based models assume that future values of traffic information depend linearly only on the previous values (Auto-Regressive - AR) and the random noise series (Moving Average - MA) [4]. The ARIMA model (Auto-Regressive Integrated Moving Average), an extension of ARMA, can deal with non-stationary property by considering the differences between consecutive observations. ARIMA was applied in [4] to study the arterial travel time prediction problem. According to the results, this study has shown its promising effectiveness for travel time prediction. In [5], authors applied a subset of ARIMA for short-term traffic volume forecasting. The first limitation of ARIMA model is that it can only deal for the non-stationary problem. However, traffic prediction models require to integrate other relevant properties of traffic data to obtain the more accurate estimation: (1) the seasonality that is characterized by the periodic cycles of traffic data, (2) the spatio-temporal relationship between neighboring road segments.

The **VARMA** (Vector AutoRegressive Moving Average) is a multivariate model which can estimate the dynamic interactions between multiple time series. In traffic prediction application, this model is applied to consider the impact of the measures of neighboring segments on current segment. The results in [6] showed the significant improvement of prediction performance when using multivariate time series model in cases of large road network and high number of installed loop detectors.

Despite the improvement of parametric models to deal with specific properties of traffic data, their biggest drawback is that they cannot solve non-linear problems. They achieve good performances when traffic has regular variations, but prediction error becomes significant when irregular situations occur.

Non-parametric models use historical data to build and train models which express the impacts of variables on future values. Many non-parametric models have been recently applied for traffic forecasting. In the next part, for comparison issues, we focus on models whose principle is similar to P-

ADRIP such as the **K-nearest neighbor model (KNN)**, and which can address the important challenges of time series analysis.

In traffic problem applications, KNN finds the k -closest historical traffic states of current traffic state and predict the future states regarding the next states of k neighbors. In [7], KNN was firstly applied for traffic flow prediction. The high accuracy shows the feasibility of proposed method for short-term traffic flow prediction without data constraints. An improved version of KNN for traffic prediction was introduced in [8]. The spatio-temporal correlation is highlighted to consider the impacts of neighboring roads. As a non-parametric method, KNN can deal with the non-linear and non-stationary traffic data. A serious drawback of KNN concerns the choice of the optimal value of the parameters such as k (number of nearest neighbors) or distance metric.

The simple type of Neural Networks (NN) called **Feed Forward Neural Network (FFNN)** is applied in [9] to estimate the traffic flow in one or multiple next steps for one or multiple road links. With the development of deep learning architecture, many researches have proven that the multiple hidden layers are more capable of capturing the non-linear dependencies between input variables and output than single hidden layer. The authors of [10] conducted the study which used a NN with stack of hidden layers to predict the traffic speed from the 40-previous speed measures of a set of sensors. The obtained results showed that the MSE (Mean Squared Error) decreases by 14% compared to the traditional NN with one hidden layer. A NN with single or multiple hidden layers is a powerful approach for short-term traffic prediction thanks to the ability of modeling non-linear functions. NN is flexible for integrating the environment variables, other traffic information or the traffic parameters of neighboring road/sensor. However, capturing the complex and long-term dependencies of traffic time series data is a big challenge for NN due to the increasing of model complexity. Indeed, more previous measures are considered leading to the high dimension problem where the NN is not an appropriate solution. This drawback causes that few researches used NN to analyze the spatio-temporal relations in which the number of variables increases exponentially with the number of considered neighboring roads.

[11] presented an experiment which compares the performance of ARIMA versus **Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU)** for traffic flow prediction. The results showed that the variations of RNN can reduce the error at about 10% than ARIMA models. In [12], LSTM network with multiple hidden layers was presented outperforming the ARIMA model, traditional NNs and RNNs. The results confirmed the robustness of LSTM in terms of capturing long time-dependency. However, in case of large road network, considering the long-term dependency and all correlations between road segments makes RNNs model sometimes impractical due to the problem caused by the vanishing or exploding gradient.

From the above state of the art, we can **highlight three points**. First, the challenging of long-term traffic prediction is affirmed and remains an unsolved question although many methods have been investigated it. Indeed, the existing methods mainly focus on establishing a model whose design is chosen a priori and the parameters are calibrated from historical data. Thus, the models that aim to address many properties in traffic data, have to complicate their design and consider many parameters leading to increase the computational capacity requirement. Second, most existing methods only study the training

data set to deduce the parameters model. Thus, when testing with new data sets with new traffic behavior, the model performance might be degraded since the model has not encountered these perceptions yet. Therefore, the ability of updating parameters through each perception is an important point to evaluate its adequacy. Third, existing researches focus on predicting the fundamental traffic parameters such as mean speed or mean traffic density within a large time window (ex.5 minutes). However, these parameters are not sufficient to represent traffic dynamic or driving behaviors on a road segment. For example, we cannot deduce the variation of vehicle's speed on the whole road segment.

To address these issues, we propose the P-ADRIP.

ADRIP System

ADRIP (Adaptive multi-agent system for DRIVING behaviors Prediction) aims at dealing with the traffic dynamic forecasting problem which can be described as:

- A set of vehicles $V = v_1; v_2; \dots; v_m$. Each vehicle follows an itinerary I divided into a sequence of road segments noted $I = \{rds_1, \dots, rds_N\}$.
- A set of road segments determined according to road network in Open Street Map (OSM) or simulated road scenarios. Each road segment is characterized by a starting and an ending position that can be located by GPS devices.

ADRIP is decomposed into two subsystems which function in parallel and in real-time:

- L-ADRIP [13] the lifelong learning subsystem dynamically classifies **the driving behaviors** of vehicles on each road segment based on the received data stream from vehicles.
- P-ADRIP the prediction subsystem which is the focus of this paper: predicts the driving behaviors on a road segment for a given prediction horizon based on the learned data.

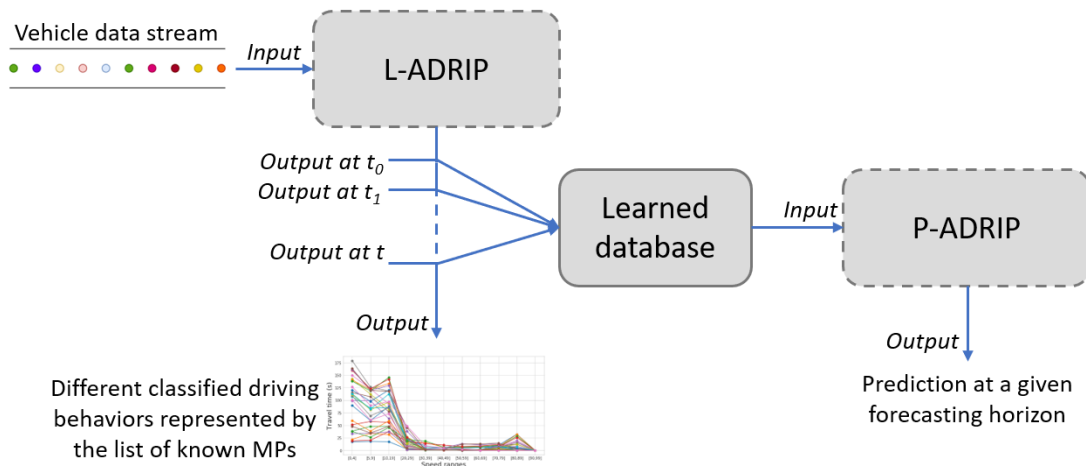


Figure 1: Illustration of ADRIP

The global architecture of ADRIP is shown figure 1. During runtime, **L-ADRIP subsystem** provides and maintains up-to-date a local database to each road segment. This local database contains different driving behaviors called **Mobility Profile (MP)** with their associated time intervals called **Ranges of Use (RU)** during which the vehicles moved with each driving behavior on this road segment. The MP is defined as the distribution of travel time at different speed ranges. Each MP has a list of RUs indicating

the moments where vehicles in a fleet moved on this road segment with this MP. The learned database of L-ADRIP is composed of the different local databases learned at each road segment. This learning mechanism is achieved by an adaptive multi-agent system described in [13].

P-ADRIP subsystem perceives the current situation and uses learned databases to predict the future driving behaviors for a given horizon of time. P-ADRIP predictions are updated whenever changes in the current situation or in the learned database occur. In P-ADRIP, the traffic forecasting prediction on a road network is performed by a multi-agent-based system composed of **Segment Agents (SA)**. Each SA is associated to a road segment of the road network and predicts the traffic on this road segment.

Segment Agents in P-ADRIP

In P-ADRIP, SA aims at estimating its future traffic dynamics until a given prediction horizon. To do that, SA investigates the *temporal dependency* and the *spatial correlation*. First, the temporal dependency is integrated by analyzing its own learned database obtained from L-ADRIP. Second, the spatial correlation is considered using the exchanged information between SA and the SAs associated with the upstream and downstream road segments. The information of neighboring SAs constitutes a part of the input of prediction algorithm defined as **Configuration**: *the configuration at the instant T under the point of view of a road segment is the set of the MPs with their corresponding RU at T of its neighboring road segments.*

The SA behavior can be described using two main parts: the nominal behavior and the cooperative behavior. **The nominal behavior** consists of computing the chain of the different predictive changes of MPs as the prediction for a forecasting horizon. From the current timestamp (T_s), SA launches the prediction algorithm whose main steps are illustrated in figure 2 to predict the next MP and the timestamp in the future when this change occurs. This timestamp is also the beginning of the predicted next MP. If the changing timestamp does not reach the demanded prediction horizon, SA launches again the prediction algorithm using the last computed next MP and its associated timestamp as inputs. This process is repeated until SA reaches the prediction horizon or farther.

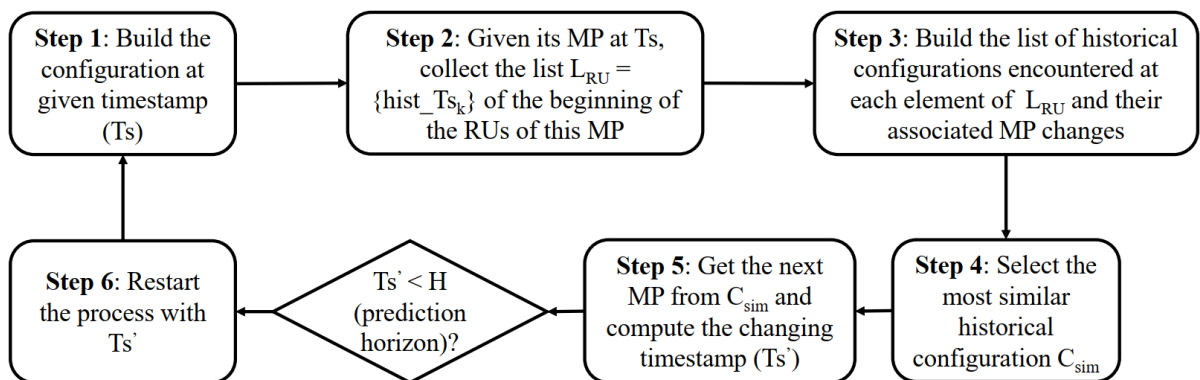


Figure 2: The main steps of the prediction process of SA

In step 4, SA compares the configuration at T_s and the historical configurations to select the most similar one. Thus, for each historical configuration computed at step 3, SA compares it to the configuration at

Ts and computes two criteria:

1. The first criterion is the number of neighboring SAs with different MPs among both configurations.
2. The second criterion refers to the comparison of the time gap between the beginning of MP of neighboring SAs and the beginning of MP of considered SA. Indeed, since the change of traffic dynamic on a road segment can be resulted from the changes of traffic dynamics on its neighbors, this criterion aims to evaluate the difference between two configurations of the time of dynamics propagation. Figure 3 illustrates the calculation of this criterion.

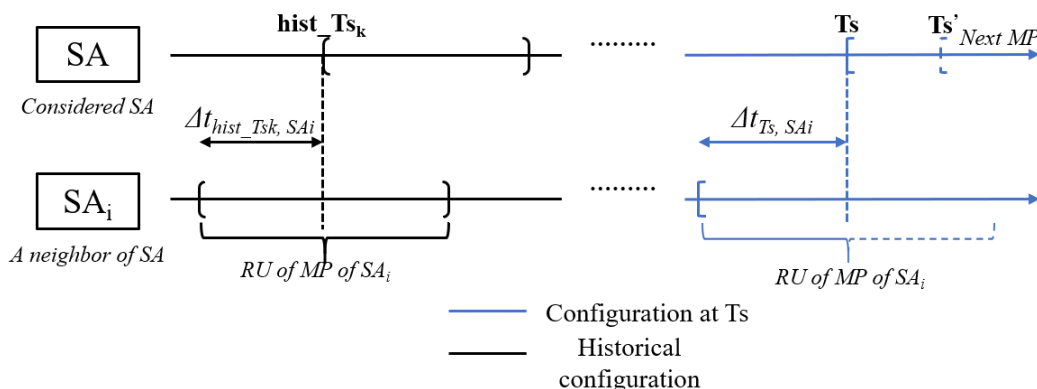


Figure 3: Illustration of the calculation of second criterion

For each SA_i belonging to the set of neighboring SAs, $\Delta t_{Ts, SA_i}$ is the time distance between the beginning of MP on SA (T_s) and the beginning of the MP of SA_i observed at T_s . Similarly, $\Delta t_{hist_Ts_k, SA_i}$ is the time distance between the beginning of MP of SA at a historical RU ($hist_Ts_k$) and the beginning of the MP of SA_i observed at $hist_Ts_k$. Thus, the second criterion is expressed as follow:

$$c_2 \left(Config_{T_s}, Config_{hist_{Ts_k}} \right) = \sum_{i=1}^M |\Delta t_{T_s, SA_i} - \Delta t_{hist_{Ts_k}, SA_i}| \quad (M = \text{number of neighboring SAs})$$

The most similar historical configuration with the configuration at T_s is the one minimizing both criteria. In step 5, the change of MP associated with the most similar historical configuration constitutes the predictive next MP of SA at T_s' . T_s' is computed by shifting forwardly T_s for a time interval equal to the size of RU associated with the chosen configuration. Finally, if T_s' reaches the prediction horizon, SA stops the algorithm, otherwise, it restarts the process computing the prediction at T_s' and stores next MP in the chain of the different predictive changes of MPs.

During its nominal behavior, SAs encounter specific situations which disturb their performance. The cooperative behaviors are defined to deal with these specific situations as follow:

- SA cannot build the complete configuration required at step 1 since some neighbors have not estimated yet their prediction at the given timestamp, thus they cannot send it to SA.

Proposed solution: SA waits for the neighboring SAs which have shorter current prediction horizon than its to launch their prediction process before. When the predictions in these SAs are updated and the prediction horizon of SA become shorter than its neighboring SAs, it can build the complete required configuration and start its prediction process.

- SA cannot propose a candidate for the prediction since the configuration built at step 1 contains a new MP for SA that was never perceived or learned by L-ADRIP.

Proposed solution: The dynamic learning process of SA (performed by L-ADRIP) will learn this new MP and update its local database. The update of learned database expands the system's knowledge and ensures that this issue can be handled. This behavior characterizes the openness property of system.

Experiments

In this section, we present the conducted experimentations to validate ADRIP behavior and compare P-ADRIP with well-known traffic prediction models.

Data Generation

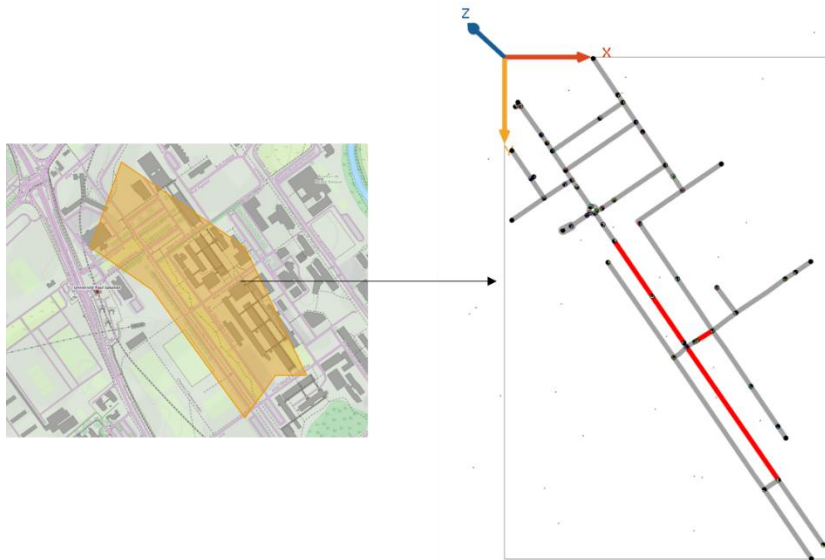


Figure 4: The scenario from OSM (left) and the projection of chosen zone in GAMA

We conduct the experiments using GAMA (GIS Agent-based Modelling Architecture) platform [14]. GAMA allows to load road networks from shape files or OSM (Open Street Map) files to perform the simulation on a real road network. For this study, the scenario locates at the campus of University Toulouse III in Toulouse consisting of 63 road segments (cf. figure 4). The properties of SAs are taken from the OSM [15]. The behavior of vehicles is defined following the skill proposed by [16] named Advanced Driving Skill. The number of vehicles at every instant of the simulation is set to 200 which allows to present a diversity of traffic dynamics. Once a vehicle finishes its trajectory, it is replaced by a new one to ensure the total number of vehicles during the simulation being always 200. Each vehicle starts at a random chosen position and crosses the road segments following a randomly taken trajectory. Thus, although the total number of agents is fixed, the traffic flow on each road segment varies through time leading to different driving behaviors. The traffic was simulated on this scenario during 3 hours for the learning data set and then for 1 hour for the testing data set, totalizing 9520 vehicle trajectories.

Data Preprocessing

From the simulation of the considered scenario, we obtain the trajectory data of vehicle e.g. speed at each GPS position identified by the longitude and the latitude. However, the methods in the literature

review are tested using the data forming the time series of average speed on road segments at each f minutes (f is the frequency of time series) while ADRIP works on the Mobility Profiles (MP). Thus, a data preprocessing step is required to prepare the data sets for the comparison between the different considered methods.

First, we convert the trajectory data into time series of mean speed by averaging all vehicle's speeds on each road segment for every 10 seconds, obtaining by that time series of mean speed at 10s frequency for all road segments. In addition, collected data also need to be normalized using *MinMaxScaler* to avoid degrading the performance of machine learning and deep learning models during the optimization of gradient for FFNN, LSTM, GRU and to ensure the equal weight for each variable in the distance measure for KNN. Besides that, the transformation function *StandardScaler* is applied for the VARMA and ARIMA models to ensure the consistent scale of the data in different time series. These functions are implemented in the library *sklearn* of Python.

Second, for ADRIP, the trajectory data of vehicles is segmented according to the road network. Then, the segmented data on each road segment is transformed into the MP. MP is the distribution of travel time on 7 speed ranges which are [0%, 5%], [5%, 10%], [10%, 20%], [20%, 30%], [30%, 40%], [40%, 60%], [60%, 200%] of the limited speed on road segment. The last speed range is extended at 200% of limited speed to cover some exceeding speeds which happen sometimes. For example, if the limited speed on road segment is 30 km/h, the speed ranges are [0, 1.5], [1.5, 3.0], [3.0, 6.0], [6.0, 9.0], [9.0, 12.0], [12.0, 18.0], [18.0, 60.0]. We observe the vehicle's speed through time and count the travel time corresponding to each speed range. Note that, those MPs are constructed by vehicle agent and used at runtime by the L-ADRIP subsystem to learn local historical databases at each road segment.

Parameter settings

For **ARIMA**, the orders are (30, 0, 1), and the model is implemented using the *statsmodel* python package. The parameters are adopted from the paper [17] except the number of lags (the number of used previous values) which is at 30 to adapt to our experiment design.

For **VARMA**, the model is applied with the number of lags set to 30. VARMA is implemented using the *statsmodel* python package.

For **KNN**, the optimal value of k (number of nearest neighbors) is fixed to 18 according to [7].

For **FFNN**, Feed Forward Neural Network with two hidden layers, each layer contains 256 units. The learning rate is $1e^{-3}$, the dropout rate is 0.5 and the decay rate $1e^{-2}$. The model is trained with batch size 64 and the algorithm optimization of Adam is used with the MAE (Mean Absolute Error) as the loss function. The parameters are adopted from [17].

For **LSTM**, **GRU**¹, the cells of LSTM and GRU models are implemented using the *keras.layers.recurrent* package. These models have two hidden cells with 64 units, a dropout layer set at 0.2 and a dense layer with the number of units at the output size.

Evaluation metrics

To compare the performance of the considered methods for mean speed prediction, two metrics are

¹ The Python code can be found at : <https://github.com/xiaochus/TrafficFlowPrediction>

adopted: MAE and RMSE (Root Mean Squared Error).

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}$$

where y_i is the true value, \hat{y}_i is the predicted value and N is the number of data points.

In addition, to evaluate the performance of ADRIP for the prediction of MP, we define the **MP difference** metric.

$$MPDiff(MP^k, MP^l) = (|MP_i^k - MP_i^l|)_{i=1, \dots, N}$$

where N is the number of speed ranges, MP_i^k and MP_i^l are the values of time travel corresponding to the i^{th} speed range.

Results and analysis

We first compare ADRIP with the chosen methods for the prediction of mean speed for the next 5 minutes. The compared methods select the mean speed of the past 5 minutes as input. Note that, although the prediction horizon is just of 5 minutes, our experiment can be considered as a long-term test regarding to the data window at 10s since this horizon corresponds to 30 next data points. Unlike the tests conducted in other papers, we aim at obtaining all the predictive values between the current timestamps until the next 5 minutes. Thus, the response variable is a vector of size 30.

We select 4 road segments (shown in red in figure 4) from the testing scenario which have the most diversity of traffic dynamics. Indeed, these road segments are interesting for the evaluation of traffic forecasting methods since they can show whether the methods can capture the complex evolution of traffic data. Table 1 summarizes the obtained results. Note that the obtained prediction of ADRIP is the MP, thus, the predicted mean speeds at each 10s are deduced using the following expression:

$$\bar{v}_t = \frac{l}{\sum_{i=1}^N MP_t^i}$$

where \bar{v}_t is the mean speed at t , l is the length of road segment, N is the number of speed range in MP and MP_t^i is the travel time of vehicle with the i^{th} speed range of the MP at the timestamp t . The $\sum_{i=1}^N MP_t^i$ gives us the total travel time when vehicles move with the MP.

Table 1: Average MAE and RMSE for road segments obtained by the state-of-the-art methods and ADRIP

	Metric	VARMA	ARIMA	KNN	FFNN	LSTM	GRU	ADRIP
Average on	MAE	1.51	1.51	1.50	1.59	1.58	1.72	1.31
4 roads	RMSE	2.42	2.36	2.40	2.54	2.52	2.69	1.77
1 st road segment	MAE	1.05	1.09	0.96	1.02	0.95	0.96	1.00
	RMSE	1.80	1.68	1.62	1.72	1.59	1.72	1.41
2 nd road segment	MAE	1.10	1.16	1.08	1.13	1.03	1.05	0.91
	RMSE	2.15	2.10	2.09	2.11	2.05	2.05	1.42
3 rd road segment	MAE	2.44	2.41	2.58	2.80	2.95	3.49	1.80
	RMSE	3.82	3.77	4.05	4.42	4.57	5.28	2.40
4 th road segment	MAE	1.43	1.40	1.38	1.41	1.42	1.38	1.52
	RMSE	1.91	1.90	1.82	1.89	1.88	1.82	1.83

First, the average errors on 4 road segments show that P-ADRIP achieves the best average performance regarding both evaluation metrics, which refers to the effectiveness of the spatio-temporal dependency modelling. Second, KNN - a simple non-parametric model is slightly better than the parametric models (ARIMA and VARMA) that emphasizes the importance of non-linearity dependency consideration. Third, FFNN and RNN-based models (LSTM, GRU) give poorer performances than KNN and the linear models. This can be due to the degradation caused by the model complexity or the insufficiency of training data set. This remark enhances the advantage of P-ADRIP as an efficient solution for the complex spatio-temporal dependency in traffic forecasting with a reasonable size of data set.

Next, considering the detail of model performance on each road segment, we observe the following remarks. For the 2nd and 3rd road segments, P-ADRIP outperforms the state-of-the-art methods for both evaluation metrics that is coherent with the interpretation of average results. However, for the 1st and 4th road segments, P-ADRIP sometimes gives the lower RMSE but higher MAE comparing to other models. Since RMSE penalizes the large error more harshly than MAE, we can deduce from this phenomenon that, ADRIP makes more small-scale error but less large-scale error than others.

Then, we evaluate the prediction accuracy of P-ADRIP for different horizons. Figure 5 shows the average prediction error for each speed range on the 4 chosen road segments for the predictions at 2 minutes, 3 minutes, 4 minutes and 5 minutes. We observe that P-ADRIP does not highly increase the prediction error when extending the prediction horizon. This result allows to enhance the ability of ADRIP for long-term traffic forecasting.

Conclusions and perspectives

In this paper, we propose a prediction system for traffic dynamics, P-ADRIP which provides drivers on the prediction of MP, which represents traffic dynamics in the scope of our research. Coupled with L-ADRIP introduced in [3], P-ADRIP completes the global functionality of ADRIP. P-ADRIP shows its ability of dealing with spatio-temporal dependency issues of traffic forecasting based on two main mechanisms. First, the temporal dependency is studied by analyzing the local learned historical database at each segment road. Second, the analysis of spatial dependency is based on the exchanged information among neighboring segment roads.

P-ADRIP was compared with state-of-the-art methods using generated data by GAMA. The comparison test for mean speed prediction shows that P-ADRIP achieves better performance in addition of providing the prediction of MPs which is a more adequate representation of traffic dynamics than mean speed.

In comparison with state-of-the-art methods, our solution has gained some advantages relating to (1) integrate many traffic data properties such as non-linearity, spatio-temporal dependency while maintaining the model complexity thanks to decentralized decisions, (2) include the free-parameter models, (3) enable the real-time and dynamic prediction, (4) evolve the decision according to the evolution of environment and (5) require more tolerant calculation time and computation capacity thanks to the distributed mechanism.

For future works, we firstly aim to apply P-ADRIP on real-world large-scale datasets to reinforce its efficiency and reliability. Then, we will integrate a control mechanism for P-ADRIP which can

automatically detect, correct the prediction error and update it in real-time.

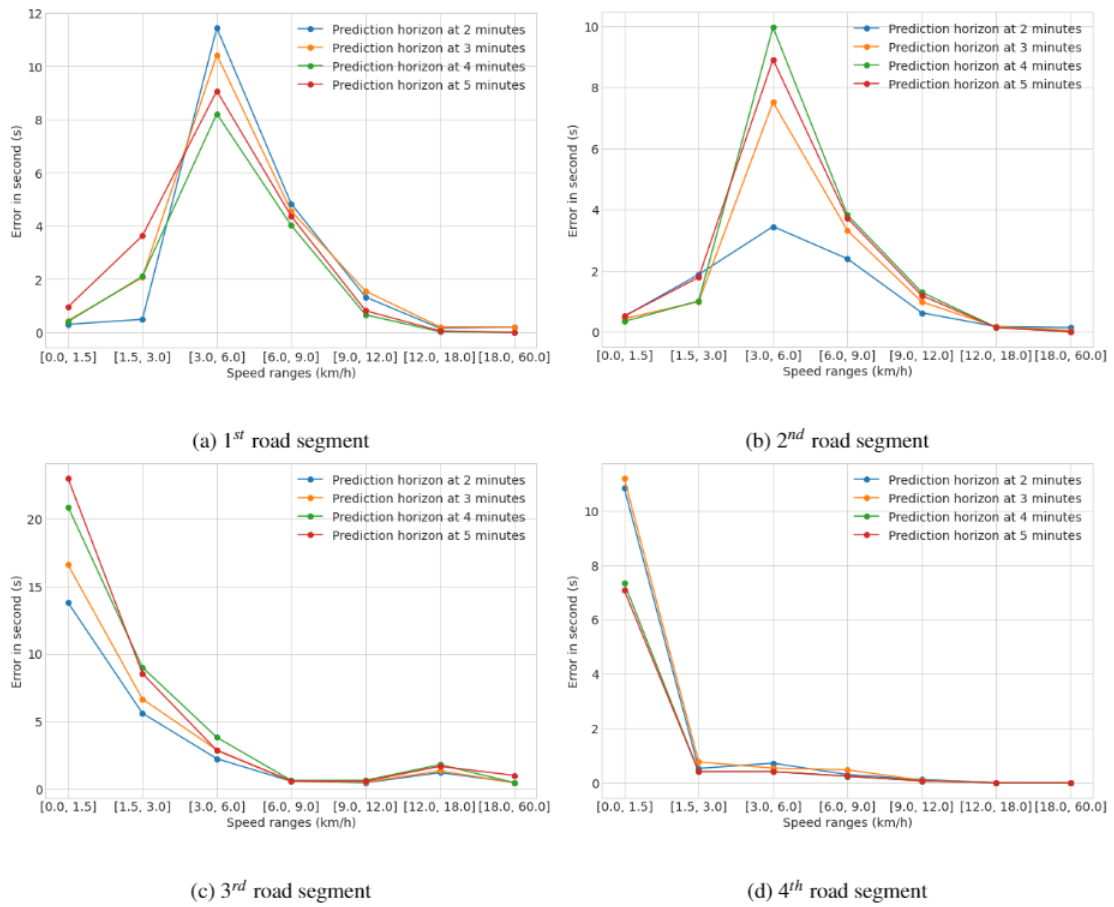


Figure 5: The mean prediction error for each speed range for different prediction horizons

References

1. Di Marzo Serugendo, G., Gleizes, M.-P., & Karageorgos, A. (2011). Self-organising software: From natural to artificial adaptation.
2. Nagy, A. M., & Simon, V. (2018). Survey on traffic prediction in smart cities. *Pervasive and Mobile Computing*, 50, 148–163.
3. 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).
4. Billings, D., & Yang, J.-S. (2006). Application of the ARIMA models to urban roadway travel time prediction-a case study. In *IEEE International Conference on Systems, Man and Cybernetics*, vol. 3.
5. 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.
6. Kamarianakis, Y., & Prastacos, P. (2003). Forecasting traffic flow conditions in an urban network:

- Comparison of multivariate and univariate approaches. *Transportation Research Record*, 1857(1).
7. Zhang, L., Liu, Q., Yang, W., Wei, N., & Dong, D. (2013). An improved k-nearest neighbor model for short-term traffic flow prediction. *Procedia-Social and Behavioral Sciences*, 96, 653–662.
 8. Cai, P., Wang, Y., Lu, G., Chen, P., Ding, C., & Sun, J. (2016). A spatiotemporal correlative k-nearest neighbor model for short-term traffic multistep forecasting. *Transportation Research Part C: Emerging Technologies*, 62, 21–34.
 9. Sun, S., Huang, R., & Gao, Y. (2012). Network-scale traffic modeling and forecasting with graphical lasso and neural networks. *Journal of Transportation Engineering*, 138(11), 1358–1367.
 10. Polson, N. G., & Sokolov, V. O. (2017). Deep learning for short-term traffic flow prediction. *Transportation Research Part C: Emerging Technologies*, 79, 1–17.
 11. Fu, R., Zhang, Z., & Li, L. (2016). Using lstm and gru neural network methods for traffic flow prediction. In 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), (pp. 324–328). IEEE.
 12. Zhao, Z., Chen, W., Wu, X., Chen, P. C., & Liu, J. (2017). Lstm network: a deep learning approach for short-term traffic forecast. *IET Intelligent Transport Systems*, 11(2), 68–75.
 13. Ngo, H.-N., Kaddoum, E., Gleizes, M.-P., Bonnet, J., & Goursolle, A. (2021). Life-long learning system of driving behaviors from vehicle data streams. In 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), (pp. 1132–1139). IEEE.
 14. Grignard, A., Taillandier, P., Gaudou, B., Vo, D. A., Huynh, N. Q., & Drogoul, A. (2013). Gama 1.6: Advancing the art of complex agent-based modeling and simulation. In *International conference on principles and practice of multi-agent systems*, (pp. 117–131). Springer.
 15. Haklay, M., & Weber, P. (2008). Openstreetmap: User-generated street maps. *IEEE Pervasive computing*, 7(4), 12–18.
 16. Taillandier, P. (2014). Traffic simulation with the gama platform. In *Eighth International Workshop on Agents in Traffic and Transportation*, (pp. 8–p).
 17. Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2017). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*.