HAL
open science

LPV control for autonomous vehicles using a machine learning-based tire pressure estimation

Dániel Fényes, Tamás Hegedűs, Balázs Németh, Péter Gáspár, Damien Koenig, Olivier Sename

To cite this version:

Dániel Fényes, Tamás Hegedűs, Balázs Németh, Péter Gáspár, Damien Koenig, et al.. LPV control for autonomous vehicles using a machine learning-based tire pressure estimation. MED 2020 - 28th Mediterranean Conference on Control and Automation, Sep 2020, Saint-Raphaël, France. 10.1109/MED48518.2020.9183106 . hal-02940732

HAL Id: hal-02940732

https://hal.univ-grenoble-alpes.fr/hal-02940732v1

Submitted on 16 Sep 2020

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.

LPV control for autonomous vehicles using a machine learning-based tire pressure estimation

Dániel Fényes, Tamás Hegedűs, Balázs Németh, Péter Gáspár, Damien Koenig and Olivier Sename

***Abstract*— The paper presents a data-driven method for tire pressure estimation and an LPV-based control design for autonomous vehicles. The motivation of the research is that the pressures of the tires have high impacts on the lateral dynamics of the vehicle, because the loss of tire pressure may result in degradation in the lateral vehicle motion. First, a machine learning-based estimation algorithm, which uses only signals of on-board sensors, is proposed. Second, an LPV-based lateral control design is proposed, which uses the estimated tire pressure as a scheduling variable. The control is able to handle situations, in which the tire pressure decreases. The efficiency and the operation of the control system is illustrated through a comprehensive simulation example using the high-fidelity simulation software CarMaker.**

I. Introduction and motivation

Throughout the history of the automotive industry, the main goals are to develop new devices, techniques and methods. For example vehicles use sensor fusion-based approaches to get rid of expensive sensors such as high-accuracy GPS or pressure sensors. The main idea behind these methods is to use low-cost sensors, with which vehicles are already equipped, such as accelerometers, gyroscopes. These sensors provide explicit signals about the vehicle, and additionally, hidden implicit information on the unmeasured parameters of the vehicle. Data-driven methods are able to improve the estimation of vehicle parameters, which can reveal hidden information using deep-learning and machine learning algorithms.

In the literature numerous studies are found, which are related to sensor fusion tasks. A novel method to estimate the side-slip angle of the vehicle using only on-board sensors was presented by [1] . The stability analysis of the vehicle by using the onboard measurements was presented by [2]. A wavelet-based approach for the tire pressure estimation was proposed by in [3]. An indirect estimation method, which exploits the high correlation between the stiffness of the tire and the pressure, was proposed by [4]. An approach, in which the lateral dynamics measurements are used to detect the loss of pressure, was proposed by [5]. The low pressure in the tire may highly influence the lateral stability and performances of the vehicle, because the lateral tire force depends on the pressure. Several control methods have been developed to avoid this problem. An $\mathcal{H}_\infty$-based solution, which can ensure the stability of the vehicle at low tire pressure, was proposed by [6]. A flatness-based MPC (Model Predictive Control) approach was proposed for handling this task, see [7].

In this paper the estimation and control design tasks are simultaneously addressed. The main contribution of the paper is a novel estimation method for tire pressure. The method is based on a machine learning-based algorithm, more precisely, the Pace regression method. In the control design an LPV (Linear Parameter Varying) based method is applied, in which the estimation of the tire pressure is used as a scheduling parameter. Combining these functions, a more reliable control strategy is achieved, which is able to guarantee stable and safe lateral motion of the vehicle even at low tire pressures.

The paper is structured as follows: Section II presents the steps of the data acquisition and the data analysis. The applied machine learning algorithm is also presented in this section. The control design steps are detailed in Section III. Furthermore, a simulation example is presented in Section IV shows simulation results. Finally, the paper is summarized in Section V.

II. Method of vehicle data collection and analysis

The entire structure of the control system is depicted in Figure 1. The algorithm is divided into three layers which are the followings: *Simulation environment*, *Tire pressure estimation* and *Control system*. The *Simulation environment* serves to validate the algorithm, in which *CarMaker* simulation software is used. The *Control System* consists of the main steps of the control design, which can be divided into two sub-layers. The upper sub-layer generates the reference trajectory, and the lower one is responsible for the control of the vehicle. In this section the *Tire pressure estimation* layer is presented in detail.

The goal of the *Tire pressure estimation* layer is to estimate the pressure of the tires using low-cost sensors. In this

D. Fényes, T. Hegedűs, B. Németh, P. Gáspár are with Systems and Control Laboratory, Institute for Computer Science and Control, Kende u. 13-17, H-1111 Budapest, Hungary. E-mail: [daniel.fenyes;tamas.hegedus;balazs.nemeth;peter.gaspar]@sztaki.hu

D. Koenig, O. Sename are with GIPSA-lab, INPG, Universite Grenoble Alpes, 11 Rue des Mathematiques, 38402 Grenoble, France.E-mail: [damien.koenig;olivier.sename]@gipsa-lab.grenoble-inp.fr

The research was supported by the National Research, Development and Innovation Office through the project 'Integration of velocity and suspension control to enhance automated driving comfort in road vehicles' (NKFIH 2018-2.1.13-TET-FR). The research reported in this paper was partially supported by the Higher Education Excellence Program of the Ministry of Human Capacities in the frame of Artificial Intelligence research area of Budapest University of Technology and Economics (BME FIKPMI/FM).

The work of Balázs Németh was partially supported by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences and the ÚNKP-19-4 New National Excellence Program of the Ministry for Innovation and Technology.

The work of Dániel Fényes and Tamás Hegedűs was partially supported by the ÚNKP-19-3 New National Excellence Program of the Ministry for Innovation and Technology.

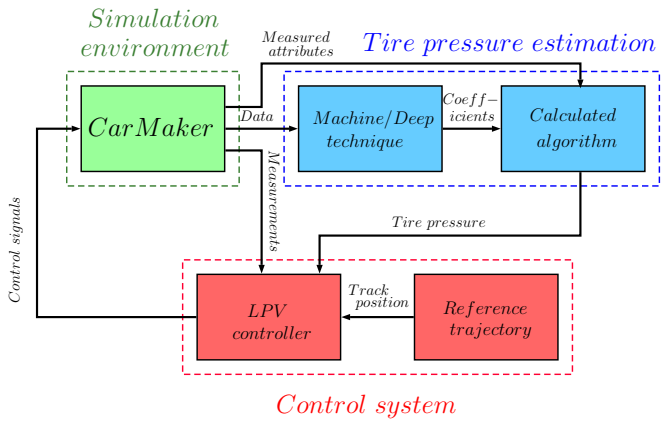


Fig. 1. Structure of the control system

estimation algorithm only the commonly used sensor signals are used (e.g. accelerations, velocities, steering angle).

A. Acquisition of data from simulations

The first step is the acquisition of the appropriate data, with which dataset for the training and validation of the machine learning-based algorithms can be provided. For this purpose several simulations have been performed in the simulation environment, which is the high-fidelity vehicle software CarMaker. During these simulations, two parameters of the car have been modified, such as tire pressure and the longitudinal velocity.

The longitudinal velocity varies in the interval $11m/s...15m/s$, whilst, the pressure of the front left tire changes between $1.0 - 2.5bar$ with the step size $\Delta p = 0.1$ bar. The velocity is set by a built-in PID-based cruise control. During the simulations several parameters are measured and saved such as yaw-rate, accelerations and velocities in various directions, steering angle, tire forces etc. The sampling time is set to $T_s = 0.01s$. In this way, more than one million distinct instances are collected.

B. Brief overview of Pace Regression

In the next step the collected data is evaluated by using the Pace Regression algorithm [8], which has been implemented in various data-mining software, e.g. Weka. Basically, the algorithm Pace regression is a standard linear regression method. This regression uses a clustering algorithm to determine the best subset of the attributes (variables). It works in the following way:

- X represents the design matrix, and its size is $n \times k$.
- It is assumed that the dataset consists of n independent instances.
- The dataset has k input variables and one output. This dataset is written in a matrix X. ζ^* is the parameter vector of the best model, then the output y is given by the following linear equation.

$$y = X\zeta^* + \epsilon \tag{1}$$

The algorithm tries to find the best parameter vector by minimizing the deviation between the calculated output $\hat{y}$ and the measured output y. During this minimization, the algorithm creates a lot of subsets to find the best fitting solution for this task. Further details of Pace Regression is found in [9].

C. Evaluation of the regression models

In the following the results of the machine learning based estimation is presented though illustrations. For example, when all of the collected attributes in the regression are used, the machine learning algorithm provides accurate estimation. Figure 2 figure shows the results of the estimation for three different cases, when the pressure of the tire is set to 1.3 bar, 1.7 bar and 2.1 bar, respectively, and the velocity is set to 14 m/s. The calculated pressures values are rounded to eliminate fluctuation in the estimation.

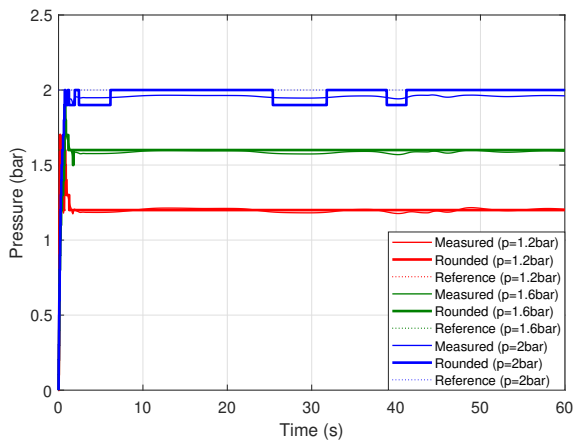


Fig. 2. Estimation of the pressure constant velocity

Although the previous estimation seems to be accurate enough, the velocity has been set to constant, which is unrealistic in various traffic scenarios. Thus, the velocity of the vehicle varies by using the CarMaker Driver, which is the in-built model of the simulator. The estimated pressure is illustrated in Figure 3.

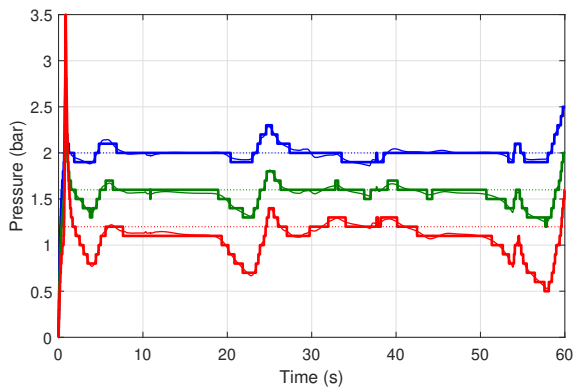


Fig. 3. Estimation of the pressure CarMaker Driver

As the velocity varies during the simulation, the estimation becomes less accurate. Therefore, a new dataset is generated,

in which the reference velocity of the vehicle varies randomly during the simulation. However, in this case the Pace Regression provides worse results using only the actual values of the attributes. Therefore, the past values of the variables must be also used. In order to determine the most influential past values, a spectral analysis is performed, which is illustrated in Figure 4. It is shown that the information content of the signal is below $6Hz$, above this value only the noises appear. Consequently, the time interval between two consecutive points is set to $T = 1/6Hz = 0.15\ s$. In this manner, the most significant values are selected and the number of the past values are reduced.

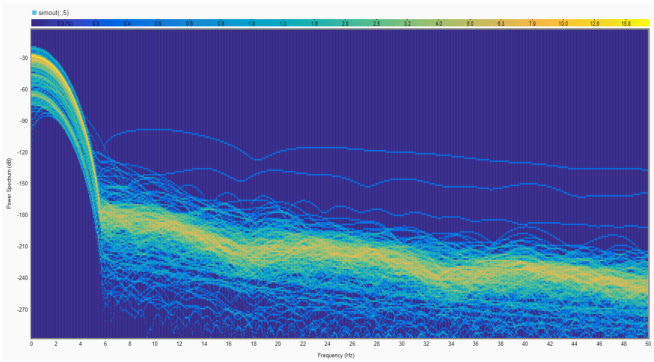


Fig. 4. Result of the spectral analysis

Moreover, in practice some of the attributes cannot be measured directly, e.g. the forces on the tires or the side slip angles. Therefore, only the signals, which are available from the on-board system (velocity, steering angle, wheel speeds, yaw rate and accelerations) are selected in the generation of the regression model. Using the selected data, a new estimation model is built up. The following table shows correlation coefficients of the generated models and the mean errors.

All data	Past data	Const. velocity	Accuracy	Av. error
✓	✗	✓	97.9	0.0063
✗	✗	✓	69.19	0.072
✓	✓	✓	96.38	0.0194
✗	✓	✓	90.1	0.047
✓	✓	✗	19.5	0.129
✗	✓	✗	9.8	0.29
✗	✗	✗	22.3	0.91

TABLE I

TABLE OF THE ACCURACY

The result of the above simulations has shown that when all of the attributes is applied, the accuracy of the estimation is the highest. When only the measurable signals are used, the accuracy reduces. However, using also the past values of attributes the rate of the correctly classified pressure values increases. In both cases, the estimation was inaccurate when the velocity varies during the simulation. Thus, new dataset is generated, in which the reference value of the velocity varies randomly. The results obtained by using the new dataset are shown in the following table, but in this case only the results of the relevant algorithms are shown.

All data	Past data	Const. velocity	Accuracy	Av. error
✗	✗	✗	26.7	0.101
✗	✓	✗	57.4	0.14

TABLE II

TABLE OF THE ACCURACY USING THE NEW DATASET

As table shows, in the second case, the accuracy is high close to 60% with a reasonable mean error of 0.14. This means that the algorithm is able to estimate the tire pressure well.

III. LPV-BASED CONTROL USING THE TIRE PRESSURES

In this section an LPV-based lateral control design is presented for automated vehicles. In the control design, the pressure of the front tire is used as a scheduling parameter. This parameter is estimated by using the machine learning technique. The model another scheduling variable, which is the longitudinal velocity. The control inputs of the system are the steering angle (δ) and the differential torque (M_d). In this section the control-oriented vehicle model, the steering model and the LPV design are proposed.

A. Modeling of the vehicle dynamics

The lateral dynamics of the vehicle is modeled by the single-track bicycle model. The idea behind this model is to replace the front and rear wheels by a single track, which are placed on the symmetrical axis of the vehicle. The model consists of three basic equations [10]:

$$I\ddot{\psi} = F_{f,y}(\alpha_1)l_1 - F_{r,y}(\alpha_2)l_2 + M_d + \Delta F(p_1)l_1 \quad (2)$$

$$mv_x(\dot{\psi} + \dot{\beta}) = F_{f,y}(\alpha_1) + F_{r,y}(\alpha_2) + \Delta F(p_1) \quad (3)$$

$$\dot{v_y} = v_x(\dot{\psi} + \dot{\beta}) \quad (4)$$

where m is the mass of the car, v_x is the longitudinal velocity, I denotes the yaw-inertia, $\dot{\psi}$ is the yaw-rate, β represents the side-slip angle, l_i are geometric parameters, $F_{i,j}$ are the lateral forces generated by the tires, ΔF is the additional force generated by the pressure loss, α_i denote the side-slip of the front and rear wheels.

In the control-oriented model, the characteristics of the lateral forces is linearized and it is rewritten as $F_{i,y} = C_i\alpha_i$, where C_i denotes the cornering stiffness. Although the pressure p_1 does not appear in the equations, this variable highly correlates to the cornering stiffness (C) which is determined as [11]:

$$C = \frac{F_y}{\alpha} \quad (5)$$

The effect the pressure loss can also be described as a linear function since it similarly influences the lateral force as the cornering stiffness, see Figure 5 .

The lateral model is transformed into a state-space representation:

$$\dot{x}_v = A_v(v_x, p_1)x_v + B_v(v_x, p_1)u_v \quad (6)$$

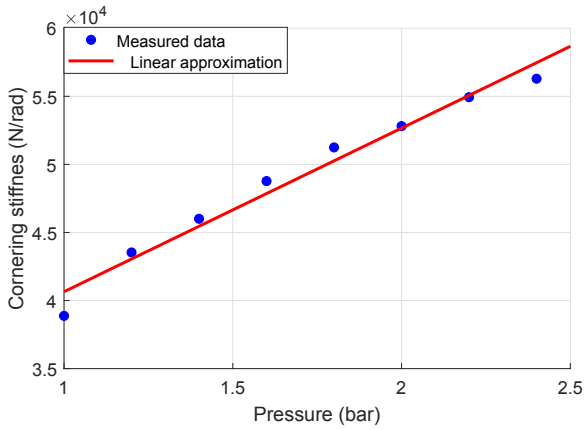


Fig. 5. Relationship between pressure and cornering stiffness

The state vector is $x_v = \begin{bmatrix}\dot{\psi} & \beta & v_y & y\end{bmatrix}^T$. The control input vector is $u_v = \begin{bmatrix}\delta & M_d\end{bmatrix}$. The scheduling parameters are the longitudinal velocity v_x and the pressure of the front tire p_1.

B. Modeling of the steering system

In order to increase the performances of the lateral control system, the bicycle model is augmented with dynamics of the steering mechanism. The dynamics of the steering system is approximated by a second-order system, whose general form is:

$$G_s(s) = \frac{b_2 s^2 + b_1 s + b_0}{s^2 + a_1 s + a_0} \tag{7}$$

Parameters b_i and a_i of the transfer function $G_s(s)$ are dformed in ARX (autoregressive with exogenous signal) structure, see [12].

$$\begin{aligned} y(t) + a_1 y(t-1) + ... + a_{n_a} y(t-n_a) \\ = b_1 u(t-1) + ... + b_{n_b} u(t-n_b) + e(t), \end{aligned} \tag{8}$$

where u is the input, y is the output of the system and e is a model noise. In this case y is the angle of the wheels on the front axle, while u is the angle of the steering wheel. In the steering system the following paremeters are to be estimated:

$$\sigma = [a_0 \quad a_1 \quad ... \quad a_{t-n_a} \quad b_0 \quad b_1 \quad ... \quad b_{t-n_b}]^T \tag{9}$$

Using (8) the following two equations are written:

$$A(q) = 1 + a_1 q^{-1} + ... + a_{n_a} q^{-n_a} \tag{10}$$

$$B(q) = b_1 q^{-1} + ... + b_{n_b} q^{-n_b} \tag{11}$$

where q^{-1} denotes the shift operator. Using (10) the transfer function of the system is formed as:

$$G(q, \sigma) = \frac{B(q)}{A(q)} \tag{12}$$

Since the method results in a discrete-time linear model, the transfer function must be transformed into a continuous system. The sampling time is set to $T_s = 0.01s$. Finally, the following continuous-time model is formed as follows:

$$\dot{x}_s = A_s x_s + B_s u_s, \tag{13}$$

$$y_s = C_s x_s, \tag{14}$$

where A_s is the system matrix, B_s is the input matrix and C_s is the output matrix.

C. Design of the lateral control of the vehicle

The two state-space representations (6) and (13) are combined into an extended form:

$$\dot{x_e} = A_e(v_x, p_1) x_e + B_e(v_x, p_1) u_e \tag{15}$$

where u_e=$[u_s \ M_d]$ and $x_e = [x_s \ x_v]^T$, while the matrices are:

$$A_e(v_x, p_1) = \left[\begin{array}{c|c} A_s & 0_{2x4} \\ \hline B_{v,1}(v_x, p_1) C_s^T & A_v(v_x, p_1) \end{array}\right], \tag{16a}$$

$$B_e(v_x, p_1) = \left[\begin{array}{c|c} B_s & 0_{2x1} \\ \hline 0_{4x1} & B_{v,2}(v_x, p_1) \end{array}\right], \tag{16b}$$

in which $B_{v,i}(v_x, p_1)$ denotes the ith column of $B_v(v_x, p_1)$.

The purpose of the control system is to ensure smooth tracking of the predefined road, while the control signals have as small as possible. The goals are formed by the following three performances.

- *Minimization of the lateral error* The control algorithm must minimize the deviation between the lateral position of the vehicle y and the current lateral position of the road y_{ref}:

$$z_1 = y_{ref} - y, \qquad |z_1| \to min, \tag{17}$$

  where y_{ref} is considered to be given.
- *Minimization of the steering angle* An inherent requirement is to minimize the control signals in order to reduce the energy consumption.

$$z_2 = \delta, \qquad |z_2| \to min. \tag{18}$$

- *Minimization of the differential drive* Another requirementd is to minimize the intervention of the differential driving.

$$z_3 = M_d, \qquad |z_3| \to min. \tag{19}$$

The performances are summarized in the following vector $z = \begin{bmatrix}z_1 & z_2 & z_3\end{bmatrix}^T$, which leads to the performance equation

$$z = C_1 x_e + D_{11} r + D_{12} u_e, \tag{20}$$

where C_1, D_{11}, D_{12} are matrices and r contains the signal y_{ref}.

In the LPV control design the extended state-space representation (15) is used. In the design all of the measured outputs and the input signals have to be scaled to construct a suitable controller. In general, transfer functions are used to weight the signals. The weighting functions and the augmented plant are illustrated in Figure 6.

The weighting function $W_{ref,1}$ is used to scale the reference signal y_{ref}. $W_{z,1}$, $W_{z,2}$ and $W_{z,3}$ are used to guarantee the predefined performances. Moreover, the functions $W_{z,2}$ and $W_{z,3}$ are used to ensure the trade-off between the interventions. They are scaled, such a manner, to use rather

the differential drive than the steering, when the pressure of the tire is low:

$$W_{z,2} = \frac{p_{max}^2}{p_{est}} G_1, \tag{21}$$

$$W_{z,3} = \frac{p_{est}^6}{p_{max}} G_2, \tag{22}$$

where G_1 and G_2 are first order transfer functions. The reason of this scaling is that the reachable lateral force decreases together with tire pressure. At low tire pressure, the steering intervention is compensated with the differential drive. The rest of the weighting functions $W_{w,1}$, $W_{w,2}$ and $W_{w,3}$ are attenuate the noises of the measurements.

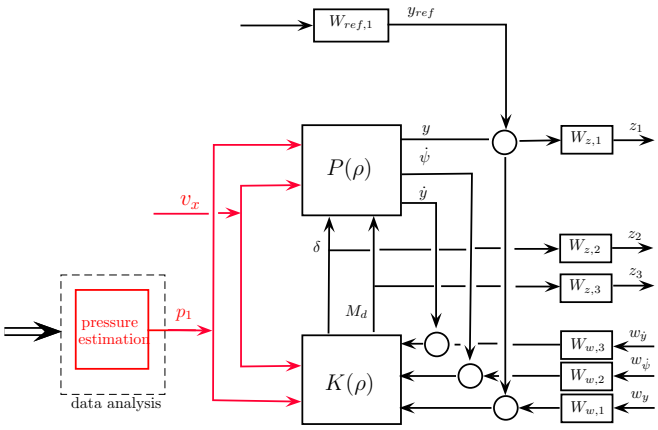


Fig. 6. Structure of LPV controller

The LPV design task leads to a quadratic function that must be minimized by choosing the appropriate $K(v_x, p_1)$ controller, which guarantees that the closed-loop system is quadratically stable. Further requirement is to ensure that the induced norm $\mathcal{L}_2$ between the performances and the disturbances is less than a given value γ.

$$\inf_{K(v_x,p_1)} \sup_{v_x,p_1 \in F_\rho} \sup_{\|w\|_2 \neq 0, w \in \mathcal{L}_2} \frac{\|z\|_2}{\|w\|_2}, \tag{23}$$

where F_ρ bounds the scheduling variables. The computed controller $K(v_x, p_1)$ is formed as

$$\dot{x}_K = A_K(v_x, p_1) x_K + B_K(v_x, p_1) y_K, \tag{24a}$$

$$u = C_K(v_x, p_1) x_K + D_K(v_x, p_1) y_K, \tag{24b}$$

where $A_K(v_x, p_1), B_K(v_x, p_1)$ and $C_K(v_x, p_1), D_K(v_x, p_1)$ are scheduling variable dependent matrices.

IV. Simulation Results

Finally, the efficiency and the operation of the control system is illustrated through a comprehensive simulation example. In the simulations, the vehicle is driven along the predefined track. In the first simulation, the vehicle is controlled by a nominal controller without using any information about the tire pressure. In the second case, the passenger car is driven by the proposed control system including the tire pressure estimation algorithm. In both cases, the pressure of the front left tire is set to $1.2bar$.

Figure 7 shows the paths of the vehicle for both simulations. It can be seen that the in-built driver model is not able to guarantee the tracking of the path at a sharp bend, because in the control strategy the effect of the pressure loss is not built into the model. Thus, the vehicle leaves the road, see Figure 8 .

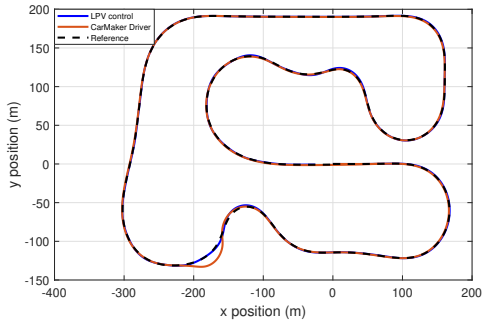


Fig. 7. Positions of the vehicles during the simulations

The proposed control system is able the ensure the trajectory tracking of the vehicle due to the consideration of the pressure loss.

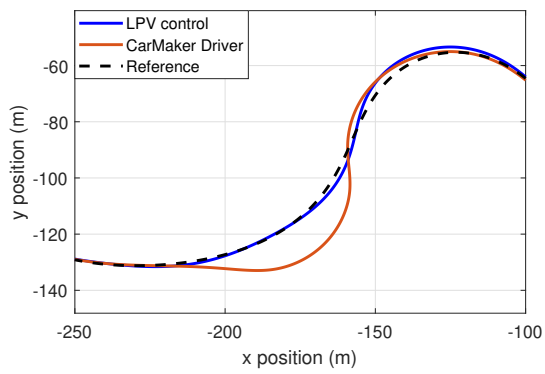


Fig. 8. The bend where the driver leaves the road

Figure 9 shows the results of the estimations using different models. The blue line represents the estimation, when only the measurable data is used with past values at constant velocity. The red one illustrates the result of the scenario, when the velocity varies. The green one is the scenario, in which all signals are used during the estimation. The estimation model provides bad results between $40sec$ and $50sec$ which is caused by the high side-slip angle and yaw rate, such as sharp bends as shown in Figure 8(b).

The LPV controller computes the steering angle and the differential drive using the estimated tire pressure and longitudinal velocity. The computed interventions are illustrated in Figure 10. As the first figure shows the LPV-based controller provides lower steering angle at the sharp bend by which the stability of the vehicle is maintained. The lowest steering

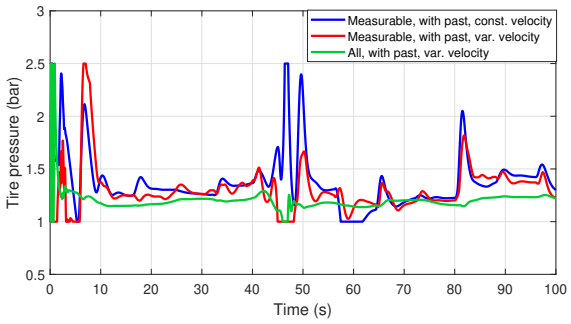
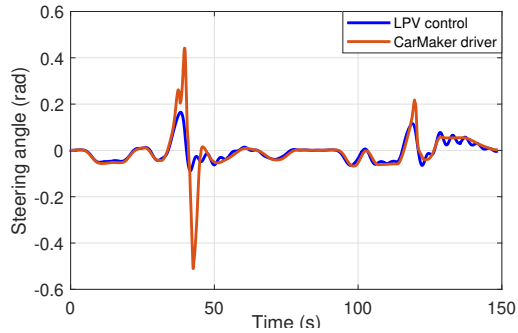
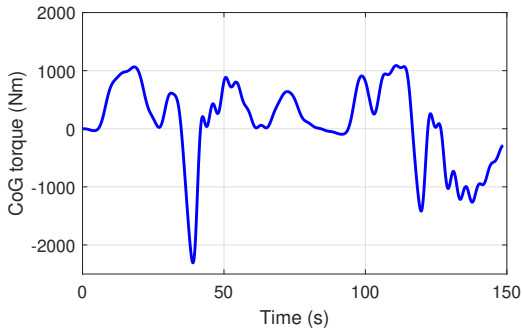


Fig. 9. Result of the pace regression based estimation

angle is compensated by the differential torque shown in Figure 10(b).



(a) Steering angle



(b) Desired torque actuation

Fig. 10. Control inputs of the system

The last figure illustrates the longitudinal velocity of the vehicle during the simulation. As the figures shows, the proposed LPV control system is able to guarantee the trajectory tracking of the vehicle at different velocities.

V. Conclusions

This paper has proposed a new method, in which the estimation of the tire pressures has been performed by a machine learning technique. The main advantage of the proposed algorithm is that only the commonly used sensor signals are used for the estimation. The estimated tire pressures is used as a scheduling parameter of the control system. The control system is based on the LPV method, which uses the longitudinal velocity as a scheduling parameter beside the estimated tire pressure. Finally, a comprehensive simulation, which is made in CarMaker vehicle simulation software, illustrates the effectiveness of the proposed algorithm.

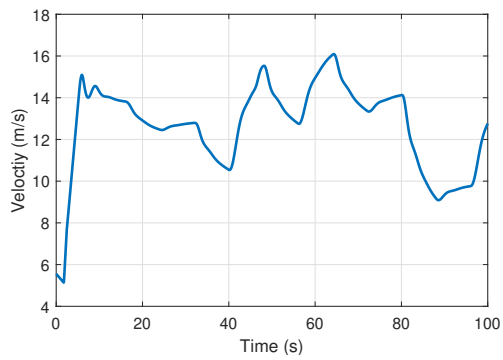


Fig. 11. Velocity of the vehicle

References

[1] D. Fenyes, B. Nemeth, and P. Gaspar, "Analysis of autonomous vehicle dynamics based on the big data approach," *European Control Conference*, pp. 219–224, 2018.

[2] D. Fényes, B. Németh, and P. Gáspár, "Analysis of autonomous vehicle dynamics based on the big data approach," in *European Control Conference*, June 2018, pp. 219–224.

[3] L. Li, F.-Y. Wang, Q. Zhou, and G. Shan, "Automatic tire pressure fault monitor using wavelet-based probability density estimation," *Vehicle System Dynamics*, pp. 80–84, 2003.

[4] H. Mayer, "Model based detection of tyre deflation by estimation of a virtual transfer function," *Proceedings of IEEE Conference on Control Applications*, pp. 285–290, 1995.

[5] S. Solmaz, "A novel method for indirect estimation of tire pressure," *9th Asian Control Conference (ASCC)*, 2013.

[6] F. Wang, H. Chen, H. Guo, and D. Cao, "Constrained h8 control for road vehicles after a tire blow-out," *Mechatronics*, pp. 371–382, 2015.

[7] H. Guo, F. Wang, H. Chen, and D. Guo, "Stability control of vehicle with tire blowout using differential flatness based mpc method," *Proceedings of the 10th World Congress on Intelligent Control and Automation*, pp. 2066–2071, 2012.

[8] I. Witten and E. Frank, *Data Mining Practical Machine Learning Tools and Techniques*. Elsevier, 1999.

[9] Y. Wang and I. H. Witten, *Pace Regression*. (Working paper 99/12). Hamilton, New Zealand: University of Waikato, Department of Computer Science., 1999.

[10] R. Rajamani, "Vehicle dynamics and control," *Springer*, 2005.

[11] H. B. Pacejka, *Tyre and vehicle dynamics*. Oxford: Elsevier Butterworth-Heinemann, 2004.

[12] L. Ljung, *System identification: theory for the user*. USA: Prentice-Hall, 2003.