Scenario Model Predictive Control for Lane Change Assistance and Autonomous Driving on Highways

Gianluca Cesari
ETH Zurich, Zurich, Switzerland
E-mail: gianluca.cesari@me.com

Georg Schildbach, Ashwin Carvalho, and Francesco Borrelli
Hyundai Center of Excellence, University of California Berkeley, Berkeley (CA), United States
E-mail: schildbach@berkeley.edu, ashwinc@berkeley.edu, fborrelli@berkeley.edu

***Abstract*—**This paper presents a novel design of control algorithms for lane change assistance and autonomous driving on highways, based on recent results in Scenario Model Predictive Control (SCMPC). The basic idea is to account for the uncertainty in the traffic environment by a small number of future scenarios, which is intuitive and computationally efficient. These scenarios can be generated by any model-based or data-based approach. The paper discusses the SCMPC design procedure, which is simple and can be generalized to other control challenges in automated driving, as well as the controller's robustness properties. Experimental results demonstrate the effectiveness of the SCMPC algorithm and its performance in lane change situations on highways.

Digital Object Identifier 10.1109/MITS.2017.2709782
Date of publication: 25 July 2017

I. Introduction

A. Traffic Prediction Models

In the past decade, new sensor technology and the increasing capacity of computational hardware has lead to a rising interest in Advanced Driver Assistance Systems (ADAS). A key element for any ADAS feature is the recognition of *traffic scenes* based on measurements of the vehicle's environment via onboard or external sensors. *Traffic predictions* comprise the joint trajectory prediction for all relevant vehicles in the estimated traffic scene, called *target vehicles* (TVs). Difficulties arise from complex vehicle dynamics, intricate tire-road interactions, and uncertainty in humans' driving actions.

Existing models for *traffic predictions* can be classified into physics-based, maneuver-based and

interaction-aware models; see [1] for a survey. Models can be further classified based on the nature of the predictions. *Deterministic models* predict a single trajectory for each object in the traffic scene, and do not reflect any uncertainty associated with the future actions of human drivers [2]. *Set-based predictions* over-approximate the future occupancy of target vehicles, given a prediction model and bounds on the drivers' actions, allowing for a formal verification of safety [3], [4]. *Stochastic models* use standard distribution functions such as Gaussian Mixtures to model a driver's behavior [5], [6]. *Scenario-based models* represent uncertainty by means of samples and do not require an explicit probability distribution for their predictions [7], [8].

B. Model Predictive Control

Model Predictive Control (MPC) is a powerful approach for the optimal control of multi-variable systems with constraints on the inputs and states [9]. MPC has been used successfully for vehicle control in a variety of ADAS applications [10]. The key benefit of MPC is the easy integration of predicted information, as well as constraints resulting from traffic predictions or road geometry.

The most basic approach, Deterministic MPC (DMPC), has successfully been adopted for path following in absence of other traffic [11]. Stochastic MPC (SMPC) is based on probabilistic traffic predictions and has been used for highway driving [12]. In SMPC, chance constraints are used to limit the risk of dangerous events.

However, a major drawback of SMPC is the requirement of a full stochastic uncertainty model and the associated computational complexity with its forecasts, in particular for nonlinear systems or non-Gaussian distributions. This paper examines the use of Scenario MPC (SCMPC) [13], [14], where traffic predictions are based on scenarios. This approach is very intuitive and amenable to numerical processing. Moreover, the approach is very versatile, as the scenarios can be sampled from any stochastic model or data set.

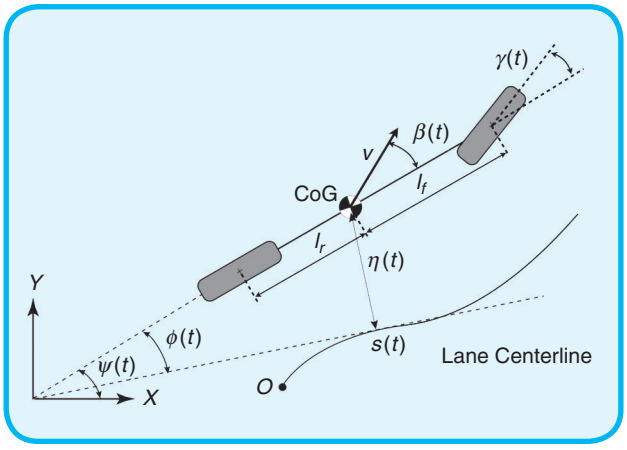

FIG 1 Vehicle position in a Frenét coordinate frame attached to the lane center line.

Recent studies have revealed a direct connection between the number of scenarios and chance constraints on the system states [14], [15], using principles from *scenario optimization* [16], [17]. The number of scenarios required by SCMPC is low, typically less than a hundred for chance constraint probabilities of a few percent. The SCMPC does not guarantee robust safety, but the goal is to explicitly bound the residual risk and relate it to the number of generated scenarios.

C. Scenario-Based Model Predictive Control

In this paper, two SCMPC algorithms for performing safe lane change maneuvers on highways are introduced: one for lane change assistance (LCA) and one for automated highway driving (AHD). A *safe trajectory* is considered to be one that respects all actuator, comfort, and safety constraints.

The main difference between the two proposed algorithms lies in the formulation of the state constraints and the optimization problem for the controller. The LCA algorithm requires that the lane change is initiated by a human driver, e.g., via the turn signal (using it for ADAS), or a higher-level path planner (using it for autonomous driving). The AHD algorithm includes a constraint generation technique which enables the controlled vehicle to plan and execute lane change maneuvers based exclusively on the measured traffic conditions and the commanded reference cruise speed.

D. Notation

Throughout the paper, τ and t are used to denote continuous and discrete time, respectively. The same variable will be used for the continuous signal and its discrete time counterpart. For example, $v(\tau)$ represents a velocity signal in continuous time, with $\tau \in [0,\infty)$ in brackets, and v_t, with $t = 0,1,2,\ldots$ as an index, represents the same signal sampled at the sampling time t_s. A dot will be used to indicate a derivative with respect to (continuous) time, e.g., acceleration $a(\tau) = \dot{v}(\tau)$. Bold letters are used to distinguish vectors and matrices from scalars.

II. Ego Vehicle Model

A. Bicycle Model in Frenét Frame

The SCMPC uses a *kinematic bicycle model* [18, Sec. 2.2] in Frenét coordinates, aligned with the lane centerline [19]; see Fig. 1. The model yields a good trade-off between model complexity and prediction accuracy [11], [20].

The curvilinear abscissa $s(\tau)$ and the distance to the lane center line $\eta(\tau)$ describe the position of the *center of gravity* (CoG) of the *ego vehicle* (EV). The relative orientation $\phi(\tau)$ is the difference between the inertial heading $\psi(\tau)$ and the orientation of the tangent to the lane centerline at the closest point. The model has two *control inputs*, the steering angle and the acceleration $\mathbf{u}(\tau) \triangleq [\gamma(\tau)^{(\mathrm{EV})}, a(\tau)^{(\mathrm{EV})}]$ and five *states* $\mathbf{x}(\tau) \triangleq [s(\tau), \eta(\tau), \phi(\tau), v(\tau), l(\tau)]$, where $l(\tau)$ indicates the current lane of the vehicle. The EV being

in its current lane of width w_{lane}, the lateral distance $\eta(\tau)$ lies within the interval $[-w_{\text{lane}}/2, w_{\text{lane}}/2]$. The following discrete-time equations of motion for the EV are obtained by an explicit Euler method with sampling time t_s:

$$\dot{\tilde{s}}_t = \frac{v_t}{1-\kappa_t \eta_t}\cos(\phi_t + \beta_t) \tag{1a}$$

$$s_{t+1} = s_t + \dot{\tilde{s}}_t t_s \tag{1b}$$

$$\tilde{\eta}_t = \eta_t + v_t \sin(\phi_t + \beta_t) t_s \tag{1c}$$

$$\eta_{t+1} = \tilde{\eta}_t + \begin{cases} -w_{\text{lane}}, & \text{if } \tilde{\eta}_t \geq w_{\text{lane}}/2 \\ w_{\text{lane}}, & \text{if } \tilde{\eta}_t < -w_{\text{lane}}/2 \\ 0, & \text{otherwise} \end{cases} \tag{1d}$$

$$\phi_{t+1} = \phi_t + \left(\frac{v_t}{l_r}\sin\beta_t - \kappa_t \dot{\tilde{s}}_t\right) t_s \tag{1e}$$

$$v_{t+1} = v_t + a_t t_s \tag{1f}$$

$$l_{t+1} = l_t + \begin{cases} 1, & \text{if } \tilde{\eta}_t \geq w_{\text{lane}}/2 \\ -1, & \text{if } \tilde{\eta}_t < -w_{\text{lane}}/2 \\ 0, & \text{otherwise} \end{cases} \tag{1g}$$

where $\beta_t = \tan^{-1}((l_r/l_f + l_r)\tan(\gamma_t^{(\text{EV})}))$, and $\kappa(s)$ represents the *lane curvature* at s as sensed by a vision system. The EV's speed is denoted by v, and l_f and l_r are the lengths from the CoG to the front and rear axles, respectively. For compact notation, (1) will be rewritten as

$$\mathbf{x}_{t+1} = f_t(\mathbf{x}_t, \mathbf{u}_t). \tag{2}$$

B. Input Constraints

Actuator limits on the steering angle, steering rate, and acceleration are captured by constraints on the control inputs:

$$\mathbf{u}_{\min} \leq \mathbf{u}_t \leq \mathbf{u}_{\max}, \quad \dot{\mathbf{u}}_{\min} \cdot t_s \leq \mathbf{u}_{t+1} - \mathbf{u}_t \leq \dot{\mathbf{u}}_{\max} \cdot t_s. \tag{3}$$

C. Road Boundaries

The road boundaries impose the following bounds on the lateral position of the EV:

$$\eta_{\text{lane,min}} + \Delta\eta_{\text{safe}} \leq \eta_t \leq \eta_{\text{lane,max}} - \Delta\eta_{\text{safe}}, \tag{4}$$

where $\Delta\eta_{\text{safe}}$ is a desired safety distance which also accounts for the EV's geometry and orientation. The upper bound is set when the vehicle travels on the leftmost lane of the road, while the lower bound when it travels on the rightmost lane.

III. Target Vehicle Predictions

The SCMPC algorithm requires predictions for all relevant TVs in the EV's environment, which are combined into *traffic scenarios*. The TV predictions comprise two parts: (a) the *maneuver prediction model* that identifies the type of maneuver that a TV is likely to perform (Section III-A), and (b) the *trajectory prediction model* that generates possible trajectories for the TV, based on the predicted maneuver (Sections III-B and III-C).

A. Maneuver Prediction Model

The learning-based approach of [21] is used to estimate the distribution over a TV's maneuvers. The human driver is modeled as a stochastic hybrid system, switching between $i = 1, 2, 3$ *control strategies*: lane keeping (LK), lane change left (LCL), and lane change right (LCR). A set of features z_t, representing the current traffic scene at time t, is considered. The dependencies among these variables are described by a Hidden Markov Model (HMM), whose parameters are learned from experimental data collected in a training phase.

The distribution over the control strategy i_t at time t, conditioned on the history of measurements $z_{1:t} = \{z_1, \ldots, z_t\}$, is denoted $\mathbb{P}[i_t = i \,|\, z_{1:t}]$ with $i \in \{\text{LK}, \text{LCL}, \text{LCR}\}$. It is recursively computed by inference on the HMM; for details see [21]. Then, the TV's trajectory is predicted separately for each control strategy.

B. Lane Keeping (LK) Strategy

LK is the maneuver performed by a driver most of the time. The resulting trajectory is approximated by a straight line with a constant orientation (relative to the lane centerline), traversed at a constant acceleration. The relative orientation of the path is modeled by a random variable $\gamma^{(\text{TV})}$.

The resulting state predictions during LK are given by

$$s_{\text{LK}}(t) = s(0) + x_{\text{long}}(t)\cos(\gamma^{(\text{TV})}), \tag{5a}$$

$$\eta_{\text{LK}}(t) = \eta(0) + x_{\text{long}}(t)\sin(\gamma^{(\text{TV})}), \tag{5b}$$

$$\phi_{\text{LK}}(t) = \gamma^{(\text{TV})}, \tag{5c}$$

$$v_{\text{LK}}(t) = v(0) + a^{(\text{TV})} t. \tag{5d}$$

Here, $x_{\text{long}} = v(0)\tau + 0.5a^{(\text{TV})}\tau^2$ is the vehicle's longitudinal distance traveled along the path, and $s(0)$, $\eta(0)$, $v(0)$ are the initial conditions for s, η, v at the beginning of the predictions. The values of $\gamma^{(\text{TV})}$ and $a^{(\text{TV})}$ are constant over the prediction horizon and drawn from the probability distribution functions (PDFs) identified in Section III-D.

C. Lane Changing (LC) Strategies

LC trajectories are influenced by many factors, such as the driver's style and the current traffic situation. The cases for a lane change to the left (LCL) and to the right (LCR) are considered symmetric and thus handled analogously. In this paper, a maneuver-based approach is used where the lane change path is modeled as a sigmoid function:

$$f(s; a, b) = \frac{b}{1 + \exp(-as)}. \tag{6}$$

Here $a = (4/b)\tan(\alpha)$, where α is the slope at the inflection point of the sigmoid, and b is the ordinate value of the function for $s \to \infty$. As the sigmoid function tends to 0 for $s \to -\infty$, a point must be selected which is considered as the beginning of the path. We choose this to be the point

at which the sigmoid function has a value of $\backsim 2\%$ of the final lateral position b. Hence, an auxiliary constant variable $k = 4$ is set so that the initial lateral offset c becomes

$$c(b) = \frac{b}{1+\exp(k)} = 0.018b. \tag{7}$$

A constant variable $d(a) = k/a$ shifts the origin of the longitudinal axis to the lane change starting point, so that the lower horizontal asymptote approximates the current lane center line. The modified formulation of the sigmoid function is

$$\tilde{f}(s;a,b) = \frac{b+c(b)}{1+\exp(-a(s-d(a)))} - c(b). \tag{8}$$

The goal of the prediction model is to recognize which trajectory is being taken by a vehicle based on its current and past states. Thus, it is necessary to determine the values of the parameters a and b to identify the path, as well as the longitudinal position of the vehicle with respect to the beginning of the maneuver at $s_{LC}(0)$. It is reasonable to expect a TV to change one lane at a time; hence b is assumed to be equal to the road's lane width, subject to some disturbance.

The first and second order derivatives of the vehicle's lateral position $\eta(\tau)$ with respect to its longitudinal position $s(\tau)$ are used to retrieve the values of a and $s_{LC}(0)$:

$$v_{\text{lat}}^{(\text{TV})} \triangleq \frac{d\eta(\tau)}{ds(\tau)} = \frac{\frac{d\eta(\tau)}{d\tau}}{\frac{ds(\tau)}{d\tau}} \text{ and } a_{\text{lat}}^{(\text{TV})} \triangleq \frac{d^2\eta(\tau)}{ds(\tau)^2} = \frac{\frac{d}{d\tau}\frac{d\eta(\tau)}{ds(\tau)}}{\frac{ds(\tau)}{d\tau}}.$$

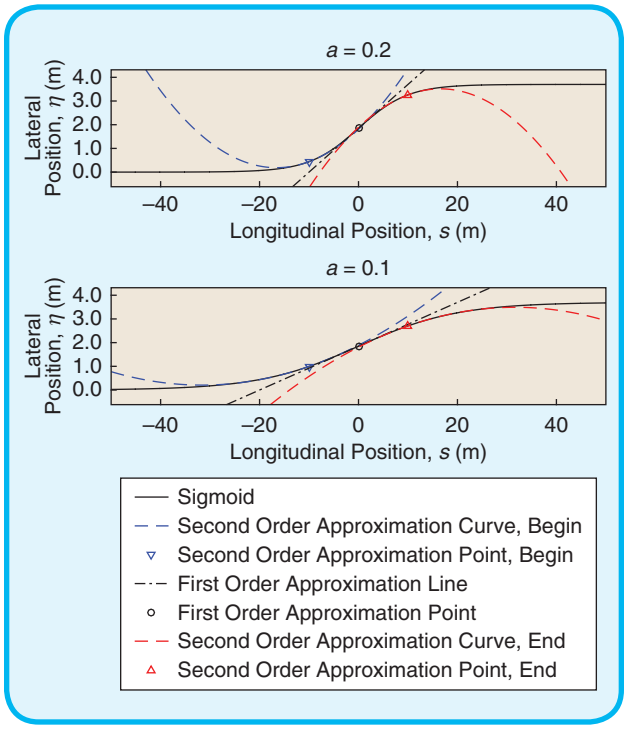

FIG 2 Approximation of different sigmoid functions.

Since it is not possible to find an explicit expression for a and s in the first and second derivatives of the sigmoid function $f(s)$, three Taylor series are considered to approximate it; see Fig. 2 for an illustration.

A first-order Taylor approximation $\phi_1(s;a,b,m_s)$ is considered at the inflection point by setting the approximation point $m_s = 0$. Its derivative

$$\phi_1^{(1)}(a;b) = \frac{d}{ds}\phi_1(s;a,b,m_s=0) = \frac{ba}{4} \tag{9}$$

allows the extraction of an explicit form of a. A second order Taylor approximation $\phi_2(s;a,b,m_s)$ has as first and second order derivatives $\phi_2^{(1)}(s;a,b,m_s)$ and $\phi_2^{(2)}(a;b,m_s)$, respectively, which do not allow us to find an explicit form of a. Therefore, $\phi_2^{(2)}$ is approximated with a second order Taylor series $\Psi_2(a;m_a,b,m_s)$, making it possible to determine the value of a. The approximation points can be considered as tuning parameters, and they are chosen to be: $m_{s1} = -10$ m, $m_{s2} = 10$ m and $m_a = 0.08$.

The longitudinal position of a vehicle performing a LC maneuver can be expressed in terms of the progression ratio, i.e., the ratio of the longitudinal position s with respect to the beginning of the maneuver and the length of the sigmoid function. A sigmoid length is defined as the value of s such that $\tilde{f}(s) = 0.95b$. The progression ratio function is derived by (8) and it is defined as

$$\Lambda(a,s) = \frac{sa}{2(k-\log(0.95))}. \tag{10}$$

Therefore, the sigmoid's parameters $\hat{a}$ and $\hat{s}$, corresponding to the estimated values of a and s, can be determined using the following algorithm:

if $a_{\text{lat}}^{(\text{TV})} \geq 0$ **then**
  $\hat{a} \leftarrow \Psi_2^{-1}(a_{\text{lat}}^{(\text{TV})}; m_a, b, m_{s1})$
  $\hat{s} \leftarrow \phi_2^{-1(1)}(v_{\text{lat}}; \hat{a}, b, m_{s1})$
  $\lambda \leftarrow \Lambda(\hat{a}, \hat{s})$
  if $\lambda > \lambda_{\text{threshold}}$ **then**
    $\hat{a} \leftarrow \phi_1^{-1(1)}(v_{\text{lat}}, b)$
    $\hat{s} \leftarrow 0$
  end if
else
  $\hat{a} \leftarrow \Psi_2^{-1}(a_{\text{lat}}^{(\text{TV})}; m_a, b, m_{s2})$
  $\hat{s} \leftarrow \phi_2^{-1}(v_{\text{lat}}; \hat{a}, b, m_{s2})$
  $\lambda \leftarrow \Lambda(\hat{a}, \hat{s})$
  if $\lambda < \lambda_{\text{threshold}}$ **then**
    $\hat{a} \leftarrow \phi_1^{-1(1)}(v_{\text{lat}}, b)$
    $\hat{s} \leftarrow 0$
  end if
end if

In the above algorithm, the first order approximation is adopted when the value of $\hat{s}$ calculated using the second order approximation is closer to the inflection point than a threshold given by the tuning variable $\lambda_{\text{threshold}}$.

The path is assumed to be tracked at a speed affected by a constant, random acceleration, $a^{(\text{TV})}$:

$$\tilde{s}(\tau) = \int_0^{\tau} v_{\text{LC}}(\tilde{\tau}) \cos(\phi_{\text{LC}}(\tilde{\tau}))\, d\tilde{\tau}, \tag{11}$$

$$s_{\text{LC}}(\tau) = s(0) - \hat{s} + \tilde{s}(\tau), \tag{12}$$

$$\eta_{\text{LC}}(\tau) = \eta(0) - \tilde{f}(\hat{s}) + \tilde{f}(\tilde{s}(\tau); \hat{a}, b), \tag{13}$$

$$\phi_{\text{LC}}(\tau) = \tan^{-1}(\tilde{f}^{(1)}(\tilde{s}(\tau); \hat{a}, b)), \tag{14}$$

$$v_{\text{LC}}(\tau) = v(0) + a^{(\text{TV})}\tau. \tag{15}$$

Here $\tilde{s}$ represents the maneuver progression and $s(0)$, $\eta(0)$ and $v(0)$ are the measured values at prediction time $t = 0$. The LCR follows from a LCL maneuver by inverting the signs of $\tilde{f}(\tilde{s}(\tau); \hat{a}, b)$ and $\tilde{f}^{(1)}(\tilde{s}(\tau); \hat{a}, b)$. The values of b and $a^{(\text{TV})}$ are drawn from PDFs, as shown in the next section.

D. Traffic Model Identification

For the LK maneuver, the probability distributions of the vehicle orientation and the longitudinal acceleration are

$$\gamma^{(\text{TV})} \sim \mathcal{N}(\mu_{\gamma^{(\text{TV})}}, \sigma^2_{\gamma^{(\text{TV})}}), \tag{16}$$

$$a^{(\text{TV})}_{\text{LK}} \sim \mathcal{N}(\mu_{a^{(\text{TV}),\text{lk}}}, \sigma^2_{a^{(\text{TV}),\text{lk}}}), \tag{17}$$

where $\mu_{\gamma^{(\text{TV})}}$, $\mu_{a^{(\text{TV}),\text{lk}}}$ are the means and $\sigma^2_{\gamma^{(\text{TV})}}$, $\sigma^2_{a^{(\text{TV}),\text{lk}}}$ the variances of the corresponding normal distributions.

For LC maneuvers, the probability distributions of the lane change offset and the longitudinal acceleration are given by

$$b \sim \mathcal{N}(\mu_b, \sigma^2_b), \tag{18}$$

$$a^{(\text{TV})}_{\text{LC}} \sim \mathcal{N}(\mu_{a^{(\text{TV}),\text{lc}}}, \sigma^2_{a^{(\text{TV}),\text{lc}}}), \tag{19}$$

where μ_b, $\mu_{a^{(\text{TV}),\text{lc}}}$ are the means and $\sigma^2_b, \sigma^2_{a^{(\text{TV}),\text{lc}}}$ the variances of the corresponding normal distributions.

The LK and LC distribution parameters are identified from a large data set provided by the Next Generation Simulation (NGSIM) program [22]. The set of recorded trajectories is represented in Fig. 3. Every gray line corresponds to the path of a single vehicle. As no information about the lane widths is available, lane positions are estimated according to the assigned lane identifier. Changes in the lane identifier are represented by red dots in the Fig. 3.

After pre-processing the data, the PDFs were identified as follows.

1) Select a point on a given trajectory.
2) Compute the predicted trajectories for many combinations of $(\gamma^{(\text{TV})}, a^{(\text{TV})}_{\text{LK}})$ or $(b, a^{(\text{TV})}_{\text{LC}})$ depending on the type of maneuver over a time horizon of 2 s.

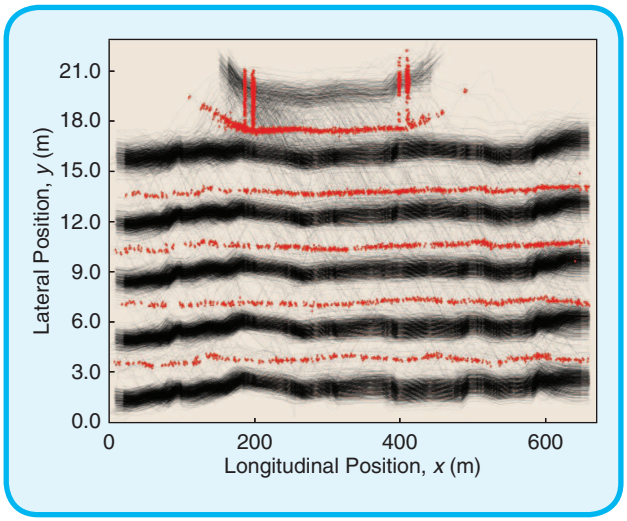

FIG 3 Raw traffic data and lane detection.

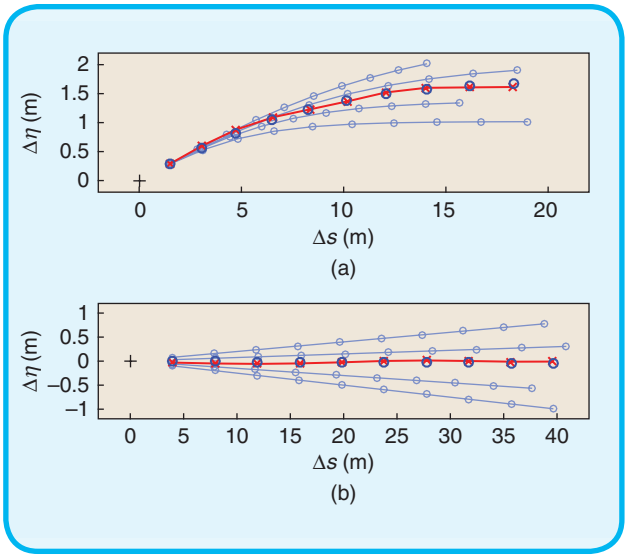

FIG 4 Lane change (a) and lane keeping (b) trajectories fitting: comparison between the real trajectory (red) and some calculated trajectories (blue) given the initial conditions (black cross). The best fitting trajectory has thicker circles.

3) Among all trajectories computed in Step 2, select the one with the lowest root-mean-square error (RMSE) with respect to the original trajectory.
4) Store the values of the parameters $(\gamma^{(\text{TV})}, a^{(\text{TV})}_{\text{LK}})$ or $(b, a^{(\text{TV})}_{\text{LC}})$ and the RMSE value corresponding to the trajectory chosen in Step 3.

Steps 2 and 3 are illustrated in Fig. 4. The outcome of the PDF identification and the corresponding numerical values are shown in Figs. 5 and 6. The average RMSE values for the predictions were 0.15 m ($v < 10$ m/s) and 0.17 m ($v \geq 10$ m/s) for the LK trajectories, and 0.21 m ($v < 10$ m/s) and 0.24 m ($v \geq 10$ m/s) for the LC trajectories.

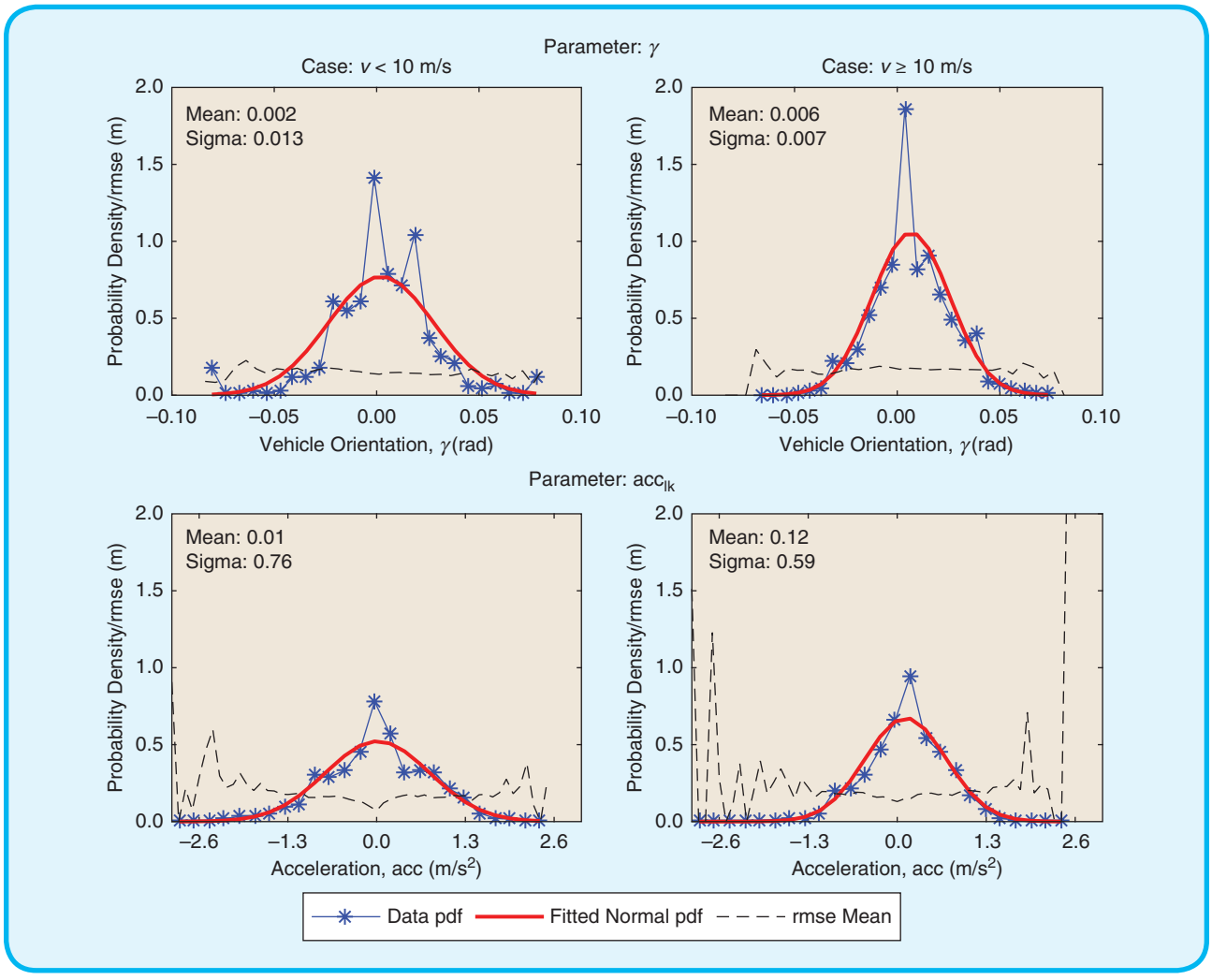

FIG 5 Lane keeping parameters PDFs fitting.

IV. Scenario Generation

Given the present *traffic scene*, SCMPC requires forecasts of *traffic scenarios* with associated uncertainty. Both the scenario generation based on TV predictions (Section III) and the controller design (see Sections V and VI) assume discretized time steps $t = 0, 1, 2, \ldots$, with a common sample time t_s. The SCMPC algorithm, however, can be based on any generic model-based or data-based method for scenario generation, if it fulfills the assumptions stated below. These scenarios can then be simplified or consolidated, as shown for our algorithm.

Let N be the time horizon for the traffic predictions. Denote all uncertainty involved in the traffic predictions over the horizon by the variable $\delta_t \in \Delta_t$, where the support Δ_t has an arbitrary nature and dimensions. The following technical assumption on the scenario generation is needed to derive the theoretical properties in Section VII.

Assumption 1 (Traffic Scenarios). *(a) At every time step t, an arbitrary number K of samples of the uncertainty $\delta_t \in \Delta_t$ over the horizon N ('scenarios') can be drawn from the traffic prediction model. (b) These sampled scenarios $\delta_t^{(1)}, \ldots, \delta_t^{(K)}$ are independent and identically distributed (i.i.d.) with the real uncertainty outcome δ_t.*

Assumption 1 requires only that sufficient scenarios are available and that they are representative of the actual uncertainty in the traffic scene. The availability of representative data is a minimal requirement for any uncertainty prediction.

Note that the scenario-based uncertainty model can be time-varying and hence be conditioned on the latest information available at each time step, unlike many other MPC approaches [23]. Neither the current joint probability distribution of δ_t nor the support Δ_t need to be known explicitly. These are further advantages over similar approaches using Stochastic MPC [12] or Robust MPC [24].

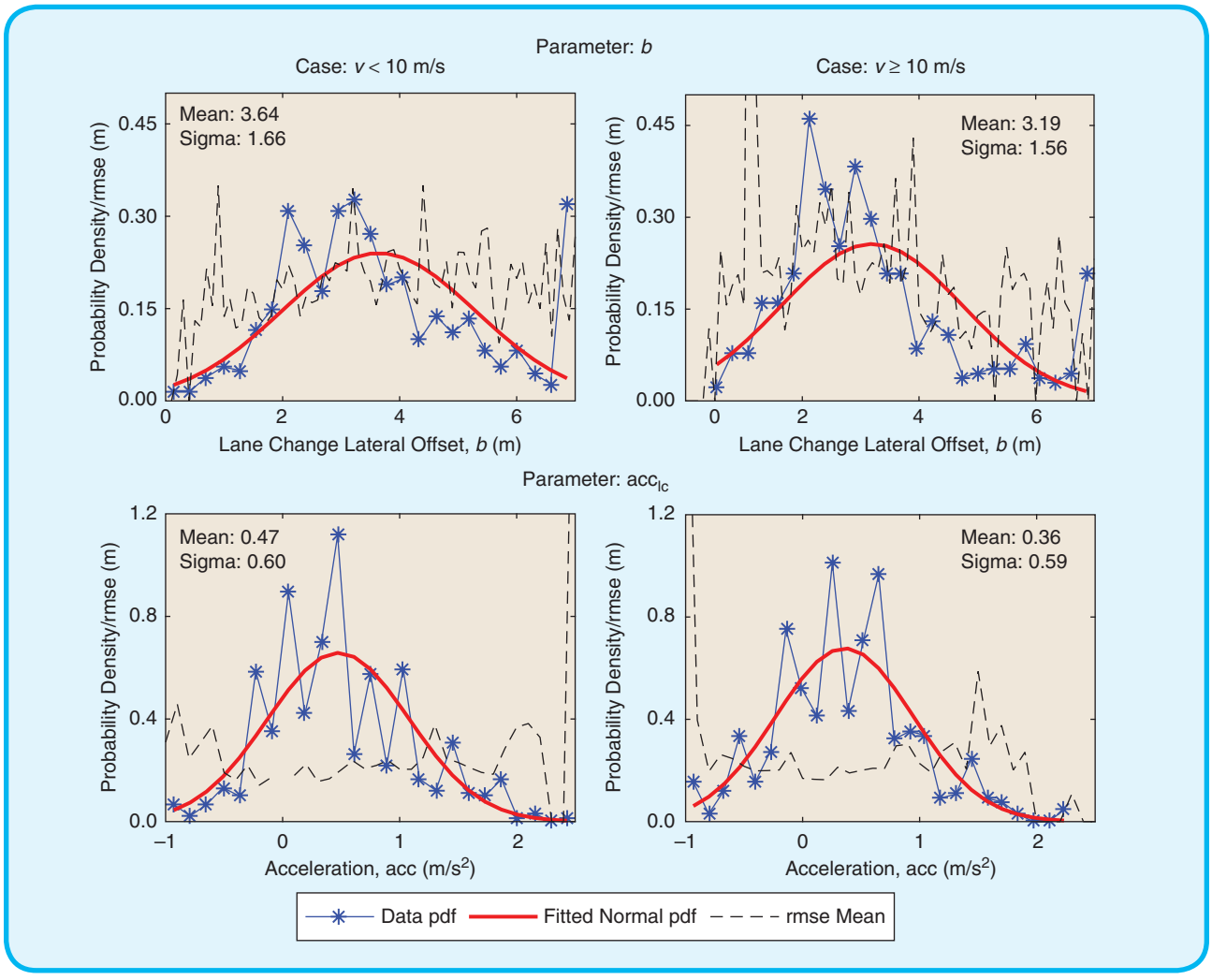

FIG 6 Lane changing parameters PDFs fitting.

A. Target Areas

Conditioned on the measured state of the TVs, scenarios for a TV j are obtained based on its estimated control strategy $\bar{\imath}_j \in \{\text{LCL}, \text{LCR}, \text{LK}\}$, as discussed in Section III-A. Then, for each strategy, a scenario number $K_{j,\text{LCL}}$, $K_{j,\text{LCR}}$, or $K_{j,\text{LK}}$ is identified:

$$K_{j,\text{LCL}} = \text{round}(\mathbb{P}[\bar{\imath}_j = \text{LCL}] \cdot K), \tag{20}$$

$$K_{j,\text{LCR}} = \text{round}(\mathbb{P}[\bar{\imath}_j = \text{LCR}] \cdot K), \tag{21}$$

$$K_{j,\text{LK}} = K - K_{\text{LCL}} - K_{\text{LCR}}. \tag{22}$$

Finally, a predicted trajectory is obtained for each scenario by feeding the sampled parameters into the corresponding prediction model (Sections III-B, III-C). The predicted positions are based on (5) for a LK and (11) for a LC maneuver, and they are estimated at the discrete sample times.

The range occupied by a TV is called *target area* and modeled by a rectangle. For each scenario of TV j and at each time step t, the vertices of the corresponding target area are associated with the set $\rho_{j,t}^{(k)}$, where $k = 1, \ldots, K$ is the scenario index. For each type of maneuver $m_j \in \{\text{LCL}, \text{LCR}, \text{LK}\}$, the sets $\rho_{j,t}^{(k)}$ are combined into the convex hull $R_{j,t}^{m_j}$ containing all the corresponding $\rho_{j,t}^{(k)}$, e.g., $R_{j,t}^{\text{LCL}}$ contains $K_{j,\text{LCL}}$ subsets $\rho_{j,t}^{(k)}$ associated to the LCL maneuver. Each convex hull includes a number of coordinates of vertices of the corresponding target areas and it is characterized by a centroid (C_x, C_y), or geometric center, which is used to reference the position of the whole set. An example of scenario-based predictions of a vehicle performing a lane change is shown in Fig. 7. The rectangular shapes representing the target areas are included inside the irregular polygons corresponding to the convex hulls $R_{j,t}^{m_j}$ considered within the prediction horizon. To reduce the computational complexity, the convex hull is substituted by the smallest rectangular box aligned

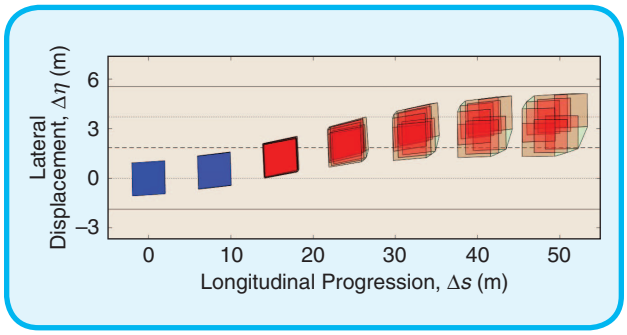

FIG 7 Scenario-based predictions of a LCL maneuver. The vehicle area (blue box while in the initial lane, red box when moving to the other lane) is drawn for each scenario at all predicted time steps. The green convex hull represents the target area. Δs and $\Delta\eta$ are the TV's longitudinal and lateral displacement with respect to its initial position.

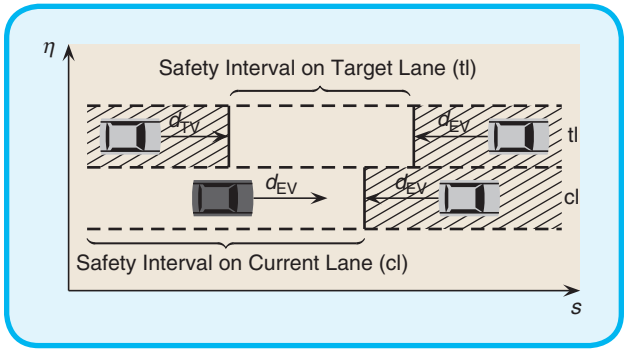

FIG 8 Construction of safety intervals $[s^{l,(k)}_{\min,t}, s^{l,(k)}_{\max,t}]$ at time step t for a particular scenario k. The EV is shown in dark gray and three TVs in light gray. The hatched regions on the current lane $l = \text{cl}$ and the target lane $l = \text{tl}$ are unsafe.

with the lane which contains the convex hull. This exemplification was chosen at the cost of introducing a larger uncertainty in TV prediction.

B. Safety Distances

Safety distances are included in the control algorithm by expanding the area occupied by each TV. First, each TV is embedded in a bigger safety box. Then, for each TV j, the vertices of the associated safety box substitute the vertices in the sets $\rho_j^{(k)}, k = 1,\ldots,K$. As a result, the corresponding convex hulls $\hat{R}_j^m, m \in \{\text{LCL, LCR, LK}\}$ become enlarged as well. For simplicity of notation, since the safety constraints are generated at each horizon step, the index t is dropped in the sequel.

Define $\mathcal{E}$ and $\mathcal{R}_j^m$ as the closed sets of the points inside the rectangle representing the EV's shape and the points inside the convex hull $\hat{R}_j^m$, respectively. Hence, the signed distance between the two sets is given by

$$\xi_j^m \triangleq \begin{cases} \min\limits_{a\in\varepsilon, b\in\mathcal{R}_j^m} d(a,b) & \text{if } \nexists b \in \varepsilon, a \in \mathcal{R}_j^m, \\ -\min\limits_{a\in\varepsilon, b\in\mathcal{R}_j^m} \{d(a,b) \,|\, a \in \mathcal{R}_j^m, b \in \varepsilon\} & \\ \qquad\qquad \text{if } \exists b \in \varepsilon, a \in \mathcal{R}_j^m. \end{cases} \tag{23}$$

where $d(a,b)$ is the Euclidean distance. When $\xi_j^m > 0$, it means that the EV is keeping the required safety distance from the scenario-based prediction of TV j.

The EV must strictly respect the safety distances under all sampled scenarios. Because of the finite sampling, there is a residual probability ε that the safety distances may indeed be violated. This probability will be bounded in Section VII.

V. Lane Change Assistance

The objective of SCMPC for lane change assistance is to detect and perform safe lane change trajectories on highways. A safe trajectory is considered to be one that respects all actuator, comfort, and safety constraints, as described in Section II. In order to simplify the optimization problem, the safety constraints to TVs, as described in Section IV-B, are simplified to comprise only longitudinal safety distances.

Assumption 2 (Safety Distances). *(a) If the EV stays on its current lane, it is only responsible to maintain a longitudinal safety distance d_{EV} to its front vehicle. (b) In a lane change situation, the EV needs to maintain a safety distance d_{EV} to the TVs in its front on the current lane and target lane, as well as a safety distance d_{TV} to the TV behind it on the target lane. (c) In closed-loop, the safety distances may occasionally be violated (e.g., if the TVs perform emergency braking maneuvers).*

Lateral constraints are also in place, confining the EV to its current lane or, in a lane change situation, to its current lane and target lane. Assumption 2 will translate into the chance constraints on the state of the EV [14], [25]. In a particular scenario $\delta_t^{(k)}$, the safety distances translate directly into state constraints on the longitudinal and lateral position of the EV.

If a lane change is desired, the algorithm requires it to be triggered by a human driver, e.g., via the turn signal or by a higher-level path planner, e.g., for autonomous driving. Hence the initiation of a lane change is given as an external signal to the SCMPC controller.

As illustrated in Fig. 8, the current lane (cl) and target lane (tl) are considered separately under each scenario. The safety distances on lane $l \in \{\text{cl,tl}\}$ under scenario $k \in \{1,\ldots,K\}$ are based on the trajectories of all TVs, as given by $\delta_t^{(k)}$.

TVs behind the EV are required to maintain a safety distance d_{TV}, depending on the corresponding TV's speed. TVs in the front require the EV to maintain a safety distance of d_{EV} that varies with the EV's speed. Hence the longitudinal position of the EV (s coordinate) is constrained by a *safety interval* $[s^{l,(k)}_{\min,t}, s^{l,(k)}_{\max,t}]$, as shown in Fig. 8. The lateral position of the EV (η coordinate) is constrained by the boundaries of the current lane ($l = \text{cl}$) and/or the target lane ($l = \text{tl}$), $[\eta^l_{\min,t}, \eta^l_{\max,t}]$.

A. Lane Change Timing

The SCMPC algorithm has a prediction horizon of N time steps, i.e., a total preview time of Nt_s. It consists of an *outer loop* and an *inner loop*. The outer loop iterates to find the

earliest time to initiate a lane change t_{lc} within the interval $[0, Nt_s]$. The inner loop attempts to compute a safe trajectory given the lane change initiation time t_{lc}; i.e., observing the lane change constraints under all K scenarios.

Assumption 3 (Lane Change Constraints). *(a) When integrated into the current (or target) lane, the EV must observe the safety interval on the current (or target) lane. (b) During a LC maneuver, the EV must respect also the safety interval on the target lane. (c) During a LC maneuver, the EV must be able to safely return to the current lane, so it must observe the safety intervals on both the current and target lanes.*

The EV may not be able to move into the safety interval on the target lane immediately, as in Fig. 8. Then it can first adjust its position and speed on the current lane before performing the lane change. It is assumed that an earlier lane change time is always preferred. The outer loop performs a simple search over a set of candidate lane change times

$$T_{lc} \subset \{0, t_s, \ldots, N{\cdot}t_s\}. \tag{24}$$

For each candidate lane change time $t_{\text{lc}} \in T_{\text{lc}}$, the inner loop computes the control inputs (steering, acceleration) and evaluates the feasibility of a lane change within t_{lc}; see Section V-C. Note that, by Assumption 3, if a lane change is feasible within t_{lc}, then it is also feasible within any candidate lane change time greater than t_{lc}. Hence a simple bisection algorithm can be used for the outer loop.

B. State Constraints

Let $[s^{l,(k)}_{\min,t}, s^{l,(k)}_{\max,t}]$ denote the safety interval on lane $l \in \{\text{cl}, \text{tl}\}$ at time $t \in \{1, \ldots, N\}$ under scenario $k \in \{1, \ldots, K\}$:

$$s^{l}_{\min,t} \triangleq \max_{k \in \{1,\ldots,K\}} s^{l,(k)}_{\min,t}, \tag{25a}$$

$$s^{l}_{\max,t} \triangleq \min_{k \in \{1,\ldots,K\}} s^{l,(k)}_{\max,t}. \tag{25b}$$

The *longitudinal limits* $s_{\min,t}, s_{\max,t}$ and the *lateral limits* $\eta_{\min,t}, \eta_{\max,t}$ are then defined as the limits of the current lane while $t \le t_{\text{lc}}$ and that of both lanes when $t > t_{\text{lc}}$. The resulting state constraints are

$$\mathbf{x}_{\min,t} \le \mathbf{x}_t \le \mathbf{x}_{\max,t}, \tag{26}$$

where $\mathbf{x}_{\min,t} \triangleq [s_{\min,t}\ \eta_{\min,t} - \infty\ 0\ l_t]^T$ and $\mathbf{x}_{\max,t} \triangleq [s_{\max,t} \eta_{\min,t} + \infty v_{\max}\ l_t]^T$, and $v_{\max}$ denotes the speed limit.

Remark 4 (Constraint Consolidation). *Given the sampled values of all scenarios* $\delta_t^{(1)}, \ldots, \delta_t^{(K)}$*, the safety intervals for each lane and hence the state constraints* (26) *can be readily computed for each time step t.*

C. Multi-Stage Scenario Program

For the *inner loop* of SCMPC, the lane change time t_{lc} is fixed. The control inputs $\mathbf{u}_t, \ldots, \mathbf{u}_{t+N-1}$ over the prediction horizon are then computed by solving the following Multi-Stage Scenario Program (MSSP):

$$\min_{\mathbf{u}_T,\, \mathbf{x}_{T+1}} \sum_{i \in T} (\mathbf{x}_{i+1} - \mathbf{x}_{\text{ref},i+1})^{\text{T}} \mathbf{Q} (\mathbf{x}_{i+1} - \mathbf{x}_{\text{ref},i+1}) + \mathbf{u}_i^T \mathbf{R} \mathbf{u}_i, \tag{27a}$$

$$\text{s.t.} \quad \mathbf{x}_{i+1} = f_i(\mathbf{x}_i, \mathbf{u}_i) \qquad \forall i \in T, \tag{27b}$$

$$\mathbf{u}_{\min}(\mathbf{x}_i) \le \mathbf{u}_i \le \mathbf{u}_{\max}(\mathbf{x}_i) \qquad \forall i \in T, \tag{27c}$$

$$\dot{\mathbf{u}}_{\min} \le \mathbf{u}_{i+1} - \mathbf{u}_i \le \dot{\mathbf{u}}_{\max} \qquad \forall i \in T, \tag{27d}$$

$$\mathbf{x}_{\min,i+1} \le \mathbf{x}_{i+1} \le \mathbf{x}_{\max,i+1} \qquad \forall i \in T. \tag{27e}$$

Here $T \triangleq \{t, \ldots, t+N-1\}$, and $\mathbf{Q} \in \mathbb{R}^{5\times5}, \mathbf{R} \in \mathbb{R}^{2\times2}$ are positive definite weighting matrices for tuning. The reference trajectory $\mathbf{x}_{\text{ref},t+1}, \ldots, \mathbf{x}_{\text{ref},t+N}$ for the EV is constructed according to the reference speed and the lane change time t_{lc}. Since the SCMPC chooses the final trajectory in a safe and dynamically feasible manner, the exact shape of the reference is secondary. Here it is chosen as a simple step change in the lateral coordinate.

The MSSP is a nonlinear program, which can be solved by standard algorithms [26], [27]. The objective function (27a) is a convex quadratic function that penalizes the deviation of the state from the reference as well as the input usage. The dynamic constraints (27b) are nonlinear. The input constraints (27c,d) can be linearized by substituting the initial state $x_i \equiv x_t$. The state constraints (27e) are implemented as soft constraints, to ensure feasibility of the program.

VI. Autonomous Highway Driving

A. State Constraints

In the case of autonomous driving, the state constraint generation is based on classifying the TVs into four categories. The categories depend on the longitudinal distance between the TV and the EV, and the lane in which the TV is driving.

Let Δs and $\Delta\eta$ be the longitudinal and lateral distance in the lane reference frame of a TV convex hull $\hat{R}$ centroid (C_s, C_η) and the EV. It follows that

$$\Delta s = C_s - s \quad \text{and} \quad \Delta\eta = C_\eta - \eta.$$

A TV convex hull $\hat{R}$ is defined to be on the same lane as its centroid. Thus, as shown in Fig. 9, a scenario-based prediction of a TV, denoted by $\hat{R}$, belongs to category T1, T2, T3 or T4 depending on: (i) the relative longitudinal distance Δs between the EV and $\hat{R}$ and (ii) the lane on which $\hat{R}$ lays. Since each category may include more than a TV convex hull $\hat{R}$, only that which is characterized by the smallest value of ξ is considered in the process of generating the constraint. We formulate the obstacle avoidance constraints as linear constraints on the state of the ego vehicle (similar to [28]). The constraint generation algorithm makes use of the longitudinal and lateral extremity values of the TV convex

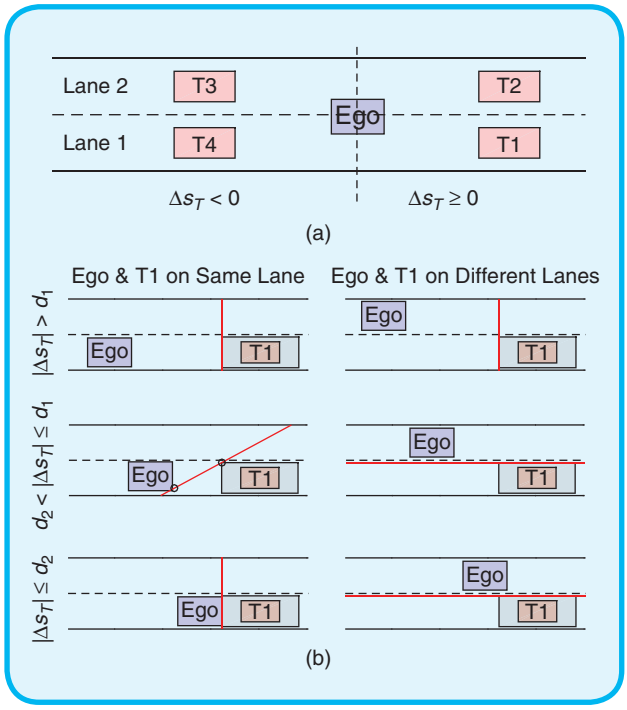

FIG 9 Target vehicles notation (a) and T1 constraint generation (b). In all figures the vehicles are assumed to drive towards the right direction.

hulls. In particular, the lateral maximum distances from the hull's centroid are denoted σ_{left} and σ_{right} while the front and the rear distances are denoted σ_{front} and σ_{rear}. Given a convex hull $\hat{R}$ in the lane reference frame with n vertices $(s_i, \eta_i), i \in \{1, \ldots, n\}$ including the safety boxes of a TV's scenario-based predictions, define

$$\sigma_{\text{right(left)}} \triangleq \min_{i \in \{1,\ldots,n\}} (\max) \{\eta_i - C_\eta \,|\, (s_i, \eta_i) \in \hat{R}\}, \tag{28}$$

$$\sigma_{\text{front(rear)}} \triangleq \max_{i \in \{1,\ldots,n\}} (\min) \{s_i - C_s \,|\, (s_i, \eta_i) \in \hat{R}\}. \tag{29}$$

The algorithm considers for every category three ranges of $|\Delta s|$ delimited by parameters d_1 and d_2 chosen so that $d_1 > d_2 > 0$. Therefore, three distinct rules are applied to generate the constraint depending on $|\Delta s_j|$.

- $|\Delta s_j| > d_1$: a linear constraint is set perpendicular to the lane direction, at the TV's closest vertex to the EV. Hence,

$$\begin{cases} s < C_s - \sigma_{\text{rear}}, & \text{for T1, T2,} \\ s > C_s + \sigma_{\text{front}}, & \text{for T3, T4.} \end{cases} \tag{30}$$

- $d_2 < |\Delta s_j| \leq d_1$: a linear constraint is built connecting a vertex of the EV to one of the TV are in order to obtain the steepest positive slope (for T1 and T3) or negative slope (for T2 and T4). In case of T1 and T3, the slope never decreases below 0; while for T2 and T4, it never increases above 0.
- $|\Delta s_j| \leq d_2$: if the EV and TV are on the same lane, a linear constraint is set perpendicular to the lane direction at the TV's closest vertex to the EV as in (30). Instead, if the EV and TV are on different lanes, a constraint parallel to the direction of march is applied to the TV's closest vertex to the EV. That is

$$\begin{cases} \eta < C_\eta + \sigma_{\text{left}}, & \text{for T1, T4,} \\ \eta > C_\eta - \sigma_{\text{right}}, & \text{for T2, T3.} \end{cases} \tag{31}$$

Fig. 9 shows an example of the rules used to generate the constraints for the T1 case. In this figure, the constraint is drawn as a red line, while the safety box is the gray surface centered on the corresponding TV. The constraints for the other cases (T2, T3 and T4) can be deducted by symmetrically applying the given rules.

Finally, since the safety constraints on the state are linear, they can be expressed at each time step t in the form $\mathbf{A}_{s,t}\mathbf{x}_t \leq \mathbf{b}_{s,t}$ where the subscript s underline the fact that these scenario-based constraints are generated as a result of a *stochastic process*.

B. Multi-Stage Scenario Program

The optimal control input $\mathbf{u}_i^*$, for $t \in T \triangleq \{t, \ldots, t+N-1\}$, is computed at every time step by solving the following Multi-Stage Scenario Program (MSSP):

$$\min_{\mathbf{u}_i} \Big(\sum_{i \in T} (\mathbf{x}_i - \mathbf{x}_{\text{ref},i})^T \mathbf{Q} (\mathbf{x}_i - \mathbf{x}_{\text{ref},i}) + \mathbf{u}_i^T \mathbf{R} \mathbf{u}_i + \Delta \mathbf{u}_i^T \mathbf{R}_R \Delta \mathbf{u}_i \Big) + (\mathbf{x}_N - \mathbf{x}_{\text{ref},N})^T \mathbf{Q}_N (\mathbf{x}_{t+N} - \mathbf{x}_{\text{ref},t+N}), \tag{32a}$$

$$\text{s.t.} \quad \mathbf{x}_{i+1} = f_i(\mathbf{x}_i, \mathbf{u}_i), \qquad \forall i \in T, \tag{32b}$$

$$\Delta \mathbf{u}_i = (\mathbf{u}_i - \mathbf{u}_{i-1})/t_s, \qquad \forall i \in T, \tag{32c}$$

$$\mathbf{A}_{d,i+1}\mathbf{x}_{i+1} \leq \mathbf{b}_{d,i+1}, \qquad \forall i \in T, \tag{32d}$$

$$\mathbf{A}_{s,i+1}\mathbf{x}_{i+1} \leq \mathbf{b}_{s,i+1}, \qquad \forall i \in T \tag{32e}$$

$$\mathbf{u}_{\min} \leq \mathbf{u}_i \leq \mathbf{u}_{\max}, \qquad \forall i \in T, \tag{32f}$$

$$\Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_i \leq \Delta \mathbf{u}_{\max}, \qquad \forall i \in T. \tag{32g}$$

Here $\mathbf{Q} \in \mathbb{R}^{5\times5}$, $\mathbf{R} \in \mathbb{R}^{2\times2}$, $\mathbf{R}_r \in \mathbb{R}^{2\times2}$ and $\mathbf{Q}_N \in \mathbb{R}^{5\times5}$ are positive definite weighting matrices, which can be used for tuning. The matrices $\mathbf{Q}$ and $\mathbf{Q}_N$ penalize the deviation of the state from the reference trajectory, while $\mathbf{R}$ and $\mathbf{R}_R$ respectively weight the magnitude and the rate of the inputs. The system prediction (32b) is initialized with the state $\mathbf{x}_t$ of the EV, measured at time t. The polyhedron defined by the deterministic constraint is described by $\mathbf{A}_{d,i+1}$ and $\mathbf{b}_{d,i+1}$, while the scenario-based constraints are expressed by $\mathbf{A}_{s,i+1}$ and $\mathbf{b}_{s,i+1}$.

VII. Bound On Safety Distance Violations

So far, no indication has been made for the selection of the sample size K. In this section, the framework of

scenario-based optimization [15]–[17] is used to establish a connection between K and the satisfaction of chance constraints, in the sense defined below [25].

Definition 5 (Chance Constraints). *(a) A chance constraint is defined by the frequency of safety distance violations V_t between EV and TVs over time $t = 0, 1, \ldots$. In particular, the SCMPC controller must guarantee that the safety distance of each EV and TV is not violated in more than a fraction $\varepsilon \in (0, 1)$ of all time steps $t = 0, 1, \ldots$.*

Here $\varepsilon \in (0, 1)$ is a tuning parameter that represents the *conservatism* of the driving style. A small ε corresponds to a low-risk driving style, a large ε makes the controller perform more aggressive maneuvers. The following results are based on the SCMPC theory in [14], to which the reader is referred for further details.

Let $\mathbb{P}[V_t]$ denote the probability of a safety constraint violation at time t. It is hard to bound $\mathbb{P}[V_t]$ directly, because the MSSPs in this paper are not convex, as required by the theory presented in [14]. However, for each case it can be shown that the *support constraints* contributed to the MSSP through the traffic scenarios can be upper bounded, even though the optimization problem is not convex. The argument in both cases is essentially the same: for the case of lane change assistance, all but one constraints are completely redundant; and for the case of autonomous highway driving, all but two constraints are redundant. The following technical assumption is required for the theoretical analysis of the algorithms presented in Sections V and VI.

Assumption 6 (Feasibility). *(a) There always exists a feasible solution to the MSSPs* (27) *and* (32). *(b) For the lane change assistance algorithm, the outer loop iterations are replaced by a fixed lane change time t_{lc}; e.g., the highest value in T_{lc}.*

A. Lane Change Assistance

Let the lane $l \in \{\text{cl}, \text{tl}\}$ and time step $t \in \{0, 1, \ldots\}$ be fixed and consider the front safety distance $i \in \{1, \ldots, N\}$ steps into the future (the back safety distance works analogously). The violation of the front safety distance can occur only if $s^l_{\max,t+i}$ has been chosen higher than

$$\bar{s}^l_{\max,t+i}(\delta_t), \tag{33}$$

which is the unknown true longitudinal position for the EV under the unknown true scenario δ_t:

$$\mathbb{P}[V_t] \leq \mathbb{P}[\bar{s}^l_{\max,t+i}(\delta_t) < s^l_{\max,t+i}]. \tag{34}$$

Therefore, to upper bound $\mathbb{P}[V_t]$ by ε, it suffices to upper bound the right-hand side of (34). This can be accomplished by a basic sampling lemma [29].

Theorem 7 (Violation Bound). *Let Assumptions 1, 2, 3, 6 hold and (27b) be an exact model of the controlled vehicle. Then*

$$\mathbb{P}[\bar{s}^l_{\max,t+i}(\delta_t) < s^l_{\max,t+i}] \leq \frac{1}{K+1}, \tag{35}$$

and hence the sample size $K \geq (1/\varepsilon) - 1$ ensures satisfaction of the chance constraints according to Definition 5.

Proof: All scenarios $\delta_t^{(1)}, \ldots, \delta_t^{(K)}$ are independent and identically distributed with δ_t (Assumption 1), so each one is equally likely to generate the tightest bound for the safety distance [29], and hence (35). State constraint satisfaction follows by Assumption 6 along the lines of [14]. ■

B. Autonomous Highway Driving

The argument now follows a similar pattern to the previous case. By construction, it holds true that if the safety distance is violated, i.e., the signed distance $\xi^m_{j,k}$ is smaller than 0, then the scenario-based constraints are violated, i.e.,

$$\mathbf{A}_{s,t}(\delta_t)\tilde{\mathbf{x}}_t > \mathbf{b}_{s,t+1}(\delta_t). \tag{36}$$

Note that the converse is not true. However, it follows that

$$\begin{aligned}\mathbb{P}[V_t] &= \mathbb{P}[\xi^m_{j,t} \leq 0, \forall j, m = \text{LCL}, \text{LCR}, \text{LK}] \\ &\leq \mathbb{P}[\mathbf{A}_{s,t}(\delta_t)\tilde{\mathbf{x}}_t > \mathbf{b}_{s,t+1}(\delta_t)].\end{aligned} \tag{37}$$

Hence, in order to upper bound the safety distance violations, it suffices to upper bound the right-hand side of (37).

At a fixed time t, the front and lateral safety distances are considered. While performing lane keeping, the EV is only responsible for maintaining a safety distance to the front TV (T1 or T2) and the lateral TV (the most restrictive between T1 and T4, or T2 and T3). During a LC maneuver, the EV must also respect the safety distance required by the oncoming vehicles (T3 or T4). In both cases, by the geometric constructions described in Section VI-A, there can be at most two support constraints.

Theorem 8 (Violation Bound). *Let Assumptions 1, 2, 6 hold and (32b) be an exact model of the controlled vehicle. Then*

$$\mathbb{P}[\mathbf{A}_{s,t}(\delta_t)\tilde{\mathbf{x}}_t > \mathbf{b}_{s,t+1}(\delta_t)] \leq \frac{2}{K+1}, \tag{38}$$

and hence the sample size $K \geq (2/\varepsilon) - 1$ ensures satisfaction of the chance constraints according to Definition 5.

VIII. Experimental Results

In this section, we demonstrate the ability of the SCMPC algorithm for autonomous highway driving, as described in Section VI, to operate in real-time on our experimental vehicle in the presence of virtual TVs. An Oxford Technical Solutions (OxTS) RT 2002 sensing system, comprising of a global positioning system (GPS) and an inertial measurement unit (IMU), combined with a GPS base station are used for localizing the vehicle. Computations related to the SCMPC algorithm are performed on a Speedgoat real-time mobile target machine (2.16 GHz Intel Core 2 Duo

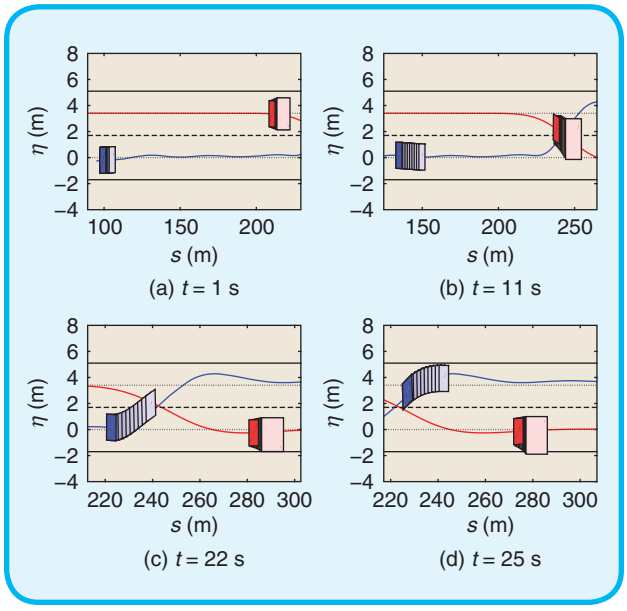

FIG 10 Snapshots of the traffic scene during the experiment. The EV (blue) and a TV (red) are represented with their corresponding open-loop predictions (light blue and red) and closed-loop trajectories (solid lines).

processor, 1 GB RAM) running a Simulink Real-Time application. The sensors, actuators and computing platform on the vehicle communicate via a CAN bus.

The MSSP (32) is solved in real-time using the nonlinear optimization package NPSOL [30]. A prediction horizon of $N = 10$ is used which results in $2 \times N = 20$ optimization variables. There are N_{TV} number of state constraints per time step of the horizon yielding a total of $N \cdot N_{\text{TV}}$ constraints. The scenario-based traffic prediction and controller run at a sampling time of 0.2 s which is the maximum computation time of the proposed approach on our platform.

The traffic scene for the experiments consists of one TV initially in the left lane with the EV in the right lane. For safety reasons, the reference speed for the EV is 8 m/s. A virtual (simulated) TV is used which tracks a reference speed of 4 m/s. At $t = 0$ s, the vehicles are at rest and the TV is 100 m ahead of the EV. The TV begins a lane change to cut in front of the EV at approximately $t = 4$ s.

Snapshots of the traffic scene during the experiment at four time instants are shown in Fig. 10. The solid blue and red rectangles depict the EV and TV, respectively, at the given time instants. The shaded blue rectangles show the predicted open-loop trajectory of the EV as computed by the SCMPC algorithm. The shaded red rectangles are the convex hulls of the TV areas over the prediction horizon. The solid blue and red lines are the closed-loop trajectories of the EV and TV, respectively.

At $t = 1$ s (Fig. 10a), the EV and TV are proceeding in the same direction on different lanes. At $t = 11$ s (Fig. 10b), the TV is in the process of changing lane. It is seen that the actual trajectory of the TV lies within the scenario based predictions of the TV's future motion. Also, the speed difference of the two vehicles is making the EV approach the TV ahead. However, the EV is still far away from the TV, hence, it plans to stay in its lane. At $t = 22$ s (Fig. 10c), the TV is classified as a T1 vehicle and the relative distance between the TV and the EV is between the thresholds d_1 and d_2 (see Section VI-A). The corresponding safety constraint causes the EV to plan a collision avoidance maneuver and change lanes to the left. Fig. 10d shows the traffic scene at $t = 25$ s while the EV is changing lanes, gaining the ability to overtake the slower TV on the left.

IX. Conclusion

This paper has presented algorithms to perform safe lane changes on highways. It has been shown how scenario-based traffic prediction models can be combined with the latest developments in scenario MPC. A maneuver-based model to rapidly predict lane keeping and lane changing trajectories of target vehicles has been developed and validated using real traffic data from a large data set. The real-time capability of the proposed approach has been demonstrated via experiments on our test vehicle.

Future research will focus on extensive field tests on highways in real traffic. The sensors on our experimental vehicle will be used to detect surrounding target vehicles. Furthermore, it will involve a more detailed study and modeling of traffic scenarios, based on actual traffic data, to reduce the uncertainty affecting the probabilistic distributions from which scenarios are sampled.

Acknowledgments

This material is based upon work partially supported by the National Science Foundation under Grant No. 1239323. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. The work was also supported by the Hyundai Center of Excellence at UC Berkeley. Gianluca Cesari was funded by the Master Thesis Grant issued by the Zeno Karl Schindler Foundation.

About the Authors

Gianluca Cesari received his 'Laurea' degree in Automation Engineering (2012) from the University of Bologna, Italy, and B.Eng. degree in Electrical Engineering (2013) from the Tongji University, Shanghai, China. In 2016, he completed his M.S. in Robotics, Systems and Control at ETH Zurich, Switzerland. During his education, he has worked more than one year in the field of industrial automation. Now, his area of interest and specialization is optimization-based predictive control applied to autonomous mobile robots.

Georg Schildbach obtained his Masters degrees in Applied Mechanics (Dipl.-Ing.) and Industrial Engineering (Dipl. Wirtsch.-Ing.) from the Technical University of Darmstadt in 2008. After working in the financial industry for two years, he joined the Automatic Control Laboratory at ETH Zurich, where obtained his Ph.D. degree in the field of control and optimization in 2014. He held the position of an Associate Director at the Hyundai of Excellence at UC Berkeley until 2016, and now works for Elektronische Fahrwerksysteme GmbH in the area of vehicle control and automated driving.

Ashwin Carvalho received the B.Tech. degree in Mechanical Engineering from the Indian Institute of Technology Bombay in 2011. He is currently pursuing the Ph.D. degree in Mechanical Engineering (Controls) at the University of California, Berkeley, with minors in Artificial Intelligence and Optimization. His research interests lie in stochastic model predictive control, environment modeling and real-time optimization for control under uncertainty.

Francesco Borrelli received the 'Laurea' degree in computer science engineering in 1998 from the University of Naples 'Federico II', Italy. In 2002 he received the PhD from the Automatic Control Laboratory at ETH-Zurich, Switzerland. He is currently a Professor at the Department of Mechanical Engineering of the University of California at Berkeley, USA.

He is the author of more than one hundred publications in the field of predictive control. He is author of the book Constrained Optimal Control of Linear and Hybrid Systems published by Springer Verlag, the winner of the 2009 NSF CAREER Award and the winner of the 2012 IEEE Control System Technology Award.

In 2008 he was appointed the chair of the IEEE technical committee on automotive control. His research interests include constrained optimal control, model predictive control and its application to advanced automotive control and energy efficient building operation.

References

[1] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *Robomech J.*, vol. 1, no. 1, pp. 1–14, 2014.

[2] J. Jansson, "Collision avoidance theory with application to automotive collision mitigation," Ph.D. Dissertation, Linköping University, Linköping, Sweden, 2005.

[3] M. Althoff, D. Heß, and F. Gambert, "Road occupancy prediction of traffic participants," in *Proc. 16th Int. IEEE Conf. Intelligent Transportation Systems*, 2013, pp. 99–105.

[4] M. Althoff and J. M. Dolan, "Online verification of automated road vehicles using reachability analysis," In *IEEE Trans. Robot.*, vol. 30, pp. 903–918, 2014.

[5] C. Laugier, I. E. Paromtchik, M. Perrollatz, M. Yong, J.-D. Yoder, C. Tay, K. Mekhnacha, and A. Négre, "Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety," *IEEE Intell. Transport. Syst. Mag.*, vol. 2, no. 4, pp. 4–19, 2011.

[6] J. Wiest, M. Höffken, U. Kreßel, and K. Dietmayer, "Probabilistic trajectory prediction using Gaussian mixture models," in *Proc. Intelligent Vehicles Symp.*, Alcalá de Henares, Spain, 2012.

[7] A. Broadhurst, S. Baker, and T. Kanade, "Monte Carlo road safety reasoning," in *Proc. Intelligent Vehicles Symp.*, Las Vegas (NV), United States, 2005.

[8] A. Eidehall and L. Petersson, "Statistical threat assessment for general road scenes using Monte Carlo sampling," *IEEE Intell. Transport. Syst. Mag.*, vol. 9, no. 1, pp. 137–147, 2008.

[9] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Madison (WI): Nob Hill Publishing, 2009.

[10] A. Carvalho, S. Lefèvre, G. Schildbach, J. Kong, and F. Borrelli, "Automated driving: The role of forecasts and uncertainty: A control perspective," *Eur. J. Control*, vol. 24, pp. 14– 32, 2015.

[11] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Autonomous driving using model predictive control and a kinematic bicycle vehicle model," in *Proc. Intelligent Vehicles Symp.*, Seoul, Korea, 2015.

[12] A. Carvalho, Y. Gao, S. Lefèvre, and F. Borrelli, "Stochastic predictive control of autonomous vehicles in uncertain environments," in *Proc. 12th Int. Symp. Advanced Vehicle Control*, Tokyo, Japan, 2014.

[13] L. Blackmore, M. Ono, A. Bektassov, and B. C. Williams, "A probabilistic particle-control approximation of chance-constrained stochastic predictive control," *IEEE Trans. Robot.*, vol. 26, no. 3, pp. 502–516, 2010.

[14] G. Schildbach, L. Fagiano, C. Frei, and M. Morari, "The scenario approach for stochastic model predictive control with bounds on closed-loop constraint violations," *Automatica*, vol. 50, no. 12, pp. 3009–3018, 2014.

[15] G. Schildbach, L. Fagiano, and M. Morari, "Randomized solutions to convex programs with multiple chance constraints," *SIAM J. Optim.*, vol. 23, no. 4, pp. 2479–2501, 2013.

[16] M. C. Campi and S. Garatti, "The exact feasibility of randomized solutions of uncertain convex programs," *SIAM J. Optim.*, vol. 19, pp. 1211–1230, 2008.

[17] M. C. Campi and S. Garatti, "A sampling and discarding approach to chance-constrained optimization: Feasibility and optimality," *J. Optim. Theory. Appl.*, vol. 148, pp. 257–280, 2011.

[18] R. Rajamani, Ed., *Vehicle Dynamics and Control*. 2nd ed. New York: Springer-Verlag, 2012.

[19] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a Frenét frame," in *Proc. IEEE Conf. Robotics and Automation*, Anchorage (AK), United States, 2010.

[20] C. M. Kang, S.-H. Lee, and C. C. Chung, "Comparative evaluation of dynamic and kinematic vehicle models," in *Proc. 53rd Conf. Decision and Control*, Los Angeles (CA), United States, 2014.

[21] S. Lefèvre, Y. Gao, D. Vasquez, E. Tseng, R. Bajcsy, and F. Borrelli, "Lane keeping assistance with learning-based driver model and model predictive control," in *Proc. 12th Int. Symp. Advanced Vehicle Control*, 2014.

[22] U.S. Department of Transportation. (Jan. 2007). Next generation simulation program, U.S Highway 101 data set Avaiable: http://ngsim.fhwa.dot.gov

[23] G. Schildbach, "Scenario-based optimization for multi-stage stochastic decision problems," Ph.D. Dissertation, Eidgenössische Technische Hochschule Zürich, Zürich, Switzerland, 2014.

[24] Y. Gao, A. Gray, H. E. Tseng, and F. Borrelli, "A tube-based robust nonlinear predictive control approach to semiautonomous ground vehicles," *Vehicle Syst. Dyn.*, vol. 52, no. 6, pp. 802–823, 2014.

[25] G. Schildbach and F. Borrelli, "Scenario model predictive control for lane change assistance on highways," in *Proc. IEEE Intelligent Vehicles Symp.*, Seoul, South Korea, 2015, pp. 611–616.

[26] D. Luenberger and Y. Ye, *Linear and Nonlinear Programming*, 3rd ed. Berlin: Springer-Verlag, 2008.

[27] J. Nocedal and S. Wright, *Numerical Optimization*, 2nd ed. New York: Springer-Verlag, 2006.

[28] J. Nilsson, Y. Gao, A. Carvalho, and F. Borrelli, "Manoeuvre generation and control for automated highway driving," in *Proc. 19th world Congress of the Int. Federation of Automatic Control*, 2014.

[29] G. C. Calafiore and M. C. Campi, "Uncertain convex programs: Randomized solutions and confidence levels," *Math. Prog. Series A*, vol. 102-1, pp. 25–46, 2005.

[30] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, "User's guide for NPSOL: Version 4.0," 1986.

ITS