Lateral Control Design for Autonomous Vehicles Using a Big Data-Based Approach

Dániel Fényes(✉), Balázs Németh, and Péter Gáspár

Systems and Control Laboratory, Institute for Computer Sciences and Control, Hungarian Academy of Sciences, Kende u. 13-17, Budapest 1111, Hungary
{daniel.fenyes,balazs.nemeth,peter.gaspar}@sztaki.mta.hu

Abstract. In the paper an improved Model Predictive Control (MPC) design is presented for autonomous vehicles. The improvement of the control design is based on big data analysis of the lateral vehicle dynamics. In the big data analysis, the decision tree algorithm, C4.5 is used to determine the stable regions of the vehicle. Moreover, C4.5 is extended with the MetaCost algorithm, which is able to weight the percentages of certain misclassifications. In this way, the safe motion of the vehicle can be guaranteed. The results of the big data analysis are states-sets, which are used as constraints in the MPC control design.

Keywords: Big data analysis · Model Predictive Control · Machine learning

1 Introduction

The application of the big-data methods is a novel field of vehicle control. One of the most important applications of big data is the coordination of autonomous vehicles in vehicular networks [1]. Another field of big data is its application in individual autonomous vehicles for estimation, prediction and control purposes. Big data provides large amount of relevant information about the environment, with which the perception can be improved [2]. Moreover, big data have been used in the prediction of vehicle slip through the combination of individual measurements of the vehicle and database information [3]. Another application of big data techniques in vehicle control is the estimation of the adhesion coefficient (μ) between tire and the road. The estimation of the road surface is still a challenging problem since the relationship between the slip angle and the tire force is highly nonlinear and depends on several external circumstances. For example, a BP neural network based solution is described in [6], which is able to estimate μ coefficient at stationary velocities. Apart from the neural networks, other machine learning techniques can be used for estimating the road surface. A support vector machines (SVM) based estimation can be found in [5]. Of course, in the literature, many other big data and machine learning based solutions can be found, which are related to vehicle control problems.

M. Klomp et al. (Eds.): IAVSD 2019, LNME, pp. 1137–1143, 2020.
https://doi.org/10.1007/978-3-030-38077-9_133

The contribution of the paper is the improvement of lateral autonomous vehicle control design through vehicle dynamic constraints, which are derived from the big data analysis on the measured signals. The analysis is performed through machine learning methods, such as C4.5 and MetaCost decision tree generation algorithms [4]. The main goal of the MetaCost algorithm is to decrease the percentage of the misclassification of unstable instances. It is crucial issue since this kind of misclassification can result in unstable motion of the vehicle. The results provide information about the state space regions of the vehicle, in which its motion can be acceptable from the viewpoint of path following. The tracking control problem of the autonomous vehicle is formed in a Model Predictive Control (MPC) structure, in which the result of the big data analysis is incorporated. Through the enhanced design algorithm the path following functionality of the autonomous vehicles in a wide set of vehicle maneuvers is improved.

2 Big Data-Based Analysis of Lateral Vehicle Dynamics

For big data-based analyses, a lot of measured data is required. In this paper, the dataset is provided by the high fidelity simulation software CarSim, in which numerous simulations have been performed with various parameters, e.g. longitudinal velocity v_x, steering angle δ, yaw-rate $\dot{\psi}$, side-slip β, adhesion coefficient, etc. In this way more than 10 million instances have been saved.

Initially, the instances have been divided into two classes, such as 'acceptable' and 'unacceptable'. The acceptable class includes the instances, in which the lateral vehicle motion is considered to be stable and with good tracking performances. The approach of this paper is based on the idea that the motion of the vehicle is generally acceptable in the linear region of the tire force characteristics. The defined criterion expresses the similarity between the current side-slip of the front axle $(1 + \alpha_1)$ and the expected side-slip based on the linear formulation of the vehicle:

$$\text{if } -\varepsilon \leq \frac{|1+\alpha_1|}{|1+\delta-\beta-\frac{l_1\dot{\psi}}{v_x}|} - 1 \leq \varepsilon, \text{ then } i^{th} \text{ instance is acceptable,} \tag{1}$$

where ε is an experimentally defined parameter. Further instances are classified as 'unacceptable'.

The main purpose of the big data analysis is to create an agent, which is able to classify the instances as acceptable/unacceptable using only the measured attributes. In this paper the agent is a decision tree, which is designed through the widely-used C4.5 machine learning algorithm, see [7]. The main advantage of the decision trees is that their results can be visualized easily and the formed rules can be used for on-line classification. The fundamental concept of the algorithm is to create subsets, whereby the entropy $E(S)$ of the dataset S can be minimized.

$$E(S) = \sum_{i=1}^{n} p_i log_2 p_i \tag{2}$$

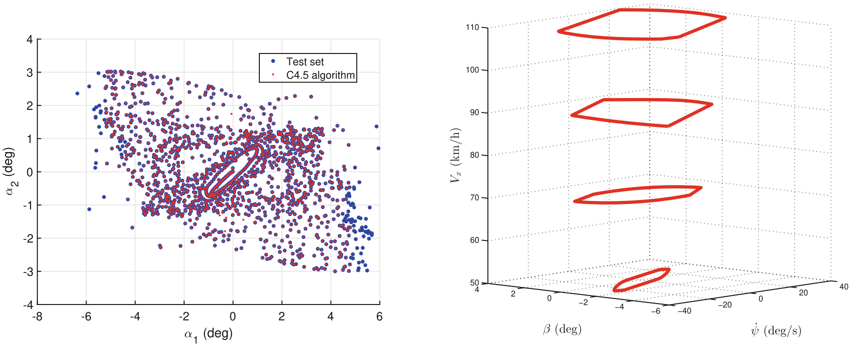
where p_i represent the probabilities of the classes.

During the generation of the decision tree the misclassification of the 'unacceptable' instances must be minimized. Therefore, the MetaCost algorithm is also used, with which a specific misclassification can be penalized during the classification process. Briefly, the classification is performed in the following way:

1. The C4.5 algorithm is performed on the data set S, which results in the decision tree T.
2. Through the MetaCost algorithm the results of T are evaluated.
3. If the result of the confusion matrix is not acceptable, the original data set S is extended with some new elements, which are related to the misclassified instances, considering the weights of misclassification.
4. Then, C4.5 is performed again on the extended data set to generate new T.

The detailed description of the optimization task and its iterative solution are found in [4].

As an example, the result of the decision tree is illustrated in Fig. 1. The validation of the decision tree is shown in Fig. 1(a), which illustrates the 'acceptable' instances of the test set (blue). It can be seen that the results of the generated decision tree (red) fits well on the test data, there are only few misclassified instances. The computation is performed in a wide range of instances, which yielded Fig. 1(b), in which the resulted 'acceptable' sets in the plane of $\dot{\psi}$ and β at different v_x are illustrated.



(a) α_1 and α_2 sets (b) $\dot{\psi}$ and β sets

Fig. 1. Results of the decision tree

3 Design of Model Predictive Control Using Big Data

In this section, a MPC based control design is presented for passenger cars. The goal of the control design is to guarantee the trajectory tracking of the vehicle. The conventional MPC design is extended with the results of the big data analysis. These results are built in the control design as constrains for states of the vehicle. The control design is based on the following lateral vehicle model, which is described by three equations:

$$mv_x(\dot{\psi} + \dot{\beta}) = C_1\alpha_1 + C_2\alpha_2, \tag{3a}$$

$$J\ddot{\psi} = C_1\alpha_1 l_1 - C_2\alpha_2 l_2, \tag{3b}$$

$$\dot{v}_y = v_x(\dot{\psi} + \dot{\beta}), \tag{3c}$$

where J is the yaw inertia, m is the mass of the vehicle C_i represents cornering stiffness on the front and the rear axles and l_i is the distance between vehicle's COG and the wheels and $\alpha_1 = \delta - \beta - \dot{\psi}l_1/v_x$ and $\alpha_2 = \beta + \dot{\psi}l_2/v_x$ are the side-slip angles of the wheels. The lateral velocity of the vehicle is v_y, from which the lateral displacement y can be computed. The basic idea of this model is that the first and rear wheels are replaced by one-one wheels, which are placed on the longitudinal symmetrical axis of the vehicle. Therefore, α_1, α_2 are the averaged side-slip angles of the front and rear wheels. For the MPC design, this model is transformed into a state space representation, whose states are $x = \left[\beta\ \dot{\psi}\ v_y\ y\right]^T$ and the state space is:

$$\dot{x} = Ax + Bu, \tag{4}$$

where u is the steering angle.

The MPC control design requires a discrete-time model of the continuous system, therefore the presented state space representation is discretized using the sampling time T_s. The discrete state-space representation is:

$$x(k+1) = \phi x(k) + \Gamma u(k), \tag{5}$$

The motion of the vehicle is predicted for n steps ahead of the vehicle.

$$y_{pred}(k,n) = \begin{bmatrix} y(k+1) \\ y(k+2) \\ \vdots \\ y(k+n) \end{bmatrix} \tag{6}$$

This prediction is calculated from the discrete state space representation of the lateral vehicle model. Of course, the main goal of the control design is to guarantee the trajectory tracking of the vehicle, which can be formalized as the minimization of the tracking error:

$$e_y(k,n) = y_{ref}(k,n) - y_{pred}(k,n), \tag{7}$$

where the reference signal is derived from the road geometry, which is considered to be known, at least, n-step ahead. These predefined performances can be guaranteed through a cost function, such as:

$$J = \frac{1}{2}e_y(k,n)^T Q e_y(k,n) + U(k,n)^T R U(k,n), \tag{8}$$

where $U(k,n) = \left[u(k) \ldots u(k+n-1)\right]^T$. Moreover, Q and R are weighting matrices, which guarantee a balance between lateral error minimization and control actuation. Using (6) and (7) the cost function can be transformed into the following MPC problem:

$$\min_{U(k,n)} U(k,n)^T \sigma U(k,n) + \nu^T U(k,n). \tag{9}$$

The solution of (9) can ensure the trade-off between the tracking and the steering of the vehicle. Nonetheless, this linear approach can sometimes result in the unstable motion of the vehicle. Therefore, this MPC solution is extended with the result of the presented big data analysis. It means that the MPC controller must guarantee the trade-off between the tracking and the actuation while, in parallel, it must also guarantee that the states of the vehicle stay inside the presented sets. This condition can be formalized as:

$$U_{max} = \left[u_{max} \ldots u_{max}\right]^T, \qquad U_{min} = \left[u_{min} \ldots u_{min}\right]^T, \tag{10a}$$

and the sizes of both vectors are $n-1 \times 1$. The upper and lower limits must guarantee that the yaw rate and the side-slip angle of the vehicle are inside the sets of the acceptable states (R_{good}) on the horizon n ahead. The prediction of $\dot{\psi}(k+1) \ldots \dot{\psi}(k+n)$ and $\beta(k+1) \ldots \beta(k+n)$ are computed as

$$\dot{\psi}_{pred}(k,n) = \begin{bmatrix} \dot{\psi}(k+1) \\ \dot{\psi}(k+2) \\ \vdots \\ \dot{\psi}(k+n) \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}^T \begin{bmatrix} \phi \\ \phi^2 \\ \vdots \\ \phi^n \end{bmatrix} x(k)$$

$$+ \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}^T \begin{bmatrix} \Gamma & 0 & \cdots & 0 \\ \phi\Gamma & \Gamma & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \phi^{n-1}\Gamma & \phi^{n-2}\Gamma & \cdots & \Gamma \end{bmatrix} \begin{bmatrix} u_i(k) \\ u_i(k+1) \\ \vdots \\ u_i(k+n-1) \end{bmatrix}, \tag{11a}$$

$$\beta_{pred}(k,n) = \begin{bmatrix} \beta(k+1) \\ \beta(k+2) \\ \vdots \\ \beta(k+n) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}^T \begin{bmatrix} \phi \\ \phi^2 \\ \vdots \\ \phi^n \end{bmatrix} x(k) \tag{11b}$$

$$+ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}^T \begin{bmatrix} \Gamma & 0 & \cdots & 0 \\ \phi\Gamma & \Gamma & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \phi^{n-1}\Gamma & \phi^{n-2}\Gamma & \cdots & \Gamma \end{bmatrix} \begin{bmatrix} u_i(k) \\ u_i(k+1) \\ \vdots \\ u_i(k+n-1) \end{bmatrix},$$

where u_i represents u_{min} or u_{max}. Moreover, it is necessary to select u_{min} and u_{max}, so that R_{good} contains the entire trajectory:

$$\max u_{max} \quad \text{s.t.} \quad \dot{\psi}_{pred}(k,n), \beta_{pred}(k,n) \in R_{good}, \tag{12a}$$

$$\min u_{min} \quad \text{s.t.} \quad \dot{\psi}_{pred}(k,n), \beta_{pred}(k,n) \in R_{good}. \tag{12b}$$

The result of (12a) is formed in a constraint on the control input $U(k,n)$

$$MU(k,n) \leqq H, \tag{13}$$

where

$$M = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}, \quad H = \begin{bmatrix} U_{max} \\ -U_{min} \end{bmatrix}. \tag{14}$$

Finally, the improved MPC optimization task is formed using (9) and (13)

$$\min_{U(k,n)} U(k,n)^T \sigma U(k,n) + \nu^T U(k,n) \tag{15a}$$

$$\text{s.t.}$$

$$MU(k,n) \leqq H. \tag{15b}$$

The result of the optimization is $U(k,n)$, and it is necessary to actuate $u(k) = \delta(k)$ at the time step k of the computation.

4 Conclusion

The paper has presented a new MPC based lateral control design for autonomous vehicles. The conventional control algorithm has been extended with the result of the big data analysis. The big data analysis has been carried out with the well-known decision tree algorithm, C4.5. In addition, the result of the decision has been improved by the MetaCost algorithm. In this way, the percentage of the misclassified 'bad' instances has been reduced, which enhanced the stability of the vehicle. Finally, the last section has shown a way to build the result of the decision tree into the MPC problem as constraints for the input signal.

Acknowledgment. This work has been supported by the GINOP-2.3.2-15-2016-00002 grant of the Ministry of National Economy of Hungary and by the European Commission through the H2020 project EPIC under grant No. 739592.

The work of Balázs Németh was partially supported by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences and the ÚNKP-18-4 New National Excellence Program of the Ministry of Human Capacities.

The work of Dániel Fényes was partially supported by the ÚNKP-18-3 New National Excellence Program of the Ministry of Human Capacities.

References

1. Cheng, N., Lyu, F., Chen, J., Xu, W., Zhou, H., Zhang, S., Shen, X.S.: Big data driven vehicular networks. IEEE Netw. **32**(6), 160–167 (2018)
2. Coyte, J., Li, B., Du, H., Li, W., Stirling, D., Ros, M.: Decision tree assisted EKF for vehicle slip angle estimation using inertial motion sensors. In: IEEE International Joint Conference on Neural Network (IJCNN 2014), Beijing, China (2014)

3. Fenyes, D., Nemeth, B., Asszonyi, M., Gaspar, P.: Side-slip angle estimation of autonomous road vehicles based on big data analysis. In: 26th Mediterranean Conference on Control and Automation, pp. 849–854, June 2018
4. Domingos, P.: MetaCost: a general method for making classifiers cost-sensitive. In: 5th International Conference on Knowledge Discovery and Data Mining, San Diego, pp. 155–164 (2019)
5. Li, S., Pei, X., Ma, Y.: A new road friction coefficient estimation method based on SVM In: 2012 IEEE International Conference on Mechatronics and Automation, pp. 1910–1914 (2012)
6. Song, S., Min, K., Park, J., Kim, H., Huh, K.: Estimating the maximum road friction coefficient with uncertainty using deep learning. In: 21st International Conference on Intelligent Transportation Systems, pp. 3156–3161 (2018)
7. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo (1993)