Available online at www.sciencedirect.com
ScienceDirect
IFAC-PapersOnLine 48-15 (2015) 257–262

Prediction of Vehicle Velocity for Model Predictive Control

Amir Rezaei * **Jeffrey B. Burl** **

* *Electrical and Computer Engineering, Michigan Technological University, Houghton, MI 49931 USA (e-mail: arezaei@mtu.edu).*
** *Electrical and Computer Engineering, Michigan Technological University, Houghton, MI 49931 USA (e-mail: burl@mtu.edu)*

Abstract: In model predictive control, knowledge about the future trajectories of the set points or disturbances is used to optimize the overall system performance, Camacho and Bordons (2007). For hybrid electric vehicles, by predicting the future Driver's Desired Velocity (DDV), fuel economy, or emissions can be improved, Debert et al. (2010). For predicting DDV, different approaches have been suggested, for example, artificial neural networks, Fotouhi et al. (2011), statistical methods, or methods based on GPS and Geographical Information Systems(GIS), Keulen et al. (2009). In this work, some of these approaches are introduced and autoregressive methods with GPS/GIS information are evaluated.

Keywords: Model Predictive Control, Velocity Prediction, Time Series.

1. INTRODUCTION

The idea of forecasting the future behavior of statistical time series has a wide range of applications in economics, sociology, climate analysis, communications, control systems, etc. In control systems, time series forecasting is specifically applicable for improving the performance of instantaneous optimal controllers that try to minimize a cost function by modifying the controllable inputs of the plant. The idea is that instead of optimizing the cost function at each moment (instantaneous optimal control), the optimization be done over a time horizon (Model Predictive Control, MPC). So, the cost function trajectory is closer to the globally optimal trajectory. To achieve this goal, it is necessary to predict the future trajectories of all of the reference signals $\boldsymbol{r}(t)$ of the system. When the plant is a vehicle, DDV is the only reference signal that needs to be predicted.

DDV time series by nature is a non-stationary random process and most of the methods for modeling and forecasting non-stationary time series are applicable for DDV prediction. But, the DDV trajectory is highly affected by some predictable environmental factors like traffic signs, road condition, etc. The possibility of predicting some of the factors (events) that can affect DDV, is one of the advantages of forecasting DDV over other time-series forecasts. For DDV prediction, different approaches have been suggested: off-line modeling on recorded data, artificial neural networks, models based on GPS/GIS information, and statistical methods.

This work introduces some of the approaches that are suggested for predicting DDV in section 2. Then in section 3, the benefit of using statistical methods with GPS/GIS information over other approaches, is discussed. Finally, in section 4, a combination of time series methods with *a priori* knowledge of deterministic environmental factors is presented for DDV forecasting.

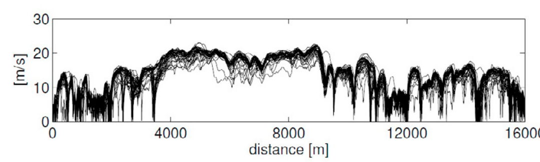

Fig. 1. Velocity data set on a constant path Johannesson et al. (2005)

2. OVERVIEW OF SUGGESTED METHODS FOR DDV PREDICTION

2.1 Off-line modeling using recorded data

Off-line modeling, discussed by Lin et al. (2004), Johannesson et al. (2005), is applicable for repeated paths like home to work drive cycles or the path of a bus. The path is traveled several times to create a data set for the vehicle speed v, acceleration a, position d, and the road gradient g, along that path. Then a model is developed from the data set and used for DDV prediction under real conditions on that path. This idea can also be extended on a large scale by creating a huge database for all streets in a city or even a country.

For example in Johannesson et al. (2005) and Lin et al. (2004), a Markov chain model is developed on the data set (Fig. 1): First a set of discrete numbers (grids) is defined for each recorded data: $v = \{v_1, ..., v_p\}, a = \{a_1, ..., a_q\}, d = \{d_1, ..., d_r\}, s = \{g_1, ..., g_p\}$. Then each recorded data point (v, a, d, g) is quantized on the defined grid to achieve a set of states $x = \{v, a, d, g\}$. Now, since the sequence of recording data is known, it is possible to define a state probability transition among the set of states $p_{i,j} = f(x_{k+1}|x_k)$ where $p_{i,j}$ represents the probability of trqansistion from state i at time k to state j at time $k+1$. The same logic can be used for prediction times of more than 1 second.

In Yokoi et al. (2004), instead of samples, features of driving pattern are extracted and an unsupervised learning

10.1016/j.ifacol.2015.10.037

method (clustering) is used as a model to represent the data.

An artificial neural network (ANN) is another approach for mapping a nonlinear function onto a set of data. In Fotouhi et al. (2011), multi-layer perceptron neural networks are used for up to 10 seconds forecasting during city driving. They trained 10 different multi-layer perceptron ANNs for each second ahead.

2.2 Recurrent artificial neural networks

ANN can be described by a topology and a set of synaptic weights $\{w_i\}$ and during the training phase, the weights are updated to achieve the least error between the ANN outputs $\hat{y}$ and desired outputs y. So, an ANN can can be presented as a dynamic system:

$$\boldsymbol{w}[k+1] = f(\boldsymbol{w}[k], \boldsymbol{e}[k]) \ , \hat{\boldsymbol{y}}[k] = g(\boldsymbol{w}[k], \boldsymbol{u}[k]) \quad (1)$$

where $\boldsymbol{u}$ is the vector of inputs and $\boldsymbol{e} = \boldsymbol{y} - \hat{\boldsymbol{y}}$ is the error vector. In the previous section, an ANN was introduced as an approach for off-line modeling of the recorded data, Fotouhi et al. (2011). However, when an ANN is trained with a pre-recorded database, it may still fail under real conditions due to the infinite possible realizations of the DDV as a non-stationary stochastic process. This happens because after the training phase, the set of $\boldsymbol{w}$ or the ANN state remains constant. To resolve this issue, an adaptive ANN can be used instead and Kalman filter equations can be employed to train the network. The idea is to assume the evolution of the ANN weights is a measurement process:

$$\begin{cases} \boldsymbol{w}[k+1] = \boldsymbol{w}[k] + \boldsymbol{n_1}[k] \\ \hat{\boldsymbol{y}}[k] = g(\boldsymbol{w}[k], \boldsymbol{u}[k]) + \boldsymbol{n_2}[k] \end{cases} \quad (2)$$

where n_1 is the plant noise and and n_2 is the measurement noise and both are white noises: $E\{n_i[k]n_i[k+m]\} = \Sigma_i \delta[m]$. Hence, at each moment when new data $\hat{\boldsymbol{y}}$ is measured, the ANN can be trained again. The process of training the ANN can be treated as optimum estimation of the system states $\boldsymbol{w}$ by minimizing the state/measurement errors. As a result, the extended Kalman filter is a suitable algorithm for real time training of the ANN. The above method is used in Alanis et al. (2009) with a recurrent high order ANN for wind speed prediction. However, at this moment no work was found for DDV forecasting with this method.

2.3 Models based on GPS/GIS information

The sequence of events like traffic signs, traffic flow, road curve/grade, weather condition, etc., will affect the speed limit on a route and consequently affect the DDV trajectory as is shown in Fig. 2. With recent advances in navigation systems, *a priori* knowledge of upcoming events as a deterministic time series S_t can be acquired. This prior information about S_t can be used for estimating the DDV time series.

Almost all of the approaches based on prior knowledge of S_t, divide the upcoming road into segments with the same speed limits. Then, the vehicle speed during that segment is estimated with the initial condition that each segment final speed is equal to the next segment initial speed or vice versa. Different methods have been suggested to estimate DDV on one segment. For instance in Keulen et al. (2009), acceleration or deceleration phases are predicted based on

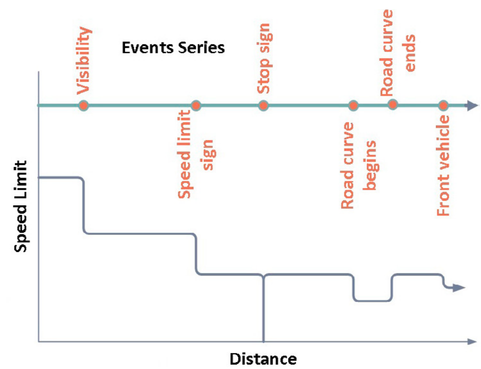

Fig. 2. Speed limits affected by a sequence of events S_t

minimizing fuel consumption. In Mller et al. (2004), first a critical acceleration a_{crit} is calculated at the end of each segment to reach the speed limit of the next segment. Then, a_{crit} has been used as the basis for predicting driver's desired acceleration under braking, deceleration and acceleration situations.

2.4 Statistical Methods

Statistical time series forecasting may be divided into time and frequency domain analysis, Ao (2010). In time domain analysis, autoregressive (AR), moving average (MA), autoregressive moving average ($ARMA$) are some of the tools that can model stationary time series. For non-stationary time series, models like autoregressive integrated moving average ($ARIMA$)are suggested.

The $ARMA(p, q)$ model for a zero mean time series x_t is:

$$ARMA(p,q): \ x_t = \sum_{i=1}^{p} \varphi_i (x_{t-i} - \mu) + \sum_{i=1}^{q} \theta_i \omega_{t-i} + \omega_t \quad (3)$$

where p and q are the orders of the AR and MA components, respectively; φ_i and θ_i are the AR and MA parameters, respectively, and ω_t is zero mean white noise. $ARMA(p, 0)$ is equivalent to a simple autoregressive model of order p: $AR(p)$. Similarly, $ARMA(0, q)$ is equivalent to a simple moving average model of order q: $MA(q)$.

The $ARMA$ model produces a stationary time series. As a result when modeling a non-stationary random process with $ARMA$, first the time series has to be transformed to a stationary time series. Different techniques like detrending, taking difference or nonlinear transformations are introduced for this purpose. The $ARIMA(p, d, q)$ model is an extended version of $ARMA(p, q)$ that applies a difference operation d times to a non-stationary time series to yield a stationary time series.

For estimating a time series with an $ARIMA(p, d, q)$ process, first the model parameters p, d, q have to be identified. The basic method for model identification is the evaluation of the sample auto-correlation function (ACF) and partial auto-correlation function (PACF) of the recorded data.

For prediction based on $AR(p)$, if the joint probability of the target parameter x_{t+h} and the observations $\{x_t, x_{t-1}, x_{t-N+1}\}$ is unknown, then the best linear estimation is achieved by applying the orthogonality principle between the prediction error and the observed data (also known as Yule-Walker estimator; See Box et al. (1994)). The predictor is: