2022 American Control Conference (ACC)
Atlanta, USA, June 8-10, 2022

Multi-stage Perception-aware Chance-constrained MPC with Applications to Automated Driving

Angelo D. Bonzanini, Ali Mesbah, and Stefano Di Cairano

***Abstract*— Perception-aware Chance-constrained Model Predictive Control (PAC-MPC) accounts for the interdependence between perception and control for systems operating in uncertain environments. The environment is discovered by perception, which imposes chance constraints on system operation. PAC-MPC can handle a perception quality that depends on the system states and/or inputs, thus affecting uncertainty quantification in the chance constraints. In this paper, we extend PAC-MPC by introducing a scenario-based prediction for future measurements, so that the resulting multi-stage PAC-MPC does not require a conservative estimate of the measurement prediction error covariance. We demonstrate PAC-MPC for automated vehicle control when obstacles and road boundaries are uncertain and perceived by variable precision sensors subject to an overall sensing budget and when the scenarios are generated based on possible obstacle behaviors.**

I. Introduction

A major challenge in automated vehicle operation [1] is to make safe and effective decisions in environments that are not perfectly known and are rapidly changing. Typically, information about the environment is acquired by perception algorithms operating on data from multiple sensors [2], [3]. Thus far, the proposed control architectures execute perception and control sequentially, i.e., they first perceive the environment and then decide the actions that the vehicle will take. However, this ignores the two-sided coupling between perception and control.

The acquired knowledge of the environment (or the lack thereof) influences the decision of a vehicle to operate in (or avoid) specific areas. But at the same time, acquiring this environment knowledge depends on how the vehicle moves (e.g., due to the field of view or focus of sensors) and on how the sensor and the sensor processing are controlled. As such, there exists an interdependence between perception and control: the perception of the environment depends on how the vehicle is controlled, and the control of the vehicle depends on the environment information acquired.

Yet, the two-way interaction between perception and control remains under-explored. In our recent works [4], [5], we proposed and analyzed the properties of a MPC design that addresses the interaction of perception of an uncertain environment and control of a known system, which are coupled through constraints. Due to the stochastic nature of the environment, the state constraints are formulated as chance constraints [6]–[8]. Our MPC approach propagates the first and second moments of the environment state distribution by taking into account the future effects of the environment estimator, which, in turn, depends on the control decisions. Due to accounting for the interaction between perception and control, we named the approach *Perception Aware Chance-constrained* MPC (PAC-MPC).

In this paper, we further extend the PAC-MPC framework with a scenario-based prediction to consider multiple realizations of the future measurements of the environment state. This scenario-based approach enables closed-loop prediction of the different environment measurements, as different control inputs would be available for the different scenarios. Thus, the proposed approach avoids a conservative robustification that would account for the measurement prediction error. We demonstrate the multi-stage PAC-MPC (msPAC-MPC) for automated vehicle control, where the scenario approach is particularly useful for predicting different (discrete) future behaviors of other vehicles, e.g., keeping lane, changing lane, or staying out of the road.

The paper is organized as follows. Section II presents the problem setting, followed by the scenario-based formulation for PAC-MPC in Section III. Section IV discusses the implementation of the msPAC-MPC on an automated vehicle case study. In Section V, we show various simulation scenarios where the lanes and other vehicles are uncertain and determined through perception, comparing msPAC-MPC and PAC-MPC. Finally, Section VI concludes the paper.

Notation: $\mathbb{R}$, $\mathbb{R}_{0+}$, $\mathbb{R}_{+}$, $\mathbb{Z}$, $\mathbb{Z}_{0+}$, and $\mathbb{Z}_{+}$ are the sets of real, nonnegative real, positive real, integer, nonnegative integer, and positive integer numbers, respectively. Intervals are denoted by $\mathbb{Z}_{[a,b)} = \{z \in \mathbb{Z} : a \leq z < b\}$. For a vector x, the i-th component is denoted by $[x]_i$. $\|\cdot\|$ is the 2-norm, and for a positive (semi) definite matrix, $Q > 0$, ($Q \geq 0$), $\|x\|_Q^2 = x^\top Q x$. $X = \mathrm{diag}[x]$ is a diagonal matrix such that $[X]_{ii} = x_i$. $\mathbb{P}[A]$ is the probability of event A. A normally distributed random vector x with mean μ^x and covariance Σ^x is denoted by $x \sim \mathcal{N}(\mu^x, \Sigma^x)$.

II. Preliminaries on PAC-MPC

In this section, we describe the individual components of the PAC-MPC and formulate the optimal control problem.

A. Vehicle Motion and Uncertain Environment Models

The vehicle motion model is described by the *known* discrete-time nonlinear dynamics

$$x_{k+1}^s = f^s(x_k^s, u_k^s), \tag{1a}$$

$$y_k^s = q^s(x_k^s, u_k^s), \tag{1b}$$

A. D. Bonzanini and A. Mesbah are with the Department of Chemical and Biomolecular Engineering at the University of California, Berkeley, CA. {adbonzanini,mesbah}@berkeley.edu

S. Di Cairano is with Mitsubishi Electric Research Laboratories, Cambridge, MA. dicairano@ieee.org

A.D. Bonzanini was an intern at MERL during this work.

where $x^s \in \mathbb{R}^{n_x}$ is the state vector, $u^s \in \mathbb{R}^{n_u}$ is the input vector, and $y^s \in \mathbb{R}^{n_y}$ is the performance output vector. Model (1) is subject to state and input constraints

$$x^s \in \mathcal{X}, \quad u^s \in \mathcal{U}, \tag{2}$$

where $\mathcal{X}$ and $\mathcal{U}$ are the state and input constraint sets, respectively. Although (1) is considered to be known, model uncertainty can be accounted for using standard methods [9]. The vehicle modeled by (1) operates in an uncertain environment, which is modeled by a vector of random variables $x^e \in \mathbb{R}^{m_x}$, e.g., the road boundaries/features and obstacle positions. The environment dynamics and measurements are modeled as

$$x^e_{k+1} = f^e(x^e_k, \psi_k), \tag{3a}$$
$$y^e_k = q^e(x^e_k, \zeta_k, x^s_k, u^s_k), \tag{3b}$$

where $y^e \in \mathbb{R}^{m_y}$ is the vector of environment measurements, and ψ, ζ are random process and measurement noise, respectively. While (3a) does not depend on (1), the environment measurements (3b) depend on the system states and inputs. This allows accounting for the dependence of sensing on states as well as on control actions.

B. Estimator and Uncertainty Propagation

The propagation of the first and second moments of the environment state x^e is based on a model of the environment in closed-loop with the perception algorithm

$$\mu^e_{k+1} = g_\mu\left(\mu^e_k, y^e_k, x^s_k, u^s_k\right), \tag{4a}$$
$$\Sigma^e_{k+1} = g_\Sigma\left(\Sigma^e_k, x^s_k, u^s_k\right), \tag{4b}$$

where μ^e and Σ^e are the mean and covariance estimates of x^e, which depend on the states and inputs of (1) due to y^e. Since y^e is not known in advance, for prediction, we propagate the mean and covariance of x^e by a model of (4)

$$\hat{\mu}^e_{k+1} = \hat{g}_\mu\left(\hat{\mu}^e_k, \hat{y}^e_k, x^s_k, u^s_k\right), \tag{5a}$$
$$\hat{\Sigma}^e_{k+1} = \hat{g}_\Sigma\left(\hat{\Sigma}^e_k, \Sigma^\epsilon_k, x^s_k, u^s_k\right), \tag{5b}$$

where $\hat{y}^e$ is the predicted measurement, $\hat{\mu}^e, \hat{\Sigma}^e$ are the predicted mean and covariance of x^e, and Σ^ϵ is the covariance of the measurement prediction error $\epsilon_y = q^e(f^e(\mu^e_{k-1}, \mu^\psi), \mu^\zeta, x^s_k, u^s_k) - y^e_k$ [5].

C. Constraints, Objective and Cost Function

Due to stochastic uncertainty in the model of the environment, we impose linear individual chance constraints (ICCs) between vehicle and environment,

$$\mathbb{P}\left[h^s_l x^s + h^e_l x^e \leq h^b_l\right] \geq 1 - \varepsilon_l, \quad l \in \mathbb{Z}_{[1,n_c,]}, \tag{6}$$

where ε_l is the maximum allowed probability of constraint violation for the l^{th} constraint.

Our objective is to control the vehicle motion based on (1) such that y^s tracks a reference $r^s \in \mathbb{R}^{n_y}$ while satisfying (2) and (6). Since (4) and (5) depend on the system states and inputs, the PAC-MPC strikes a balance between tracking and improving perception. Enhanced perception will reduce uncertainty and hence can improve tracking in the future. To this end, we proposed a cost function that stabilizes the system state x^s and the environment covariance Σ^e [5]

$$\begin{aligned} V_N\left(x^s_k, U_k, \Sigma^e_k, r^s_k\right) &= F(x^s_{N|k}, \Sigma^e_{N|k}, r^s_k) + \sum_{j=0}^{N-1} \ell(\Sigma^e_{j|k}, r^s_k) \\ &= F_c(x^s_{N|k}, r^s_k) + F_p(\Sigma^e_{N|k}, r^s_k) + \\ &\quad \sum_{j=0}^{N-1} \left(\ell_c(x^s_{j|k}, u^s_{j|k}, r^s_k) + \ell_p(x^s_{j|k}, u^s_{j|k}, \Sigma^e_{j|k}, r^s_k)\right), \end{aligned} \tag{7}$$

where $N \in \mathbb{Z}_+$ is the prediction horizon; $\ell_c(x, u, r)$ and $F_c(x, r)$ are the *control* stage and terminal costs, respectively; $\ell_p(x, u, \Sigma, r)$ and $F_p(x, \Sigma, r)$ are the *perception* stage and terminal costs, respectively; and $U_k = (u^s_{0|k}, \ldots, u^s_{N|k})$.

III. Multi-stage PAC-MPC

Even though in Section II the covariance propagation (4b) is fully defined by the system states and inputs, the estimator mean (4a) depends on the environment measurement y^e, which is an exogenous signal. In our previous works [4], [5], we used a prediction $\hat{y}^e$, while increasing the uncertainty by Σ^ϵ in (5b), to compensate for the prediction error. This may result in conservative performance if Σ^ϵ is over-estimated. Here, we propose a multi-stage approach [10], which predicts the environment measurements y^e using a tree of discrete scenarios. In each branch, we predict a different environment measurement realization, as shown in Fig. 1. Each path from the root node x^s_0 to a leaf node is a scenario denoted by the (additional) superscript $i = \{1, 2, \ldots, N_s\}$.

The multi-stage formulation allows for closed-loop prediction, i.e., it accounts for the fact that new information will become available in the future to determine the control actions. This, in turn, reduces the conservativeness compared to assuming a worst-case uncertainty realization [10]. The tree structure of the scenarios stems from the *non-anticipativity* constraints, which impose that all control inputs stemming from the same root node must be equal since realizations of the uncertainty cannot be anticipated, see Fig. 1. Since the scenario-tree grows exponentially with the prediction horizon, we branch the tree up to a certain stage N_B, the *branching horizon*. For time-steps past N_B, the uncertainty is assumed to be fixed at the nominal value for the corresponding scenario.

In msPAC-MPC, the cost function (7) is modified to minimize the expectation over the scenarios

$$\begin{aligned} \mathcal{V}_N(x^s_k, U_k, \Sigma^e_k, r^s_k) &= \mathbb{E}\left[V_N(x^s_k, U^i_k, \Sigma^e_k, r^s_k)\right] = \\ &\sum_{i=1}^{N_s} \omega_i V_N(x^s_k, U^i_k, \Sigma^e_k, r^s_k), \end{aligned} \tag{8}$$

where U^i_k is the sequence of control inputs for scenario i and ω_i is the weight or probability of each scenario, which is set to $\omega_i = 1/n_s$ in the absence of additional information.

Remark 1: In the case where the scenarios capture all the possible environment measurement realizations, Σ^ϵ in (5b)

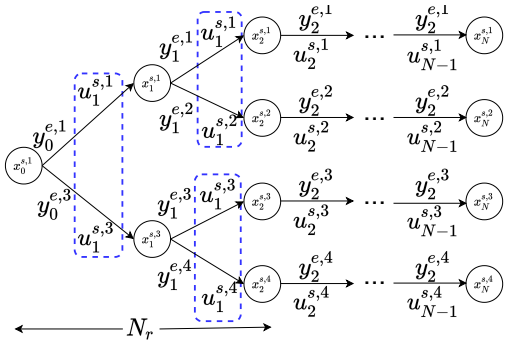

Fig. 1. Scenario tree where nodes are generated by different realizations of the measurement y^e_k. The dashed boxes represent the control inputs that must be equal according to the non-anticipativity constraints.

will be set to zero because the scenarios condition the trajectory on the realizations of the environment measurement y^e. Therefore, since $\hat{y}^e = y^e$ is "given", there will be no measurement prediction error. If the scenarios are obtained by sampling a continuous distribution, Σ^ϵ in (5b) may still be included, but it will be smaller than the one in [4], [5] since it only accounts for the discretization error, see, e.g., [11]. ■

Given the current state x^s_k, the msPAC-MPC problem is formulated as

$$V^*_N(x^s_k, \Sigma^e_k, r^s_k) = \min_{U_k} \sum_{i=1}^{N_s} \mathcal{V}_N\left(x^s_k, U^i_k, \hat{\Sigma}^e_k, r^s_k\right) \tag{9a}$$

$$\text{s.t. } x^{s,i}_{j+1|k} = f^s(x^{s,i'}_{j|k}, u^{s,i'}_{j|k}) \tag{9b}$$

$$\left(\hat{\mu}^{e,i}_{j+1|k}, \hat{\Sigma}^{e,i}_{j+1|k}\right) = \hat{g}\left(\hat{\mu}^{e,i'}_{j|k}, \hat{\Sigma}^{e,i'}_{j|k}, y^{e,i'}_{j|k}, x^{s,i'}_{j|k}, u^{s,i'}_{j|k}\right) \tag{9c}$$

$$u^{s,i}_{j|k} = u^{s,l}_{j|k} \quad \text{if} \quad x^{s,i'}_{j|k} = x^{s,l'}_{j|k} \tag{9d}$$

$$\left(x^{s,i}_{j|k}, u^{s,i}_{j|k}\right) \in \mathcal{X} \times \mathcal{U} \tag{9e}$$

$$\mathbb{P}\left[h^s_l x^{s,i}_{j|k} + h^e_l x^{e,i}_{j|k} \le h^b_l\right] \ge 1 - \varepsilon_l, \quad l \in \mathbb{Z}_{[1,n_c]} \tag{9f}$$

$$\left(x^{s,i}_{N|k}, r_{N|k}\right) \in \mathcal{Z}_f(\hat{\mu}^{e,i}_{N|k}, \hat{\Sigma}^{e,i}_{N|k}) \tag{9g}$$

$$x^{s,1}_{0|k} = x^s_k, , \quad \mu^{e,1}_{0|k} = \mu^e_k, \quad \Sigma^{e,1}_{0|k} = \hat{\Sigma}^{e,1}_{0|k}, \tag{9h}$$

where $N_s \in \mathbb{Z}_+$ is the number of scenarios, i' and l' denote the parent node of scenarios i and l, respectively, and $\mathcal{Z}_f(\hat{\mu}^{e,i}_{N|k}, \hat{\Sigma}^{e,i}_{N|k})$ is the terminal set. The optimal solution to (9) is denoted by $U^*_k = (U^{1,*}_k, \dots, U^{N_s,*}_k) = (u^{s,1,*}_{0|k}, \dots, u^{s,N_s,*}_{N|k})$. The first input $u^{s,1,*}_{0|k}$ is applied to the true system. Note that the non-anticipativity constraints ensure that $u^{s,i,*}_{0|k}$ is equal among all scenarios.

Remark 2: The scenarios may not model the environment measurements, rather functions for predicting them $\{\hat{y}^{e,i}_k = \phi^e_i(\mu^{e,i}_k, x^{s,i}_k, u^{s,i}_k, \mu^\psi, \mu^\zeta)\}_i$. Thus, measurements are predicted based on a set of possible environment evolutions, such as different behaviors of other vehicles in automated driving applications. ■

IV. PAC-MPC for Vehicle Control

In this section, we apply PAC-MPC for vehicle control.

A. Vehicle Model

To describe vehicle motion, we use the kinematic bicycle model [12]

$$\dot{p}_x = v\cos(\psi + \beta(\delta)) \tag{10a}$$

$$\dot{p}_y = v\sin(\psi + \beta(\delta)) \tag{10b}$$

$$\dot{v} = a \tag{10c}$$

$$\dot{\psi} = (v/l_r)\sin(\beta(\delta)), \tag{10d}$$

where p_x is the East (map-horizontal) position, p_y is the North (map-vertical) position, v is the velocity, ψ is the heading angle, and $\beta(\delta)$ is the body slip angle

$$\beta(\delta) = \arctan\left(\tan(\delta)\frac{l_r}{l_r + l_f}\right), \tag{11}$$

where l_f, l_r are the front and rear axle lengths. Vehicle model (10) is subject to the constraints

$$0 \le v \le v_{\max}, \qquad -\psi_{\max} \le \psi - \psi_{\text{rd}} \le \psi_{\max}, \tag{12a}$$

$$a_{\min} \le a \le a_{\max}, \qquad -\delta_{\max} \le \delta \le \delta_{\max}, \tag{12b}$$

where ψ_{rd} is the road heading. In (10), the acceleration a and the steering angle of the front wheel at the road δ are the control inputs. We discretize (10) with sampling period T_s to obtain f^s in (1), where $x^s = [p_x\ p_y\ v\ \psi]^\top$ and $u^s = [a\ \delta]^\top$. Constraints (12) define the sets $\mathcal{X}$ and $\mathcal{U}$ in (2).

B. Environment Model

For the simplicity of notation, we consider a straight road with three lanes and another vehicle, i.e., a moving obstacle. The lateral position of left and right road boundaries b^l, b^r with respect to the centerline are modeled as constant

$$\dot{b}^l = 0, \qquad \dot{b}^r = 0. \tag{13}$$

The obstacle is over-approximated by an ellipse, see Fig. 2, and its nominal longitudinal motion is modeled as

$$\dot{p}^o_{\text{lon}} = v^o_x. \tag{14}$$

The nominal lateral motion is modeled as an integrator in closed-loop with a tracking controller resulting in

$$\dot{p}^o_{\text{lat}} = -K^e p^o_{\text{lat}} + F^e r^e, \tag{15}$$

where K^e is a stabilizing gain determining the time to complete a lane change and F^e provides unitary gain.

We construct the environment prediction model by collecting (13), (14), (15), formulated in discrete time with sampling period T_s, and adding process noise $\psi \sim \mathcal{N}(0, \Sigma_\psi)$ and measurement noise $\zeta \sim \mathcal{N}(0, \Sigma_\zeta)$ to model disturbances and sensing uncertainty. This yields

$$x^e_{k+1} = A^e x^e_k + G^e r^e_k + B^e \psi_k, \tag{16a}$$

$$y^e_k = C^e_k(x^s_k, u^s_k)x^e_k + D^e_k(x^s_k, u^s_k)\zeta_k, \tag{16b}$$

where $x^e = [p^o_{\text{lon}}\ p^o_{\text{lat}}\ b^l\ b^r]^\top$. $C^e_k = C^e_k(x^s_k, u^s_k)$ and $D^e_k = D^e_k(x^s_k, u^s_k)$ define the measurement model. Here,

$$C^e_k = I,\ D^e_k = D^e_0\Big(I - \text{diag}\big([\rho_1\sigma_1\ \rho_1\sigma_1\ \rho_2\sigma_2\ \rho_3\sigma_3]\big)\Big), \tag{17}$$

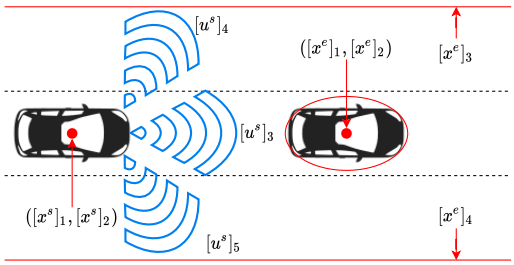

Fig. 2. Schematic of the case study, including the environment states $[x^e]_i$, $i = 1 \dots 4$, the vehicle position $p = ([x^s]_1, [x^s]_2)$, and the sensing inputs $[u^s]_i$, $i = 3 \dots 5$.

where $\sigma_i \in [0, 1]$, $i = 1, \dots, 3$, are the sensing control inputs that determine the quality of the measurements for the obstacle (longitudinal and lateral) position, the left road boundary, and the right road boundary, respectively. This represents a sensor (e.g., camera) that can be "tasked" to different directions and in different amounts. Increasing σ_1 will increase focus on the obstacle, thus reducing the uncertainty on $[x^e]_1$ and $[x^e]_2$; σ_2 will increase focus on the left road edge, reducing the uncertainty on $[x^e]_3$, as σ_3 does on $[x^e]_4$. Specifically, in (17), $D_0^e = 3$, $\rho_1 = 0.8$, $\rho_2 = \rho_3 = 0.95$.

We augment the system inputs by incorporating the sensing inputs such that $u^s = [a \ \delta \ \sigma_1 \ \sigma_2 \ \sigma_3]^\top$. The limited sensing budget is enforced by the constraint

$$[u_k^s]_3 + [u_k^s]_4 + [u_k^s]_5 \leq 2. \tag{18}$$

C. Uncertainty Propagation and Scenarios

Based on (16), we model $\hat{g}(\cdot)$ in (9c) as

$$\hat{\mu}_{k+1}^e = \Lambda_k \hat{\mu}_k^e + B^e \mu^\psi - L_k \hat{y}_k^e, \tag{19a}$$
$$\hat{\Sigma}_{k+1}^e = \Lambda_k \hat{\Sigma}_k^e \Lambda_k^\top + Q_k^e + R_k^e, \tag{19b}$$

where $L_k = L(x_k^s, u_k^s)$ is the estimator gain, $\Lambda_k = \Lambda(x_k^s, u_k^s) = A^e + L_k C_k^e$, $Q_k^e = B^e \Sigma^\psi B^{e\top}$, and $R(x_k^s, u_k^s) = L_k D_k^e \Sigma^\zeta (L_k D_k^e)^\top$. Although the covariance of the environment states $\hat{\Sigma}^e$ can be propagated deterministically, the mean of the environment states $\hat{\mu}^e$ depends on the predicted measurements $\hat{y}^e$, which are unknown. Here, we only apply the multi-stage approach for prediction of obstacle measurements since the road boundaries are constant.

Accordingly, we generate the predicted measurements $\hat{y}^e$ as the scenarios for the msPAC-MPC (9) based on scenarios for the reference of the lateral obstacle motion r^e in (16). Specifically, we consider deviations from the current position $r_{0|k}^e = x_k^e$ by half a lane width l_w within one sampling time $\hat{r}_{j+1|k}^e \in \{\hat{r}_{j|k}^e + 0.5 l_w, r_k^e, r_{j|k}^e - 0.5 l_w\}$.

D. Chance Constraints for Obstacle Avoidance

We enforce collision avoidance by $\mathcal{E}(p) \geq 0$, where $p = [p_x \ p_y]^\top = [[x^s]_1 \ [x^s]_2]^\top$ and $\mathcal{E}(p) = (p_x - [\mu^e]_1)^2/\gamma_{\text{Mj}}^2 + (p_y - [\mu^e]_2)^2/\gamma_{\text{mn}}^2 - 1$ gives an ellipsoid over-approximation of the obstacle with semi-minor/major axes $\gamma_{\text{mn}}, \gamma_{\text{Mj}}$.

We obtain ICCs as in (6) by linearizing the collision avoidance constraint, $\nabla \mathcal{E}(p)|_{p=x_b^e}^\top \cdot (p - x_b^e) \geq 0$, on a boundary point $x_b^e \in \mathbb{R}^2$. This results in $h^s x^s + h^e x_b^e \leq 0$, where $h^s = [-h^{e\top} \ 0 \ 0]^\top$ and $h^e = \left[\frac{2([x_b^e]_1 - [\mu^e]_1)}{\gamma_{\text{Mj}}^2}, \frac{2([x_b^e]_2 - [\mu^e]_2)}{\gamma_{\text{mn}}^2}\right]$. By applying the change of coordinate Φ that maps the ellipse to a circle with radius γ_{mn} centered at $([\tilde{\mu}^e]_1, [\tilde{\mu}^e]_2)$ [13], we select $\tilde{x}_b^e$ as the closest point to the position p

$$[\tilde{x}_b^e]_i = [\tilde{\mu}^e]_i + \frac{\gamma_{\text{mn}}([\tilde{x}^s]_i - [\tilde{\mu}^e]_i)}{\left(\sum_{i=1}^2 ([\tilde{x}^s]_i - [\tilde{\mu}^e]_i)^2\right)^{1/2}}, \ i = 1, 2, \tag{20}$$

and $x_b^e = \Phi^{-1} \tilde{x}_b^e$, where for x^s and μ^e we use the previous step predicted trajectories. Hence, the ICCs take the form

$$\mathbb{P}\left[h_{j+1|k-1}^s x_{j|k}^s + h_{j+1|k-1}^e (x_{j|k}^e + \Delta x_{j|k}^e) \leq 0\right] \geq 1 - \varepsilon_l,$$

where $\Delta x^e = x_b^e - x^e$; note that we approximated $\Delta x_{j|k}^e = \Delta x_{j+1|k-1}^e$ based on the previously predicted trajectory. ICCs (21) are reformulated as the deterministic constraints

$$h_{j+1|k-1}^s x^s + h_{j+1|k-1}^e \Delta x_{j|k}^e + h_{j+1|k-1}^e \mu_{j|k}^e + \alpha_l \left([h_{j+1|k-1}^e]_l \Sigma^e [h_{j+1|k-1}^e]_l^\top\right)^{1/2} \leq 0, \tag{21}$$

where $l \in \mathbb{Z}_{[1,n_c]}$ and $\alpha_l = F_N^{-1}(1 - \varepsilon_l)$ is the normal inverse cumulative distribution function.

Remark 3: In msPAC-MPC, and according (6), $\pi_{\text{sat}}^i = \prod_{l=1}^{n_c}(1 - \varepsilon_l^i)$ is the lower bound for the probability of satisfying the constraints at each step, conditioned on the scenarios $i \in \mathbb{Z}_{[1,N_s]}$. When the scenarios cover all measurement realizations, by applying the total probability law across the scenarios, this will also be the lower bound for the overall probability of satisfying the constraints at each step, $\pi_{\text{sat}} = \pi_{\text{sat}}^i$, for any $i \in \mathbb{Z}_{[1,N_s]}$. ■

Remark 4: The approximation of obstacle avoidance by separating hyperplanes has been shown to be effective in other applications [13] and is especially appropriate for automated driving. This is because many perception systems return detected features as planes or polygons, which could be handled in the msPAC-MPC problem (9). ■

E. Cost Function

For vehicle control, the msPAC-MPC cost function (7) is

$$V_N(x_k^s, U_k, \Sigma_k^e, r_k^s) = \sum_{j=0}^{N-1} (y_{j|k}^s - r_{j|k}^s) Q (y_{j|k}^s - r_{j|k}^s)^\top + u_{j|k}^s R (u_{j|k}^s)^\top + \|\hat{\Sigma}_{j|k}^e - \Sigma^r\|_F^2, \tag{22}$$

where $y^s = [v \ p_{\text{lat}} \ \psi_{\text{rd}}]$ is the performance output, p_{lat} is the lateral position with respect to the centerline, ψ_{rd} is the heading angle with respect to the road, which are expressed by a linear transformation of p_x, p_y, ψ based on the road heading ϑ [12], and r^s is the reference vector. In (22), Σ^r is the reference for the environment uncertainty, which is here defined as the steady state of (19b) for $[u]_i = 0$, $i = 3, 4, 5$. In (22), we did not include terminal cost and $\mathcal{Z}_f = \mathbb{R}^n$.

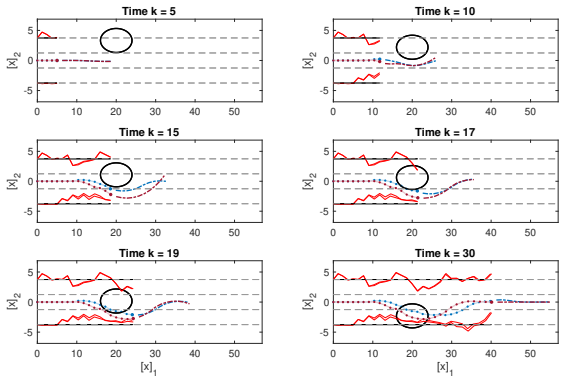

Fig. 3. Position phase plot snapshots for PAC-MPC (blue) and msPAC-MPC (dark red) in crossing obstacle scenario: past vehicle trajectory (dot marker); actual vehicle position (large dot); predicted future trajectories (dash-dot); actual environment states (obstacle ellipsoid and boundary lines, solid black); road boundaries (solid red); and lane boundaries (dash).

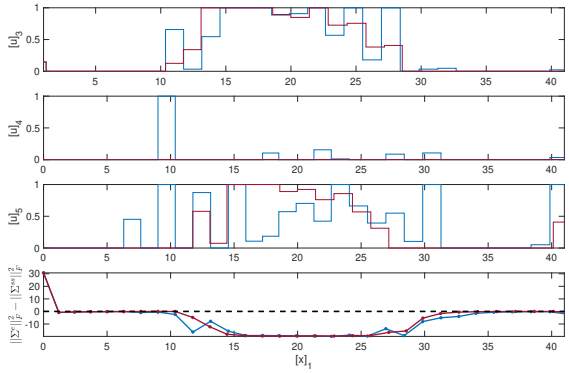

Fig. 4. Sensing inputs and the environment covariance histories with respect to longitudinal position $[x]_1$ for PAC-MPC (blue) and msPAC-MPC (dark red) in the crossing obstacle scenario.

V. Simulations

Based on the vehicle control problem described in Section IV, we evaluate the performance of msPAC-MPC for two cases: an obstacle changing lanes and a crossing obstacle on a narrowing road. We consider vehicle motion with respect to a frame that moves at 8 m/s on the center of the road , i.e., $p = ([x^s]_1, [x^s]_2)$ is the position with respect to such frame. The vehicle parameters are $l_f = 1.5$ m, $l_r = 2$ m, $v_{\max} = 30$ m/s, $\psi_{\max} = 0.8$ rad, $a_{\min} = -7$ m/s^2, $a_{\max} = 4$ m/s^2, $\delta_{\max} = 0.25$ rad, $\psi_{\mathrm{rd}} = 0$ rad, and $T_s = 0.1$ s; for the environment, $l_w = 2.5$ m, $\gamma_{\mathrm{Mj}} = 4$ m, $\gamma_{\mathrm{mn}} = 2$ m. The obstacle moves with the same speed as the frame (8m/s) and, hence, looks stationary with respect to the moving frame. The initial conditions for the system states and environment covariance are the same in the two case studies: $x_0^s = [0\ 0\ 10\ 0]^\top$ and $\hat{\Sigma}_0^e = \mathrm{diag}([5\ 5\ 1\ 1])$. The vehicle seeks to maintain a constant speed of 15 m/s with respect to the frame, while tracking the center of the road with the same heading angle $r^s = [0\ 15\ 0]^\top$. The covariance of the measurement prediction error for the PAC-MPC is $\Sigma^\epsilon = 0$ in order to asses the behavior when the design aims not to be over-conservative. The input-dependent measurement model is given in Section IV-B and the scenarios for the measurements are generated as in Section IV-C. The prediction horizon is $N = 10$, $N_B = 2$, and the simulation time is 4 s.

A. Obstacle Changing Multiple Lanes

We simulate a car changing multiple lanes to cross the road orthogonally with respect to the direction of travel. Fig. 3 shows six snapshots of the phase plot ($[x_k^s]_2$ vs. $[x_k^s]_1$) at time instants $k = \{5, 10, 15, 17, 19, 30\}$ for the PAC-MPC trajectory (blue) and the msPAC-MPC trajectory (dark red). In each snapshot, we show the past trajectories (dot marks), the current position (larger dot), and predicted (future) trajectories at current instant (dash-dot).

At the beginning of the simulation ($k \leq 5$), the trajectories of PAC-MPC and msPAC-MPC are indistinguishable since the vehicle is far from the obstacle and road boundaries and branching up to only 2 steps ahead does not affect the predicted trajectory. This is also evident from Fig. 4, where the environment covariance quickly reaches its steady-state and the sensing inputs remain inactive since there is no need to reduce the uncertainty to track the desired reference r^s.

As the vehicle and obstacle move closer together ($k = 10$–17), the sensing inputs become active and the trajectories of PAC-MPC and msPAC-MPC diverge. Due to the total sensing budget (18), PAC-MPC and msPAC-MPC reduce the uncertainty associated with the constraints that pose a more-imminent "threat" by increasing $[u^s]_3$, $[u^s]_5$. This reduces the environment covariance below its steady-state in order to find if a feasible path between the moving obstacle and the bottom road boundary exists. Fig. 3 shows that msPAC-MPC is more conservative than PAC-MPC because the multi-stage approach accounts for branching of future measurements up until the branching horizon, whereas the PAC-MPC selects its control actions based on the erroneous belief that the environment measurement will follow its open-loop prediction model since Σ^e was set to 0. This results in a trajectory that touches the obstacle ($k = 17$) in the case of PAC-MPC.

In terms of computation time, the msPAC-MPC is more demanding than PAC-MPC. In this demonstration, the average time to compute the msPAC-MPC was 0.51 s compared to 0.16 s for the PAC-MPC. We note that this is a MATLAB implementation on a mid-range 2018 laptop using a generic solver, which can be significantly sped-up using specialized C-coded solvers [14]. The msPAC-MPC computing time can be modified by the branching horizon N_B and by applying structured solvers for multi-stage optimization, see, e.g., [15].

B. Crossing Obstacle on a Narrowing Road

This case study simulates an obstacle changing multiple lanes across the road as the road narrows. That is, one road boundary is constant as in Section V-A, while the other starts as in Section V-A ($[x_{0|k}^e]_4 = -3.75$ m) but quickly narrows ($[x_{0|k}^e]_4 = -1.25$ m), representing a lane closure ahead.

Since the obstacle is moving towards the narrowing side of the road, the corridor for passing the obstacle of Fig. 3

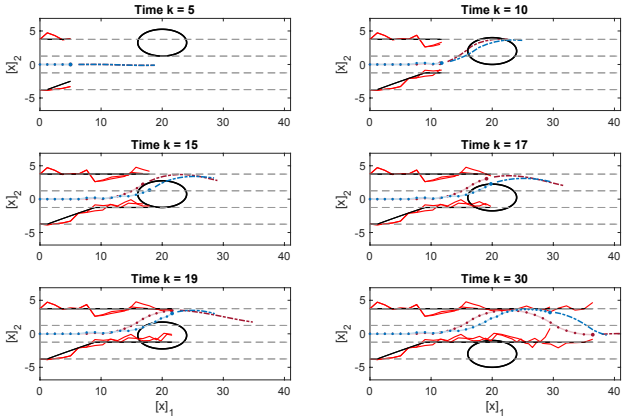

Fig. 5. Position phase plot snapshots for PAC-MPC (blue) and msPAC-MPC (dark red) in crossing obstacle on narrowing road scenario: past vehicle trajectory (dot marker); actual vehicle position (large dot); predicted future trajectories (dash-dot); actual environment states (obstacle ellipsoid and boundary lines, solid black); road boundaries (solid red); and lane boundaries (dash).

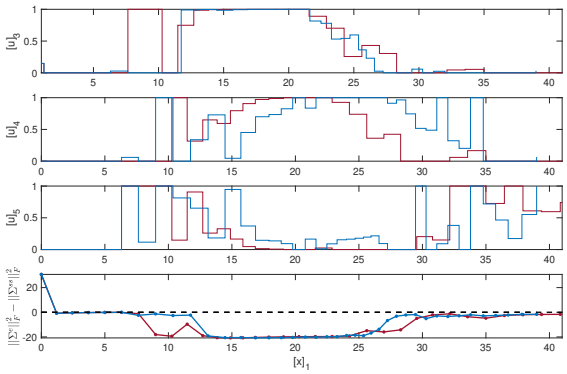

Fig. 6. Sensing inputs and the environment covariance histories with respect to longitudinal position $[x]_1$ for PAC-MPC (blue) and msPAC-MPC (dark red) in the crossing obstacle on narrowing road scenario.

becomes no longer viable. As such, the linearized constraint on the ellipse boundary would make it infeasible for the vehicle to maneuver around the obstacle from the other side. Thus, in the approach in Section IV-D, we allow to change separating hyperplane with respect to road boundary if there is no feasible path. The resulting behavior is shown in Fig. 5, where the vehicle is able to maneuver around the obstacle from above. The PAC-MPC intersects the obstacle ellipse, while the msPAC-MPC manages to remain outside of it; note that we set $\Sigma^{\varepsilon} = 0$. The input profiles in Fig. 6 are also different since $[u^s]_3$ and $[u^s]_5$ are mostly active in the beginning, when the vehicle is closer to the narrowing road boundary, and then the algorithm trades $[u^s]_5$ for $[u^s]_4$ when it gets closer to the other boundary. Again, the covariance goes below its steady-state since the sensing resources are employed to reduce the uncertainty in the environment in order to find a feasible path.

VI. Conclusions

In this paper, we extended the perception-aware chance constrained MPC (PAC-MPC) with a scenario-based prediction for future measurements, resulting in a multi-stage PAC-MPC formulation. The msPAC-MPC enforces the chance constraints without the need for including a conservative estimate of the measurement prediction error covariance. We showed how PAC-MPC and msPAC-MPC can be used for vehicle control in situations where obstacle and road boundaries are perceived through uncertain sensing of variable quality and when the scenarios are generated based on possible obstacle actions. In the future, we will speed up the computations by specialized methods for multi-stage [15] and chance-constrained MPC [14], and by adaptive scenario generation.

References

[1] M. Buehler, K. Iagnemma, and S. Singh, *The DARPA urban challenge: autonomous vehicles in city traffic*. Springer, 2009, vol. 56.

[2] C. Urmson, J. A. Bagnell, C. Baker, M. Hebert, A. Kelly, R. Rajkumar, P. E. Rybski, S. Scherer, R. Simmons, S. Singh *et al.*, "Tartan racing: A multi-modal approach to the DARPA Urban Challenge," *Carnegie Mellon University Tech. Report*, 2007.

[3] M. Montemerlo, J. Becker, S. Bhat, H. Dahlkamp, D. Dolgov, S. Ettinger, D. Haehnel, T. Hilden, G. Hoffmann, B. Huhnke *et al.*, "Junior: The stanford entry in the urban challenge," *Jour. field Robotics*, vol. 25, no. 9, pp. 569–597, 2008.

[4] A. D. Bonzanini, A. Mesbah, and S. Di Cairano, "Perception-aware chance-constrained model predictive control for uncertain environments," in *Proc. American Control Conf.*, 2021.

[5] ——, "On the stability properties of perception-aware chance-constrained mpc in uncertain environments," in *Proc. 60th IEEE Conf. on Decision and Control*, 2021.

[6] M. Farina, L. Giulioni, and R. Scattolini, "Stochastic linear model predictive control with chance constraints–a review," *Jour. Process Control*, vol. 44, pp. 53–67, 2016.

[7] A. Mesbah, "Stochastic model predictive control: An overview and perspectives for future research," *IEEE Control Systems*, vol. 36, no. 6, pp. 30–44, 2016.

[8] A. Mesbah, I. V. Kolmanovsky, and S. Di Cairano, "Stochastic model predictive control," in *Handbook of Model Predictive Control*. Springer, 2019, pp. 75–97.

[9] J. B. Rawlings and D. Q. Mayne, *Model predictive control: Theory and design*. Nob Hill Pub., 2009.

[10] D. Bernardini and A. Bemporad, "Scenario-based model predictive control of stochastic constrained linear systems," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*. IEEE, 2009, pp. 6333–6338.

[11] M. Donkers, W. Heemels, D. Bernardini, A. Bemporad, and V. Shneer, "Stability analysis of stochastic networked control systems," *Automatica*, vol. 48, no. 5, pp. 917–925, 2012.

[12] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[13] D. A. Marsillach, S. Di Cairano, and A. Weiss, "Fail-safe rendezvous control on elliptic orbits using reachable sets," in *Proc. American Control Conf.*, 2020, pp. 4920–4925.

[14] X. Feng, S. Di Cairano, and R. Quirynen, "Inexact adjoint-based sqp algorithm for real-time stochastic nonlinear mpc," in *Proc. IFAC World Congress*, 2020.

[15] J. Kang, A. U. Raghunathan, and S. Di Cairano, "Decomposition via ADMM for scenario-based model predictive control," in *Proc. American Control Conf.*, 2015, pp. 1246–1251.