

# Comparison of Parametric and Non-Parametric Approaches for Vehicle Speed Prediction

Stéphanie Lefèvre, Chao Sun, Ruzena Bajcsy, and Christian Laugier

**Abstract**—Predicting the future speed of the ego-vehicle is a necessary component of many Intelligent Transportation Systems (ITS) applications, in particular for safety and energy management systems. In the last four decades many parametric speed prediction models have been proposed, the most advanced ones being developed for use in traffic simulators. More recently non-parametric approaches have been applied to closely related problems in robotics. This paper presents a comparative evaluation of parametric and non-parametric approaches for speed prediction during highway driving. Real driving data is used for the evaluation, and both short-term and long-term predictions are tested. The results show that the relative performance of the different models vary strongly with the prediction horizon. This should be taken into account when selecting a prediction model for a given ITS application.

## I. INTRODUCTION

A large number of Intelligent Transportation Systems (ITS) applications rely on the prediction of the evolution of the local traffic situation. Safety applications in particular are strongly dependent on prediction schemes. For example lane departure warning involves predicting the future position of the ego-vehicle with respect to the lane. Another example is collision avoidance systems, where the chance of a future collision has to be assessed. Energy management is another domain where prediction plays an important role, since estimates of the future speed of the ego-vehicle can be used to optimize the energy consumption of the engine.

This paper focuses on a problem common to all the applications listed above: predicting the future speed of the ego-vehicle. This is a challenging task, as it involves modeling how human drivers react to arbitrary traffic situations. The underlying dynamics of human perception, information processing, decision making, and action execution are extremely complex and not fully understood. Over the past four decades a number of simplified models have been proposed. Some of them model the driving task as a stimulus-response system, others as a control problem where the driver's goal is to

keep a safe distance with the vehicle in front. More recently some non-parametric models have been proposed, allowing a greater flexibility in the representation of the dynamics. In this work we run a comparative evaluation between two non-parametric approaches and four parametric approaches. The evaluation is performed for different prediction horizons, so that the results can be used to draw conclusions about the adequacy of each model for different ITS applications.

A number of studies have been presented in the past comparing the performance of the most commonly used parametric models, without considering non-parametric approaches [1], [2], [3]. Bonsall et al. [4] went further and investigated the role of the choice of parameter values. The results point out the importance of deriving parameter values from observations, not only theory, in order to have realistic models. Angkititrakul et al. [5] compared the performance of two non-parametric methods for the prediction of pedal operation, namely Gaussian Mixture Models and a combination of Support Vector Machine classification with Piecewise Auto Regressive exogenous models. The results show similar performance for the two approaches, and highlight that heavy traffic conditions are challenging for speed prediction. Finally, Panwai et al. [6] compared a non-parametric approach (Artificial Neural Network) with several classic parametric methods. Since the targeted application was traffic simulation and not prediction, no results were provided about the performance of these models in the absence of measurements over the prediction period.

The remainder of this paper is organized as follows. Section II describes the objectives of the paper and formulates the problem. Section III introduces the models implemented for this study. The models are evaluated with real data for different prediction horizons, following the methodology described in Section IV. The results are presented and analyzed in Section V. Finally, Section VI concludes the paper and outlines future work.

## II. PROBLEM STATEMENT

### A. Objective

This paper compares six approaches to predict the most likely future speed of the ego-vehicle, with a focus on lane keeping situations on the highway. Among the prediction approaches tested, four are parametric and two are non-parametric.

- *Parametric* methods make an a priori assumption about the structure of the prediction model. Once the analytical form of the prediction function has been defined,

S. Lefèvre is with the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, USA (slefevre@berkeley.edu) and with Inria Grenoble Rhône-Alpes, France.

C. Sun is with the Department of Mechanical Engineering, University of California at Berkeley, USA (chaosun.email@gmail.com) and with the National Engineering Laboratory for Electric Vehicles, Beijing Institute of Technology, China

R. Bajcsy is with the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, USA (bajcsy@eecs.berkeley.edu).

C. Laugier is with Inria Grenoble Rhône-Alpes, France (christian.laugier@inria.fr).

the model parameters can be learned from data. There is a wide range of parametric approaches to model the speed of a vehicle on the highway, ranging from linear constant input models generally used for short-term prediction to more advanced models typically developed for microscopic traffic simulation. In this work we implement four models: Constant Speed (CS), Constant Acceleration (CA), SUMO simulator model (SUMO), and Intelligent Driver Model (IDM).

- *Non-parametric* methods do not fix the model structure in advance, but determine it from data instead. This implies that the number and nature of the parameters will vary depending on the training data. Non-parametric approaches are particularly useful for modeling complex systems whose underlying physics are not well understood. The “driver + car” combination is an example of such systems, as well as most systems which involve humans making decisions and interacting with a physical entity. In this work we implement two models: Gaussian Mixture Regression (GMR), and Artificial Neural Network (ANN).

The six approaches considered in this work are compared for different prediction horizons, so as to evaluate their adequacy for different intelligent transportation applications. For example, collision avoidance systems focus on short-term prediction (1 to 3 s) due to the high unpredictability of what other drivers may do [7]. On the other hand, applications related to energy management in vehicles are interested in longer-term speed prediction, up to 10 s [8].

### B. Framework for iterative prediction

To test the different prediction models and the different prediction horizons, we define a common framework to iteratively predict the future speed of the ego-vehicle<sup>1</sup>.

The situation at time  $t$  is represented by the following state variables:

$$x(t) = [d(t), r(t), s(t), a(t)]^T \quad (1)$$

where  $d(t) \in \mathbb{R}$  is the distance to the vehicle in front at time  $t$  in meters,  $r(t) \in \mathbb{R}$  is the relative speed to the vehicle in front at time  $t$  in m/s,  $s(t) \in \mathbb{R}$  is the speed of the ego-vehicle at time  $t$  in m/s, and  $a(t) \in \mathbb{R}$  is the acceleration of the ego-vehicle at time  $t$  in m/s<sup>2</sup>. In practice, the current speed and acceleration of the ego-vehicle can be obtained from proprioceptive sensors. The current distance and relative speed to the vehicle in front can be obtained from forward-looking exteroceptive sensors such as a camera or a radar, which are becoming more and more common in passenger vehicles.

The state transition equations are derived assuming a constant speed model for the vehicle in front (since no information is available at time  $t$  about the future behavior of

that vehicle) and a constant acceleration within each update time interval  $\Delta t$  [9]:

$$d(t + \Delta t) = d(t) + r(t) \cdot \Delta t - \frac{1}{2} \cdot a(t) \cdot \Delta t^2 \quad (2)$$

$$r(t + \Delta t) = r(t) - a(t) \cdot \Delta t \quad (3)$$

$$s(t + \Delta t) = s(t) + a(t) \cdot \Delta t \quad (4)$$

$$a(t + \Delta t) = f(x(0), \dots, x(t)) \quad (5)$$

The function  $f$ , which maps the past and current states to the acceleration at the next timestep, can take different forms. In this work we implement six models for  $f$ , as described in the next section.

## III. MODELS

The six models implemented in this work to predict the future acceleration of the ego-vehicle from the past and current states are described below. The parameters of each model are learned from real data during a training phase; this will be detailed in Section IV.

### A. Constant Speed model

The Constant Speed (CS) model assumes that the ego-vehicle will keep the same speed, i.e.:

$$a(t + \Delta t) = 0 \quad (6)$$

This model is popular in collision prediction applications. It is the basis of the computation of the Time-To-Collision (TTC) [10], which is a popular metric for risk assessment. The CS model is also commonly used in vehicle tracking applications, in particular when no information about the environment (e.g. other vehicles) is available. Like all linear models, it has the advantage that the propagation of Gaussian uncertainties during tracking is straightforward.

### B. Constant Acceleration model

The Constant Acceleration (CA) model assumes that the ego-vehicle will keep the same acceleration, i.e.:

$$a(t + \Delta t) = a(t) \quad (7)$$

Like the CS model, and for the same reasons, it is popular for collision prediction and vehicle tracking applications.

### C. SUMO model

SUMO is a well-known microscopic traffic simulator developed by the German Aerospace Center [11]. The lane keeping behavior of a vehicle is modeled based on the work by Krauss [12], which itself is derived from the well known Gipps’s model [13]. It is based on the computation of a “safe distance” between the ego-vehicle and the vehicle in front. A “safe distance” means that the ego-vehicle is able to stop at any time without colliding with the vehicle in front. A “safe speed” for the ego-vehicle can then be derived as:

$$s_{safe}(t + \Delta t) = -\tau \cdot b_{mx} + \sqrt{(\tau \cdot b_{mx})^2 + s_f(t)^2 + 2 \cdot b_{mx} \cdot d(t)}$$

<sup>1</sup>As an alternative to *iterative* prediction, the GMR and ANN approaches could be used to perform *direct* prediction or *multiple-output* prediction. However the four other approaches were designed for iterative prediction, therefore it was selected as common prediction scheme.

where  $\tau$  is the reaction time of the driver of the ego-vehicle in seconds,  $s_f(t) = s(t) + r(t)$  is the speed of the vehicle in front at time  $t$ , and  $b_{mx}$  is the maximum deceleration ability for the ego-vehicle in  $\text{m/s}^2$ . Then, in order to ensure that the ego-vehicle does not accelerate more than what is realistic and does not go faster than the maximum speed desired by the driver, a “desired speed” is computed as:

$$s_{des}(t + \Delta t) = \min\{s_{safe}(t + \Delta t), s(t) + a_{mx} \cdot (1 - \frac{s(t)}{s_{mx}}) \cdot \Delta t, s_{mx}\} \quad (8)$$

where  $a_{mx}$  is the ego-vehicle’s maximum acceleration ability in  $\text{m/s}^2$  and  $s_{mx}$  is the maximum desired speed in  $\text{m/s}$  (i.e. the speed that the ego-vehicle wishes to reach if there is no vehicle in front).

Finally the acceleration at time  $t + \Delta t$  is predicted as:

$$a(t + \Delta t) = \frac{\max\{0, s_{des}(t + \Delta t) - s(t)\}}{\Delta t} \quad (9)$$

#### D. Intelligent Driver Model

The Intelligent Driver Model (IDM), developed by Treiber et al. [14], relies on the computation of a “desired distance” between the ego-vehicle and the vehicle in front:

$$d_{des} = d_{mn} + T \cdot s(t) - \frac{s(t) \cdot r(t)}{2 \cdot \sqrt{a_{mx} \cdot b_{cmf}}} \quad (10)$$

where  $d_{mn}$  is the minimum gap that the driver of the ego-vehicle wishes to keep with the vehicle in front at all times in meters (even when at a complete stop),  $T$  is the desired time headway to the vehicle in front in seconds,  $a_{mx}$  is the ego-vehicle’s maximum acceleration ability in  $\text{m/s}^2$ , and  $b_{cmf}$  is the preferred (comfortable) deceleration in  $\text{m/s}^2$ .

The acceleration at time  $t + \Delta t$  is then computed by comparing the desired gap  $d_{des}$  with the current gap  $d(t)$ , and by adding a term to model the behavior of the ego-vehicle when there is no vehicle in front:

$$a(t + \Delta t) = a_{mx} \left[ 1 - \left( \frac{s(t)}{s_{mx}} \right)^4 - \left( \frac{d_{des}}{d(t)} \right)^2 \right] \quad (11)$$

where  $s_{mx}$  is the maximum desired speed in  $\text{m/s}$  (i.e. the speed that the ego-vehicle wishes to reach if there is no vehicle in front).

#### E. Gaussian Mixture Regression model

Gaussian Mixture Regression (GMR) was introduced two decades ago [15], and is popular in the field of learning by demonstration [16]. It is a non-parametric method to perform nonlinear regression between some input features  $z$  and an output  $y$ . In our case the output is defined as  $y = a(t + \Delta t)$ , and we select the input features as:

$$z = [x(t - \tau_1), x(t - \tau_2), x(t - \tau_3), x(t - \tau_4), x(t - \tau_5)]^T \quad (12)$$

with the delay terms set to cover uniformly the range of typical driver reaction times [4]:  $\tau_1 = 0.4 \text{ s}$ ,  $\tau_2 = 0.8 \text{ s}$ ,  $\tau_3 = 1.2 \text{ s}$ ,  $\tau_4 = 1.6 \text{ s}$ ,  $\tau_5 = 2.0 \text{ s}$ .

During the training phase, real data is used to build a model of the joint distribution  $p(z, y)$  in the form of a Gaussian Mixture Model (GMM) with  $K$  Gaussian functions:

$$p(z, y) = \sum_{k=1}^K \pi_k \cdot \mathcal{N}(z, y | \mu_k, \Sigma_k) \quad (13)$$

where  $\pi_k$ ,  $\mu_k$ , and  $\Sigma_k$  are the prior, mean, and covariance matrix of the  $k^{th}$  Gaussian mixture component. Details about the learning process will be provided in Section IV.

During the testing phase, if we define

$$\mu_k = \begin{bmatrix} \mu_k^z \\ \mu_k^y \end{bmatrix} \text{ and } \Sigma_k = \begin{bmatrix} \Sigma_k^{zz} & \Sigma_k^{zy} \\ \Sigma_k^{yz} & \Sigma_k^{yy} \end{bmatrix},$$

the predicted acceleration is obtained by computing the expectation of the conditional probability  $p(y|z)$ :

$$a(t + \Delta t) = E\{p(y|z)\} = \sum_{k=1}^K \beta_k(z) \cdot (\mu_k^y + \Sigma_k^{yz} (\Sigma_k^{zz})^{-1} (z - \mu_k^z)) \quad (14)$$

where  $\beta_k(z) \in \mathbb{R}_{[0,1]}$  is the probability that the observed input features  $z$  belong to the  $k^{th}$  Gaussian of the GMM.

In this work we implement a combination of GMR and Hidden Markov Model (HMM), as introduced by Calinon et al. [17]. The addition of the HMM allows to encode the temporal characteristics of the signal in the transition probabilities between the different components of the GMM. As a result, the probability that the observed input features  $z$  belong to the  $k^{th}$  Gaussian mixture component is iteratively computed as:

$$\beta_k(z) = \frac{(\sum_{j=1}^K \beta_j(z_{prev}) \cdot a_{jk}) \cdot \mathcal{N}(z | \mu_k^z, \Sigma_k^{zz})}{\sum_{i=1}^K \left[ (\sum_{j=1}^K \beta_j(z_{prev}) \cdot a_{ji}) \cdot \mathcal{N}(z | \mu_i^z, \Sigma_i^{zz}) \right]} \quad (15)$$

where  $a_{ij}$  is the transition probability between the  $i^{th}$  and  $j^{th}$  mixtures, and  $z_{prev}$  the input features observed at the previous timestep.

#### F. Artificial Neural Network model

Artificial Neural Networks (ANNs) have been shown to be a powerful method for time series forecasting [18]. In recent years they have been applied to a number of intelligent transportation problems including driving pattern recognition [19] and car-following modeling [6]. In this paper we implement a radial basis function neural network with four inputs,  $N$  hidden units, and one output. The input features are the same as for the GMR model, i.e.  $z = [x(t - \tau_1), x(t - \tau_2), x(t - \tau_3), x(t - \tau_4), x(t - \tau_5)]^T$ . The processing steps of the ANN are defined as follows:

1) At the input of the  $n^{th}$  hidden unit, the input vector  $z$  is weighted by a vector of input weights  $W_n^{in}$  and a constant “bias” term  $b^{in}$  is added:

$$h_n = W_n^{in} \cdot z + b^{in} \quad (16)$$

2) Each hidden unit contains a nonlinear activation function  $\phi(h_n)$ . In this work the Gaussian function is used, which is a common choice:

$$\phi(h_n) = \exp\left(-\frac{\|h_n - \mu_n\|^2}{2 \cdot \sigma_n^2}\right) \quad (17)$$

where  $\mu_n$  and  $\sigma_n$  are called the center and width of the  $n^{th}$  activation function.

3) Finally, the output  $y = a(t + \Delta t)$  is computed as a linear combination of the outputs of the hidden units:

$$a(t + \Delta t) = \sum_{n=1}^N w_n^{out} \cdot \phi(h_n) + b^{out} \quad (18)$$

where  $w_n^{out}$  is the weight applied to the output of the  $n^{th}$  hidden unit and  $b^{out}$  is a constant bias.

Details about the training process used to learn the parameters of the ANN will be given in Section IV.

#### IV. METHODOLOGY

The six models described in the previous section are evaluated using the same data. The objective is to compare their ability to predict the future speed of the ego-vehicle, for different prediction horizons. In this section we describe the dataset and methodology followed to run the evaluation.

##### A. Dataset description

We collected 60 minutes of lane keeping data on Highway 580 near Berkeley, California (USA). Our Hyundai Grandeur experimental vehicle is equipped with a 77GHz Delphi Electronically Scanning Radar which provides measurements of the distance and relative speed to the vehicle in front. The current speed and acceleration of the ego-vehicle are accessed via the ego-vehicle's CAN bus. In the rest of this paper, the measurements of the current state are denoted by  $\bar{x}(t) = [\bar{d}(t), \bar{r}(t), \bar{s}(t), \bar{a}(t)]^T$ . Measurements are collected every  $\Delta t_c = 0.2$  s, frequently enough to catch the dynamics of highway driving.

Traffic was heavy during the recording of the dataset, which is challenging conditions for speed prediction [5]. The dataset contains both free-flow and car-following situations, and the ranges of variation of the measurements are as follows:  $\bar{d}(t) \in [4.3, 170.0]$ ,  $\bar{r}(t) \in [-10.0, 3.8]$ ,  $\bar{s}(t) \in [4.3, 35.3]$ ,  $\bar{a}(t) \in [-3.0, 1.8]$ .

##### B. Cross-validation

For each of the models described in Section III, we perform 10-fold cross-validation. The dataset described above is partitioned into 10 partitions of equal durations, 9 of which are used for training and the remaining one is used for testing. This process is repeated 10 times, each time with a different partition for testing.

We call  $n_p$  the number of data points (i.e. measurements  $\bar{x}$ ) in each partition. Since each partition is 6-minute long,  $n_p = \frac{6 \times 60}{\Delta t_c} = 1800$ .

1) *Training*: The training process for the parametric approaches consists in finding the parameters of the analytical expression of  $f$  which best fit the training data. This is performed using standard non-linear optimization methods, with constraints on the parameters to ensure that their value is consistent with the concept they represent (e.g. the maximum acceleration parameter  $a_{mx}$  should not be negative and should not exceed a certain value). The boundaries for the different parameters are set based on the article by Bonsall et al. [4] about typical parameter values for driving models.

The training process for the non-parametric approaches is specific to each approach. For the ANN, the parameters  $\mu_n$  and  $\sigma_n$  are determined using gradient descent and the weights and biases are set using the Levenberg-Marquardt algorithm. For the GMR, the number of components in the Gaussian mixture ( $K$ ) is selected using the Bayesian Information Criterion. The parameters of each Gaussian function ( $\pi_k$ ,  $\mu_k$ , and  $\Sigma_k$ ) and the transition probabilities ( $a_{ij}$ ) are computed using the classic Expectation-Maximization algorithm.

2) *Testing different prediction horizons*: The “testing” part of the cross-validation process is conducted for different prediction horizons. As was mentioned in the problem statement (Section II), we are interested in testing both short-term and long-term prediction. We run tests with prediction horizons  $H = \{1, 2, \dots, 10\}$  seconds, so as to cover relevant prediction horizons for a large range of applications from collision avoidance to energy consumption optimization.

For each prediction horizon  $h \in H$ , and for each fold of the cross-validation process, the testing is done on each of the  $n_p$  data points of the testing dataset. For each data point  $\bar{x}(t)$ , the iterative prediction scheme described in Eq. 2-5 is applied  $\frac{h}{\Delta t}$  times to predict the state at time  $t + h$ . The predicted state is denoted by  $\hat{x}(t + h)$ . The update time interval is set to  $\Delta t = 0.2$  s, as a compromise between computation time and validity of the constant acceleration assumption made in the state transition equations<sup>2</sup>.

##### C. Metrics

For each prediction horizon, the performance of each model is evaluated using the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) of the predicted speed with respect to the true speed. These two metrics can be used together to diagnose the variation in the errors in a set of forecasts [20]: the MAE provides a measure of the average magnitude of the errors, and a comparison with the RMSE provides information about the variance in the individual errors in the sample. For one fold of the cross-validation and a prediction horizon  $h$ , the MAE and RMSE are computed as follows :

$$MAE = \frac{1}{n_p} \sum_{t=0}^{(n_p-1) \times \Delta t_c} |\bar{s}(t + h) - \hat{s}(t + h)| \quad (19)$$

<sup>2</sup>The SUMO model was designed to be used with  $\Delta t = 1$  s, but we found that the relative performance of the different models stays the same for any  $\Delta t \in [0.2, 1]$  s used for the SUMO model.

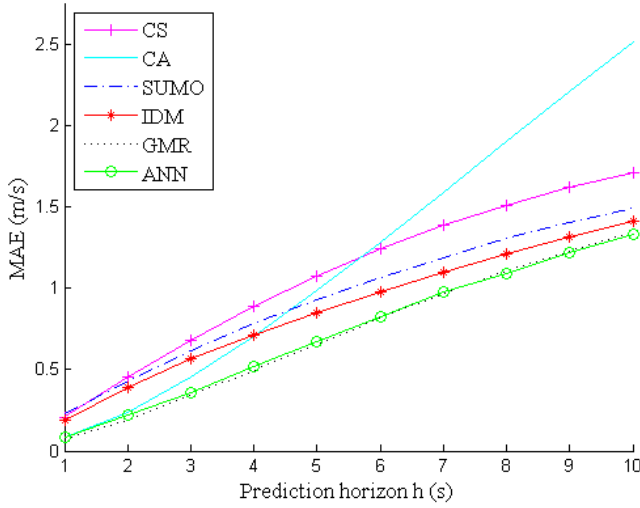


Fig. 1. MAE obtained by each model for different prediction horizons.

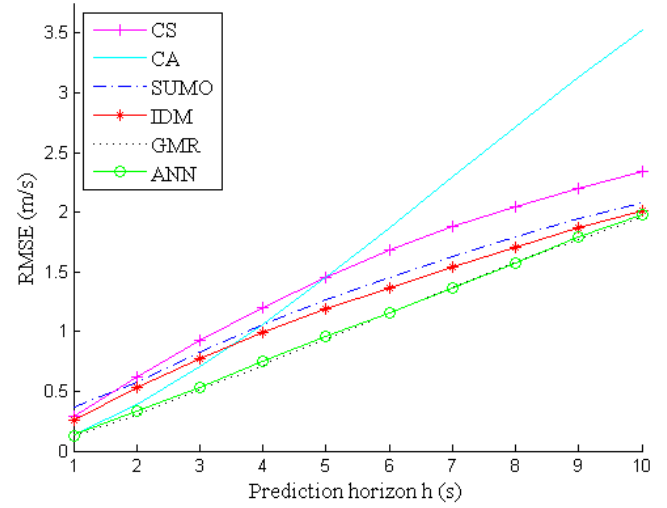


Fig. 2. RMSE obtained by each model for different prediction horizons.

$$RMSE = \sqrt{\frac{1}{n_p} \sum_{t=0}^{(n_p-1) \times \Delta t_c} (\bar{s}(t+h) - \hat{s}(t+h))^2} \quad (20)$$

where  $\bar{s}(t+h)$  is the true speed of the ego-vehicle at time  $t+h$  and  $\hat{s}(t+h)$  is the predicted speed at time  $t+h$  computed as described in Section IV-B.2. For each prediction horizon  $h \in H$ , the performance of a model is given by the average MAE and RMSE over the 10 folds of the cross-validation.

## V. RESULTS

For ease of understanding, in this section the CS and CA models will be referred to as the “simple” parametric models, while the SUMO and IDM models will be named “advanced” parametric models. The GMR and ANN models will still be referred to as the non-parametric models.

### A. Mean Average Error

This section compares the Mean Average Error (MAE) obtained by the different models for different prediction horizons, as displayed in Fig. 1.

1) *Short-term prediction* ( $h \in [1, 4]$ ): For  $h = 1$  s the models can be separated into two groups: the two non-parametric models and the CA model all have a MAE of approximately 0.1 m/s, while the MAE of the other approaches is twice larger. The interval  $h \in [1, 4]$  is characterized by a fast performance decrease for the “simple” parametric models, while the relative distance between the other models does not change much. For  $h = 4$  s the CA model still performs better than the “advanced” parametric models, but only barely. The errors of the CS model are on average 0.1 m/s higher than the “advanced” parametric models.

2) *Long-term prediction* ( $h \in [4, 10]$ ): The CA and CS models are the models with the worst performance for  $h > 4$  s. This leads us to conclude that the “simple” parametric models, while relevant for short-term prediction, are not adequate for prediction horizons larger than 4 s. This result is

known in the literature and is confirmed by our experimental evaluation.

Another noticeable trend which appears with longer prediction horizons is that the curves of the non-parametric models get closer and closer to the curves of the “advanced” parametric models. A likely explanation for this is that SUMO and the IDM were both designed to have stability properties which constrain the long-term behavior of the vehicle, while the non-parametric models were only trained to represent the behavior of the ego-vehicle between two timesteps<sup>3</sup>.

The final comment about the MAE curves concerns the gap between the two “advanced” parametric models. This gap is very small for short-term predictions, but increases after  $h = 5$  s in favor of the IDM model. The SUMO model is indeed known to have some shortcomings in dense traffic situations [11]. More specifically the SUMO model does not constrain the deceleration of the ego-vehicle, which results in excessively hard braking predictions when the traffic density is high. The IDM implements a strategy which limits the braking to a comfortable deceleration  $b_{cmf}$  most of the time, and keeps stronger decelerations for situations which require emergency braking. Our belief is that this “intelligent” braking behavior is what allows the IDM to outperform the SUMO model on our dataset.

### B. Root Mean Square Error

The Root Mean Square Error (RMSE) obtained by the different models for different prediction horizons are displayed in Fig. 2. The trends are the same as for the MAE, the only difference being the rate at which the performance of the CA model decreases. This tells us that the variance in the errors is very similar across all the models except for the CA model which has larger variances.

<sup>3</sup>This is one of the reasons why *direct* or *multiple-output* prediction is generally used instead of *iterative* prediction for long-term prediction with non-parametric methods [21].

### C. Discussion

Classically in the literature, “simple” parametric models are used for short-term speed prediction (e.g. for collision prediction) and “advanced” parametric models are used for long-term speed prediction (e.g. for energy management). Non-parametric models are a popular prediction tool in robotics, but have not been used much in ITS applications. The study presented in this paper confirmed the need to use different models for different applications, depending on the prediction horizon.

For short-term prediction, our results show that CA should be preferred over CS in heavy traffic conditions, since vehicles change their speed more often and more abruptly when traffic is congested. An other point underlined by our study is that using more “advanced” parametric models does not bring any improvement for short-term prediction. For long-term prediction, the IDM handles dense traffic better than SUMO, but overall their performance is very similar. Thanks to their flexible structure, the non-parametric models consistently outperform all the other methods. They are therefore a valid alternative to the classic models, and have the advantage that they perform well for both short-term and long-term prediction horizons.

### VI. CONCLUSIONS AND FUTURE WORK

This paper presented a comparative evaluation of six models for the prediction of the ego-vehicle’s speed on the highway. Both parametric and non-parametric approaches were implemented, and their parameters were learned from real data. For the testing, 10-fold cross validation was run for different prediction horizons. The results showed that simple parametric models perform well for short-term prediction and that more advanced parametric models are necessary for long-term prediction. The non-parametric approaches perform equally or better than the parametric approaches for all the tested prediction horizons. Future work will include testing these models with more data, in particular from more drivers. We will run analyses with the objective to assess the advantages and drawbacks of learning generic models using data from multiple drivers versus learning driver-specific models.

### ACKNOWLEDGMENTS

This work was partially supported by the National Science Foundation under grant No. 1239323 and by the Hyundai Center of Excellence.

### REFERENCES

- [1] S. Panwai and H. Dia, “Comparative evaluation of microscopic car-following behavior,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 3, pp. 314–325, 2005.
- [2] F. Gunawan, “Two-vehicle dynamics of the car-following models on realistic driving condition,” *Journal of Transportation Systems Engineering and Information Technology*, vol. 12, no. 2, pp. 67–75, 2012.
- [3] P. Ranjitkar, T. Nakatsuji, and A. Kawamua, “Car-following models: an experiment based benchmarking,” *Journal of the Eastern Asia Society for Transportation Studies*, vol. 6, pp. 1582–1596, 2005.
- [4] P. Bonsall, R. Liu, and W. Young, “Modelling safety-related driving behaviour - impact of parameter values,” *Transportation Research Part A: Policy and Practice*, vol. 39, no. 5, pp. 425–444, 2005.
- [5] P. Angkititirakul, T. Ryuta, T. Wakita, K. Takeda, C. Miyajima, and T. Suzuki, “Evaluation of driver-behavior models in real-world car-following task,” in *Proc. IEEE International Conference on Vehicular Electronics and Safety*, 2009, pp. 113–118.
- [6] S. Panwai and H. Dia, “Neural agent car-following models,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 1, pp. 60–70, 2007.
- [7] H.-S. Tan and J. Huang, “DGPS-based vehicle-to-vehicle cooperative collision warning: engineering feasibility viewpoints,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 4, pp. 415–428, 2006.
- [8] F. Yan, J. Wang, and K. Huang, “Hybrid electric vehicle model predictive control torque-split strategy incorporating engine transient characteristics,” *IEEE Transactions on Vehicular Technology*, vol. 61, no. 6, pp. 2458–2467, 2012.
- [9] A. Kesting and M. Treiber, “How reaction time, update time, and adaptation time influence the stability of traffic flow,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 23, no. 2, pp. 125–137, 2008.
- [10] K. Vogel, “A comparison of headway and time to collision as safety indicators,” *Accident Analysis & Prevention*, vol. 35, no. 3, pp. 427–433, 2003.
- [11] D. Krajzewicz, “Traffic simulation with SUMO - Simulation of urban mobility,” in *Fundamentals of Traffic Simulation*, ser. International Series in Operations Research & Management Science. Springer New York, 2010, no. 145, pp. 269–293.
- [12] S. Krauss, “Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics,” Ph.D. dissertation, University of Cologne, Germany, 1998.
- [13] P. Gipps, “A behavioural car-following model for computer simulation,” *Transportation Research Part B: Methodological*, vol. 15, no. 2, pp. 105–111, 1981.
- [14] M. Treiber, A. Hennecke, and D. Helbing, “Congested traffic states in empirical observations and microscopic simulations,” *Physical Review E*, vol. 62, no. 2, pp. 1805–1824, 2000.
- [15] Z. Ghahramani and M. Jordan, “Supervised learning from incomplete data via an EM approach,” *Advances in Neural Information Processing Systems*, vol. 6, pp. 120–127, 1994.
- [16] S. Calinon, F. Guenter, and A. Billard, “On learning, representing, and generalizing a task in a humanoid robot,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 37, no. 2, pp. 286–298, 2007.
- [17] S. Calinon, F. D’halluin, E. Sauser, D. Caldwell, and A. Billard, “Learning and reproduction of gestures by imitation,” *IEEE Robotics Automation Magazine*, vol. 17, no. 2, pp. 44–54, 2010.
- [18] G. Zhang, “Time series forecasting using a hybrid ARIMA and neural network model,” *Neurocomputing*, vol. 50, pp. 159–175, 2003.
- [19] R. Langari and J.-S. Won, “Intelligent energy management agent for a parallel hybrid vehicle - part I: system architecture and design of the driving situation identification process,” *IEEE Transactions on Vehicular Technology*, vol. 54, no. 3, pp. 925–934, 2005.
- [20] C. J. Willmott, S. G. Ackleson, R. E. Davis, J. J. Feddema, K. M. Klink, D. R. Legates, J. O’Donnell, and C. M. Rowe, “Statistics for the evaluation and comparison of models,” *Journal of Geophysical Research: Oceans*, vol. 90, no. C5, pp. 8995–9005, 1985.
- [21] S. Ben Taieb, A. Sorjamaa, and G. Bontempi, “Multiple-output modeling for multi-step-ahead time series forecasting,” *Neurocomputing*, vol. 73, no. 10–12, pp. 1950–1957, 2010.