2017 IEEE Intelligent Vehicles Symposium (IV)
June 11-14, 2017, Redondo Beach, CA, USA

A machine learning approach for personalized autonomous lane change initiation and control

Charlott Vallon[1], Ziya Ercan[2], Ashwin Carvalho[3] and Francesco Borrelli[3]

Abstract— **We study an algorithm that allows a vehicle to autonomously change lanes in a safe but personalized fashion without the driver's explicit initiation (e.g. activating the turn signals). Lane change initiation in autonomous driving is typically based on subjective rules, functions of the positions and relative velocities of surrounding vehicles. This approach is often arbitrary, and not easily adapted to the driving style preferences of an individual driver. Here we propose a data-driven modeling approach to capture the lane change decision behavior of human drivers. We collect data with a test vehicle in typical lane change situations and train classifiers to predict the instant of lane change initiation with respect to the preferences of a particular driver. We integrate this decision logic into a model predictive control (MPC) framework to create a more personalized autonomous lane change experience that satisfies safety and comfort constraints. We show the ability of the decision logic to reproduce and differentiate between two lane changing styles, and demonstrate the safety and effectiveness of the control framework through simulations.**

I. INTRODUCTION

While vehicles with level four or five autonomy remain rare, vehicles with partially automated driving features already exist on our roads. Common active safety systems such as cruise control and braking assistance are implemented regularly in a variety of vehicles, with great success and popularity [1]. While users recognize active safety features as undoubtedly beneficial, some are concerned that the car's control software will not respond to road situations the same way that individual driver would [2]. One approach to address this concern is to design controllers that replicate human driving behavior as closely as possible, while still observing important safety bounds. Critically, such controllers should be tuned, or tunable, to an individual driver, rather than an arbitrarily chosen 'representative' driver. This approach has already been shown to work well in a lane keeping, car following controller [3].

Ten percent of all freeway crashes occur during lane change maneuvers [4]. It stands to reason, therefore, that well-designed autonomous lane change controllers could create a significantly safer driving experience if drivers feel confident and secure enough to utilize them.

*This work was not supported by any organization

[1]Charlott Vallon is with the Department of Mechanical and Process Engineering, ETH Zurich, Zurich, Switzerland vallonc@ethz.ch

[2]Ziya Ercan is with the Department of Control and Automation Engineering, Istanbul Technical University, Istanbul, Turkey zyercan@gmail.com

[3]Ashwin Carvalho and Francesco Borrelli are with the Department of Mechanical Engineering, University of California Berkeley, Berkeley, CA 94720, USA {ashwinc,fborrelli}@berkeley.edu

Previous works on autonomous lane change control have focused heavily on designing natural lane change trajectories [5], [6], [7], [8]. For example, the authors in [7] used a total kinetic energy minimization approach superimposed on an underlying polynomial equation to determine a trajectory for a smooth and ergonomic lane change maneuver. In [5], lane change trajectory is calculated by solving a time-optimal control problem that minimizes maneuver time with fixed end constraints and bounds on lateral acceleration and jerk. These works attempt to capture human driving behavior by assuming a parametric base form and then carefully tuning appropriate parameters through experimental validation.

In contrast, learning-based trajectory generation approaches utilize real driving data to design natural trajectories. The work in [6] presents a learning algorithm to model individual driving styles: trajectories are defined by spline segments and the coefficients of the polynomial are learned from data rather than manual tuning. A lane change trajectory is determined by averaging the K-nearest instances in an online database of collected lane change trajectories, based on similar driving environments at lane change initiation in [8]. These learning-based approaches can replicate human driving behavior more closely than parametric methods are able to, because they utilize actual trajectory data.

All these works focus on trajectory generation, and their proposed control schemes are unable to initiate a lane change independently of the driver. Much research exists on autonomous obstacle detection and avoidance [9], [10], and such controllers do initiate lane changes in emergency situations. However, most natural lane changes are primarily motivated by a non-emergency desire to drive at a faster longitudinal speed than the preceding car. To our knowledge, no algorithm currently exists that allows a vehicle to autonomously change lanes in a safe but personalized fashion without the driver's explicit initiation (e.g. activating the turn signals).

In this paper, we present an autonomous lane change algorithm where the lane change decision is determined by a support vector machine (SVM) based classifier. We train several SVMs using data from actual lane changing and lane keeping demonstrations by human drivers. The SVM learns whether to continue lane keeping or initiate a lane change, based on the demonstrated preferences of an individual driver. A trajectory is then generated using the output of the classifier. A predictive control framework optimizes the control inputs in order to follow the reference trajectory while satisfying comfort and safety constraints. It is clear that by relying on experimental lane change training data, we can directly capture and replicate natural human driving behavior. Furthermore the lane change execution is adapted to the driving style of a particular driver, therefore resulting in a more personalized driving experience than existing controllers. Our ultimate goal is to improve passenger confidence, which results in fewer driver interventions and safer autonomous vehicle operation.

The remainder of the paper is organized as follows. Section II introduces the proposed system architecture. Details of the individual modules are described in Sections III - V. The results are discussed in Section VI and Section VII concludes the paper.

II. System Architecture

We present an autonomous lane change initiation and control architecture where an SVM-based decision logic is employed to initiate a non-emergency lane change, and a predictive controller executes the maneuver. The proposed autonomous driving framework introduces a “Lane Change Decision and Motion Planner” module, and is organized as follows (see Fig. 1):

- The “Perception & Localization” module integrates the measurements and data from various sensors and offline maps to produce an estimate of the relevant environmental states, including positions and velocities of nearby vehicles.
- The “Target-Ego Vehicle Interaction” module predicts the surrounding target vehicles’ response to the ego vehicle’s current and future states.
- The novel “Lane Change Decision and Motion Planner” employs a machine-learning method to predict the high-level behavior of the driver (e.g. lane keeping or changing) and accordingly generates a reference trajectory.
- The “MPC Controller” solves a finite time optimal control problem to optimize the actuator commands (e.g. acceleration, steering angle) so as to track the reference trajectory while satisfying actuator and safety constraints.

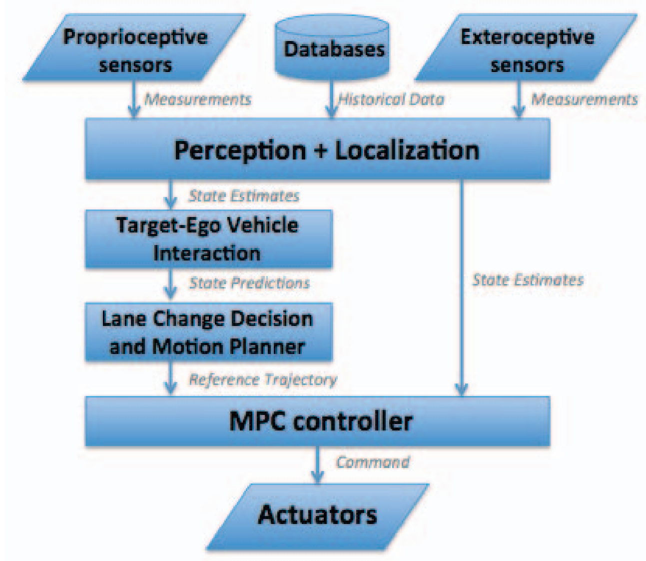

Fig. 1. The flowchart of the proposed architecture.

III. Target-Ego Vehicle Interaction Module

The “Target-Ego Vehicle Interaction” module predicts the future states of nearby target vehicles as they respond to the actions of the ego vehicle. For simplicity we utilize the Intelligent Driver Model (IDM) to predict the motion of target vehicles. The IDM has two modes: free-flow and car-following. In *free-flow* mode, the target vehicles track a desired speed. When there is a preceding vehicle, the mode changes to *car-following*, and the target vehicle follows the preceding vehicle at a safe distance. Detailed mathematical description of the model is presented in [11].

IV. Lane Change Decision and Motion Planner Module

A lane change decision model calculates a binary mode indicating if the current driving environment would typically cause a particular driver to continue lane keeping (mode -1) or initiate a lane change (mode 1). The module then generates an appropriate reference trajectory for the MPC controller.

We use a Support Vector Machine (SVM) classifier trained on experimental lane change data from particular drivers to calculate the mode. The data-driven nature of this approach ensures that the module produces lane change initiation behavior that resembles the natural driving strategy of the individual driver on whose data the SVM was trained.

A. Support Vector Machines

SVMs are a supervised learning method for data classification using low computational effort [12]. Each data vector χ_i in the training data set $\mathbf{X} = [\chi_1, ..., \chi_N] \in \mathbb{R}^{KxN}$ is labeled as belonging to one of two classes $y_i \in \{-1, 1\}$ where N is the size of the data set. In this work, these classes denote ‘maintain lane keeping’ and ‘initiate lane change’, respectively. Each data vector in $\mathbf{X}$ consists of K features believed to be influential in determining the associated class of a data point. The SVM learns the maximum-margin hyperplane that separates the two classes from each other in the K-dimensional feature space. New data vectors of unknown class can then be assigned to a class according to their location in the feature space with respect to this hyperplane boundary. The resultant hyperplane is entirely data-dependent, and thus SVMs trained on different driving data will look for different environmental conditions before initiating the lane change.

Mathematically, this hyperplane is determined by solving

$$\min_{\mathbf{w},b} \left[\frac{1}{N} \sum_{i=1}^{N} \max\left((0, 1 - y_i(\mathbf{w}^\intercal \chi_i - b) \right) \right] + \lambda ||\mathbf{w}||, \quad (1)$$

where the parameters $\mathbf{w}$ and b define the decision boundary. In (1), λ specifies the balance between the bias and accuracy of the hyperplane: an SVM with small λ will emphasize classification accuracy, whereas an SVM with large λ will

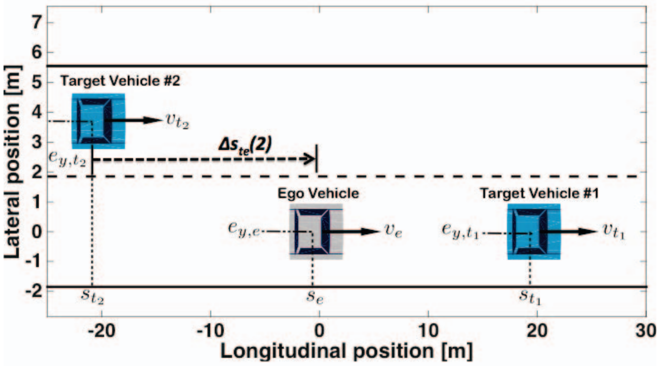

Fig. 2. The scenario that is considered for lane change decision model where M=2.

emphasize a wide margin [13]. In this work, we use cross-validation to determine the optimal parameter lambda based on the given data set.

B. Data Collection

We performed the data collection with our test vehicle at the Berkeley Global Campus in Richmond, California. We considered the scenario illustrated in Fig. 2, where the ego vehicle (EV) and two target vehicles (TV_1 and TV_2) are traveling on a straight road. TV_1 is traveling slower than the EV, eventually motivating a lane change maneuver for the driver of EV at a time which feels secure and comfortable.

Two drivers were instructed to perform 25 such lane changes as naturally as possible, and their data was recorded.

C. Training Phase

The training features are selected to be the relative distances and longitudinal velocities between the EV and each of the M TVs. The feature vector is given as:

$$\chi = [\Delta \mathbf{v}_{te} \quad \Delta \mathbf{s}_{te}]^\intercal \in \mathbb{R}^{2M}, \tag{2}$$

where $\Delta \mathbf{v}_{te}$ and $\Delta \mathbf{s}_{te}$ are the vectors of relative velocities and relative positions between the EV and all TVs. An example for determining $\Delta s_{te}(2)$ is shown in Fig. 2. Based on preliminary data analysis, we believe these states to be the most critical in determining the high-level driving.

As explained in the previous section, each SVM training feature vector is labeled as belonging to one of two classes. We label the feature vectors as $y_i = 1$ if the driver of EV initiates or performs a lane change at that time instance. A situation denotes a lane change if the driver is more than $1.5[ft]$ from the lane center line. We label the feature vectors as $y_i = -1$ when the driver prefers to stay in the lane.

We train the personalized SVMs using the data sets X and class labels Y of each driver. Each feature vector χ of an individual data set is standardized and normalized before training.

D. Motion Planner

At each time instance, the trained SVM (i.e. decision model) considers the current state estimates of the EV and TVs from the Target-Ego Vehicle Interaction module and outputs a high-level driving mode (e.g. lane keeping or lane changing) sequence to the motion planner, which in turn generates an appropriate reference trajectory sequence to be employed in the controller.

In this paper, we use a simple approach to calculate an appropriate trajectory. If the SVM decision model determines that the driver would continue lane keeping, the motion planner creates a lane-center reference trajectory in the lane the vehicle is already in. On the other hand, if the decision model determines that a lane change should be initiated, the motion planner creates a reference trajectory along the center of the adjacent lane beginning at the time instance the decision model switches classes.

V. The Controller

The objective of the controller is to track the motion planner's reference trajectory while operating within safety constraints and vehicle actuator limits.

A. The Vehicle Model

We use the nonlinear dynamic bicycle model to describe the dynamics of the ego vehicle, as presented in [14]. This model accounts for both longitudinal and lateral dynamics, and considers control inputs of steering angle (δ) and longitudinal acceleration (a_x). The model defines the following state vector:

$$\mathbf{x^e} = [\dot{x}^e,\ \dot{y},\ \dot{\psi},\ e_\psi,\ e_y,\ s^e]^\intercal \in \mathbb{R}^6, \tag{3}$$

where $\dot{x}^e$, $\dot{y}$ and $\dot{\psi}$ are defined in vehicle body axis and they denote the longitudinal and lateral velocity and the yaw rate, respectively. e_y, e_ψ and s^e are defined in a road-aligned coordinate frame and denote the lateral distance, orientation angle from the center lane and the longitudinal position of the vehicle along lane center line, respectively. The differential equations for this model are described in detail in [14]. We describe the state vector using the Euler discretization:

$$\mathbf{x^e_{i+1}} = f(\mathbf{x^e}_i, \mathbf{u}_i), \tag{4a}$$

where $\mathbf{u}_i = [\delta_i,\ a_{x,i}]^\intercal$ contains the steering and acceleration control inputs at time step i.

B. Safety and Actuator Constraints

The main objective of the controller is to ensure safety while tracking the desired trajectory with smooth and minimal control utilization. Therefore we impose following constraints on the vehicle states and control inputs:

1) Safety Constraints: We formulate the safety constraints both on longitudinal and lateral dynamics of EV. We express the time-varying lateral safety constraints over the prediction horizon H_p as:

$$e_{y_{i,\min}} \leq e_{y,i} \leq e_{y_{i,\max}} \qquad i = 1 \dots H_p, \tag{5}$$

where $e_{y_{i,\min}}$ and $e_{y_{i,\max}}$ are the lower and upper bounds on the lateral position prediction of EV at prediction index i. These bounds change with respect to the presence of a TV in the adjacent lane as described in [15].

The proposed controller should avoid collisions with the preceding vehicle by maintaining a safe following distance. This is enforced by the following longitudinal safety constraints on the relative distance between the vehicles:

$$s_i^p - s_i^e \geq d_{safe} \qquad i = 1 \dots H_p, \tag{6}$$

where s_i^p is the longitudinal position prediction of the preceding vehicle at time index i, forecast using the Intelligent Driver Model [11]. d_{safe} is the minimum safety distance.

The safety constraints (5) (6) are compactly written as:

$$h^x(\mathbf{x}_{i+1}, \mathbf{u}_i) \leq \mathbf{0} \qquad i = 0 \dots H_p - 1, \tag{7}$$

where $\mathbf{0}$ is a vector of zeros with appropriate dimension.

2) Actuator Constraints: We consider the physical limitations of the actuators by enforcing the following constraints on the control inputs:

$$a_{x,min} \leq a_x \leq a_{x,max}, \tag{8a}$$
$$\dot{a}_{x,min} \leq \dot{a} \leq \dot{a}_{x,max}, \tag{8b}$$
$$\delta_{min} \leq \delta \leq \delta_{max}, \tag{8c}$$
$$|\dot{\delta}| \leq \dot{\delta}_{max}. \tag{8d}$$

The control input constraints (8) are compactly written as:

$$h^u(\mathbf{u}_i) \leq \mathbf{0} \qquad i = 0 \dots H_p - 1, \tag{9}$$

where $\mathbf{0}$ is a vector of zeros with appropriate dimension.

Parameter values of the actuator limits and safety constraints are defined in Table I.

TABLE I
MPC CONTROLLER DESIGN VALUES

Parameter	Value	Parameter	Value
d_{safe}	$5\ [m]$	$\dot{\delta}$	$4\ [deg/s]$
$a_{x,min}$	$-3\ [m/s]$	δ_{min}	$-8\ [deg]$
$a_{x,max}$	$3\ [m/s]$	δ_{max}	$8\ [deg]$
$\dot{a}_{x,min}$	$-1.5\ [m/s^2]$	$\dot{a}_{x,max}$	$3\ [m/s^2]$
H_p	20	Q	$\mathbf{diag}(50\ 1)$
R	$\mathbf{diag}(100\ 2)$	T	$\mathbf{diag}(1\ 10\ 10\ 0\ 10\ 0)$

C. Optimal Control Problem

The constrained finite time optimal control problem

$$\min_{\mathbf{U}_k, \varepsilon} \sum_{i=0}^{H_p-1} \left(\|\mathbf{u}_{i|k}\|_Q^2 + \|\dot{\mathbf{u}}_{i|k}\|_R^2 \right) \tag{10a}$$
$$+ \sum_{i=1}^{H_p} \|\mathbf{x}_{i|k} - \mathbf{x}_{i|k}^{ref}\|_T^2 + W\varepsilon$$
$$\text{s.t. } \mathbf{x}_{i+1|k} = f(\mathbf{x}_{i|k}, \mathbf{u}_{i|k}), \tag{10b}$$
$$h^x(\mathbf{x}_{i+1|k}, \mathbf{u}_{i|k}) \leq \varepsilon, \tag{10c}$$
$$h^u(\mathbf{u}_{i|k}) \leq \mathbf{0}, \tag{10d}$$
$$\varepsilon \geq \mathbf{0}, \qquad i = 0 \dots H_p - 1$$
$$\mathbf{x}_{0|k} = \mathbf{x}_k, \qquad \mathbf{u}_{-1|k} = \mathbf{u}_{k-1}, \tag{10e}$$

is solved at each sampling interval k to compute an optimal control input sequence $\mathbf{U}_k$. Only the first control inputs of the sequence are applied to the vehicle, and this optimization routine is repeated at the next time step using new state measurements as initial conditions. $\mathbf{x}_{i|k}$ denotes the ith state prediction at time step k, obtained by applying the optimal control input sequence $\mathbf{U}_k = \{\mathbf{u}_{0|k}, \mathbf{u}_{1|k}, \dots, \mathbf{u}_{H_p-1|k}\}$ to the discrete-time dynamics (10b) starting from the measured state at the current time step (10e). The cost function in (10a) reflects control objectives of utilizing smooth and minimal control action while tracking the desired trajectory and velocity. We penalize the magnitude and rate of control inputs as well as the tracking error, using corresponding weight matrices Q, R and T, respectively. The input constraints are given in (10d). We introduce slack variable vector ε for the safety constraints (10c) in order to keep the optimization problem feasible in case of constraint violations due to model mismatch. A large matrix W strongly penalizes constraint violation ε.

VI. RESULTS

We present the results from the validation of the proposed lane change decision model and the evaluation of the predictive controller for personalized and safe autonomous driving.

A. Decision Model Performance

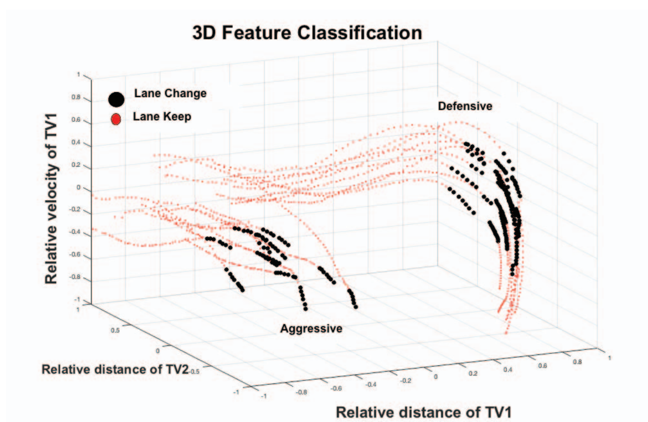

Fig. 3. Three-dimensional representation of the feature space for Driver A. Data becomes separable with the inclusion of relative velocity information.

Driver A's training data is shown in Fig. 3, plotted in a reduced normalized feature space (i.e. the plot only includes three of four SVM features). The plot shows separable lane change intent classes, with clear clusters of lane changing and lane keeping data points. This suggests that the selected features are sufficient to learn a separating hyperplane.

The plot shows defensive and aggressive lane changes. The defensive lane changes, towards the right of the plot, occur only once TV_2 has passed the EV, after the relative distance of TV_2 becomes positive. In this case, the TV_2 travels faster than EV, which forces the EV to wait until TV_2 passes before executing the lane change into the adjacent lane.

The line segments towards the left of the plot, in contrast, represent aggressive lane changes during which EV cuts in front of TV_2. TV_2 maintains a negative relative longitudinal position to EV for the entire lane change process. TV_2 consistently moves with lower velocity than EV, allowing enough space for EV to complete the lane change.

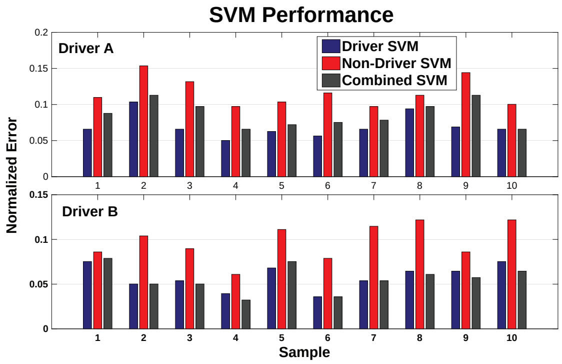

Fig. 4. A comparison of the normalized classification errors of the three SVM decision models tested on test sets from Driver A and Driver B.

SVM performance is evaluated using test data sets that were not used to train the SVMs. The personalized SVMs classify this new data, and the predictions are compared to the true class labels. Normalized testing errors (i.e. the number of instances of classification mismatch between the SVM and the true label, normalized over the size of the test set) for three types of SVMs and ten randomly selected test sets from each driver are illustrated in Fig. 4.

A "Driver SVM" represents a personalized SVM trained only on data from the same driver as the test set. In the top plot of Fig. 4, the Driver SVM was trained and tested using different data sets from Driver A. The Driver SVM performance is consistent for different test sets, with average normalized errors of 6% and 7% in test sets for Drivers A and B, respectively. The SVM is able to learn and reproduce the driving style of a particular driver accurately and reliably.

A "Non-Driver SVM" represents a personalized SVM trained on data from a different driver as the test set. In the top plot, the Non-Driver SVM was trained using data from Driver B, but tested using data from Driver A. On average, Non-Driver SVMs were only half as good at correctly classifying the testing data as Driver SVMs. Even in the exact same driving situation, the two SVMs initiate a lane change at different times, in accordance with their driver's preferences. This suggests that, as expected, there are real lane change initiation behavior differences between drivers, and personalized SVMs tuned to a particular driving style are more apt at reproducing that style. These results provide real motivation for designing personalized controllers so as to account for the demonstrated differences in driver behavior.

Finally, we combined training data sets from Drivers A and B to create a "Combined SVM". The Combined Model classified test data less accurately than the Driver SVM, but more accurately than the Non-Driver SVM. This was expected, as the Combined Model was trained partly with data similar to the testing data, while the Non-Driver decision model had not learned how to classify that driving style.

B. Controller Performance

We validate the effectiveness of the personalized predictive control architecture in simulation. The simulation environment consists of an EV and two simulated TVs, which are initially placed into lanes in a similar setup as that illustrated by Fig. 2. In the simulation presented in Fig. 5, the TV_1 is placed in front of EV in the same lane with a speed of $15m/s$ on the other hand TV_2 is placed behind of the EV in the adjacent lane with a speed of $25m/s$. The objective of the EV controller is to track both the desired speed of $20m/s$ and the motion planner's reference trajectory. The EV is autonomously controlled by its personalized SVM-MPC control framework (e.g. "Driver A" case uses the SVM A Decision Model), while the longitudinal dynamics of TVs are based on the lane-keeping IDM model.

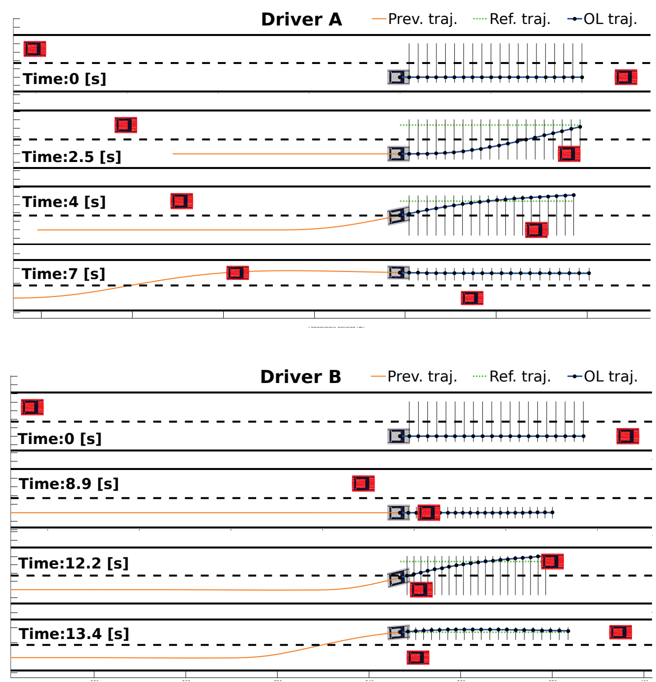

Fig. 5. Two SVM Decision Modules trained on data from Drivers A and B initiate lane changes at different times, reflecting their personal driving styles. Previous, reference, and open loop trajectories are plotted.

Figure 5 shows the differences between the driving styles of Drivers A and B, encoded in their personalized SVMs. In the results of "Driver A", a lane change is initiated by SVM A at $t = 2.5$s, and the EV cuts in front of the approaching TV_2. Under the same initial driving conditions, "Driver B" exhibits more conservative behavior, as SVM B only initiates the lane change after TV_2 has already passed the EV.

While both Decision Models lead to safe lane change maneuvers that do not violate any safety constraints, SVM A represents a clearly more aggressive driving style than SVM B does. This result demonstrates that our SVM Decision Models are able to detect and reproduce different drivers' unique lane change initiation preferences. A control framework without our Decision Model would not have initiated a lane change at all, but cause EV to decelerate to match the longitudinal velocity of TV_1. Similarly, a framework that only initiates emergency lane-changes would never result in the natural driving behavior exhibited by Driver A in Fig. 5.

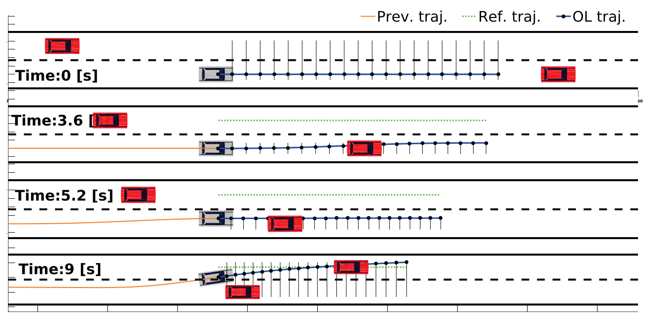

Fig. 6. The MPC framework provides safety guarantees for the data-driven Decision Model.

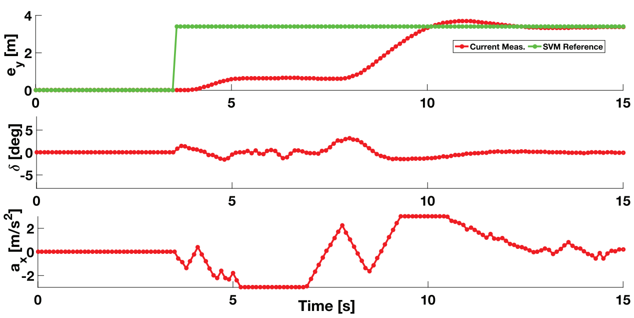

Fig. 7. The lateral position and control inputs adjust to meet the lateral and longitudinal position constraints of the ego vehicle.

We verify the controller's ability to prioritize safety constraints over the motion planner's reference input. The MPC controller tracks the reference trajectory only if doing so will not violate lateral safety constraints at any time in the prediction horizon. In Fig. 6, the decision model initiates a lane change by generating a reference trajectory to adjacent lane at $t = 3.6$ s. However, this trajectory would violate the lateral safety constraints, so the controller keeps the vehicle in the original lane at a safe following distance to the preceding vehicle until TV_2 has passed the EV (see Fig. 7). When the lateral constraints indicate a safe driving corridor for the adjacent lane at $t = 9$ s, the controller tracks the motion planner's original reference trajectory.

Thus, while our strictly data-driven decision models alone do not provide any safety guarantees, the MPC control architecture does provide a reliable safety guarantee. Any dangerous driving habits potentially learned from the training data will not be considered by the autonomous vehicle.

VII. Conclusion and Future Work

This paper presents a data-driven autonomous lane change initiation algorithm that captures and replicates human driving strategies. We use an SVM classifier trained on real driving data to model the personalized lane change decision process of particular drivers. We integrate the SVM Decision Model into a model predictive control framework that calculates the optimal control inputs for autonomous safe driving.

Preliminary results show that the personalized decision models reproduce the lane change initiation behaviors of different drivers, providing a natural and intuitive autonomous driving experience. Moreover, while the data-driven learning method alone does not have any intrinsic safety guarantees, the integrated MPC control framework guarantees safety constraints will be met. Our simulation studies validate the effectiveness of the proposed control framework.

In further work we will replace the IDM Target Vehicle Model with an improved Bayesian filtering model, again trained on real driving data, to give a more accurate prediction of the target vehicle's acceleration response. Furthermore we plan to investigate methods of learning personalized lane change trajectories, so that the entire lane change maneuver can be tuned to any particular driver. Finally, we will conduct experiments with our experimental vehicle in order to validate the proposed controller in automated highway driving scenarios.

References

[1] C. Reports, "Guide to car safety features," accessed Jan 5 2017. [Online]. Available: http://www.consumerreports.org/cro/2012/04/guide-to-safety-features/index.htm

[2] J. Hsu, "75% of u.s. drivers fear self-driving cars, but it's an easy fear to ge over," accessed Jan 5 2017. [Online]. Available: http://spectrum.ieee.org/cars-that-think/transportation/self-driving/driverless-cars-inspire-both-fear-and-hope

[3] S. Lefevre, A. Carvalho, and F. Borrelli, "Autonomous car following: A learning-based approach," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, June 2015, pp. 920–926.

[4] S. Hetrick, "Examination of driver lane change behavior and the potential effectiveness of warning onset rules for lane change or side crash avoidance systems," 1997.

[5] C. Hatipoglu, U. Ozguner, and K. A. Redmill, "Automated lane change controller design," *IEEE Trans. Intell. Transport. Syst.*, vol. 4, no. 1, pp. 13–22, March 2003.

[6] M. Kuderer, S. Gulati, and W. Burgard, "Learning driving styles for autonomous vehicles from demonstration," in *IEEE International Conference on Robotics and Automation, ICRA*, 2015, pp. 2641–2646.

[7] T. Shamir, "How should an autonomous vehicle overtake a slower moving vehicle: design and analysis of an optimal trajectory," *IEEE Trans. Automat. Contr.*, vol. 49, no. 4, pp. 607–610, April 2004.

[8] W. Yao, H. Zhao, P. Bonnifait, and H. Zha, "Lane change trajectory prediction by using recorded human driving data," in *2013 IEEE Intelligent Vehicles Symposium (IV)*, June 2013, pp. 430–436.

[9] G. Freitas and B. Hamner, "A practical obstacle detection system for autonomous orchard vehicles," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 3391–3398.

[10] U. Muller, J. Ben, E. Cosatto, B. Flepp, and Y. L. Cun, "Off-road obstacle avoidance through end-to-end learning," in *Advances in Neural Information Processing Systems 18*, Y. Weiss, P. B. Schölkopf, and J. C. Platt, Eds. MIT Press, 2006, pp. 739–746.

[11] M. Treiber and A. Kesting, *Traffic Flow Dynamics*. Berlin, Germany: Springer, pp. 181–201.

[12] R. Burbidge and B. Buston, *An Introduction to Support Vector machines for Data Mining*. Young OR 12.

[13] L. Rosasco, E. D. Vito, A. Caponnetto, M. Piana, and A. Verri, "Are loss functions all the same?" *Neural Computation*, vol. 16, no. 5, pp. 1063–1076, 2004.

[14] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Kinematic and dynamic vehicle models for autonomous driving control design," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, June 2015, pp. 1094–1099.

[15] Z. Ercan, A. Carvalho, S. Lefèvre, H. Tseng, M. Gokasan, and F. Borrelli, "Torque-based steering assistance for collision avoidance during lane changes," in *13th International Symposium on Advanced Vehicle Control*, 2016.