Learning-based Stochastic Model Predictive Control with State-Dependent Uncertainty

Angelo D. Bonzanini   ADBONZANINI@BERKELEY.EDU
Department of Chemical and Biomolecular Engineering, University of California, Berkeley, CA 94720, USA.

Ali Mesbah   MESBAH@BERKELEY.EDU
Department of Chemical and Biomolecular Engineering, University of California, Berkeley, CA 94720, USA.

Abstract

The increasing complexity of modern engineering systems can introduce a great deal of uncertainty in our knowledge of system dynamics, which can, in turn, pose a major challenge to safe model-based control. This paper presents a learning-based stochastic model predictive control (LB-SMPC) strategy with chance constraints for tracking. The LB-SMPC strategy systematically handles mismatch between the actual system dynamics and a system model via a state-dependent uncertainty term that is intended to correct model predictions at each sampling time. A chance constraint handling method is presented to ensure state constraint satisfaction to a desired level for the case of state-dependent model uncertainty. Closed-loop simulations demonstrate the usefulness of LB-SMPC for control of a safety-critical plasma system for processing of (bio)materials with hard-to-model and time-varying dynamics.

Keywords: Learning-based control, model predictive control, chance constraints.

1. Introduction

Model predictive control (MPC) is an optimal control strategy widely used for the control of multi-variable, constrained systems (Mayne, 2014). Adequate models are critical for the success of MPC applications, since closed-loop performance and constraint satisfaction are heavily dependent on a sufficiently accurate system model. However, the growing complexity of modern systems renders their modeling increasingly challenging. This introduces significant uncertainties to the system models, which are exacerbated by the time-varying nature of the dynamics as well as the presence of exogenous disturbances during system operation. In addition, control-oriented models should be amenable to real-time computations. Thus, adapting simple physics-based or data-driven models based on data collected during system operation can enable accounting for model uncertainties while retaining the low computational cost necessary for online control.

Most research on learning-based MPC (LB-MPC) focuses on leveraging closed-loop data to learn a model that describes underlying dynamics more accurately, effectively reducing the plant-model mismatch (Hewing et al., 2019). The effectiveness of LB-MPC has been demonstrated in various applications, such as pH neutralization processes (Kocijan et al., 2004), gas-liquid separation plants (Likar and Kocijan, 2007), and robot path tracking (Ostafew et al., 2014). The notion of LB-MPC for uncertain systems entails decoupling safety and performance by using two models of the system: an approximate model with bounds on uncertainty and a second model that is updated by statistical methods (Aswani et al., 2013). The main idea behind this decoupling is to use the first model for establishing robustness guarantees offline, and the second model for improv-

ing closed-loop performance. This idea is extended in Limon et al. (2017) using a non-parametric machine learning technique, where learning-based robust MPC (LB-RMPC) approaches with guaranteed stability by design are proposed. The case of LB-RMPC under state-dependent uncertainty has recently been considered by Soloperto et al. (2018), where Gaussian process (GP) regression is used to correct for model uncertainty. In particular, constraints are tightened under state-dependent uncertainty, whose worst-case realization varies over the state space, thus reducing conservatism typically associated with robust constraint tightening methods. For a recent review on LB-MPC with a focus on safe learning the reader is referred to Hewing et al. (2019).

This paper presents a learning-based stochastic MPC (LB-SMPC) strategy with state-dependent uncertainty for tracking in chance-constrained stochastic systems. The goal is to learn a state-dependent model of system uncertainty (i.e., plant-model mismatch) online in order to improve the closed-loop performance of the controller, while guaranteeing chance constraint satisfaction. The main contributions include: (i) a state-dependent chance-constraint tightening approach for LB-SMPC based on learning the plant-model mismatch, thus allowing a systematic trade-off between robustness and performance while reducing conservativeness due to the state-dependent nature of the uncertainty; and (ii) a tracking formulation for LB-SMPC that circumvents the need for re-designing the terminal set, which is critical to guaranteeing robust feasibility, every time the target is changed. The effectiveness of the proposed LB-SMPC strategy is demonstrated in closed-loop simulations of a cold atmospheric plasma system for processing of (bio)materials (Gidon et al., 2017; Gidon et al., 2018).

2. Problem Setup

Consider the discrete-time stochastic system described by

$$x_{k+1} = Ax_k + Bu_k + w(x_k, u_k), \tag{1a}$$

$$y_k = Cx_k + Du_k, \tag{1b}$$

with states $x \in \mathbb{R}^n$, inputs $u \in \mathbb{R}^m$, zero-mean, stochastic uncertainty $w(x, u) \in \mathbb{R}^n$, and controlled variables $y \in \mathbb{R}^p$. The uncertainty $w(x, u)$ is bounded by a compact, state- and input-dependent set $\mathbb{W}(x, u)$. This uncertainty can be due to plant-model mismatch and/or exogenous disturbances. System description (1) provides an advantageous model structure because it offers the flexibility of learning the general (possibly nonlinear) uncertainty description $w(x, u)$ from data, while it incorporates a linear state-space model as a nominal system model. This structure enables decoupling safety and robustness aspects of the controller design from performance. The nominal model allows using reachability tools to provide guarantees for safety and robustness, while the additive uncertainty term $w(x_k, u_k)$ can be learned from data to update the model for performance optimization (Aswani et al., 2013).

Furthermore, system (1) is subject to hard constraints on states and inputs as well as individual chance constraints on states

$$\left[x_k^\top, u_k^\top \right]^\top \in \mathbb{X} \times \mathbb{U} = \mathbb{Z}, \tag{2}$$

$$\mathbb{P}[h_i^\top x_{k+1} \leq 1] \geq 1 - \varepsilon_i, \ i = 1 \ldots, n_c, \tag{3}$$

where $\mathbb{X}$ and $\mathbb{U}$ are the state and input constraint sets, ε_i is the maximum allowable probability of constraint violation for the ith chance constraint, and n_c is the total number of individual chance

constraints. The chance constraints (3) can be reformulated into deterministic constraints by defining a new set $\mathbb{X}^*$ as

$$\mathbb{X}^* = \left\{ x \in \mathbb{R}^n : h_i^\top x \le 1 - \gamma_i^* \right\}, \quad i = 1, \ldots, n_c, \tag{4}$$

$$\text{where} \qquad \gamma_i^* = \min \gamma_i \qquad \text{s.t.} \qquad \mathbb{P}[h_i^\top w_k \le \gamma_i] \ge 1 - \varepsilon_i.$$

We define the nominal system by neglecting the uncertainty $w(x_k, u_k)$ as

$$\tilde{x}_{k+1} = A\tilde{x}_k + Bu_k, \tag{5a}$$

$$\tilde{y}_k = C\tilde{x}_k + D\tilde{u}_k. \tag{5b}$$

3. Constraint Tightening for Stochastic Systems with State-Dependent Uncertainty

3.1. Characterization of Steady States

One of the control objectives in this work is to track time-varying reference targets. At steady state, there is no dynamic evolution of nominal system (5), i.e., $\tilde{x}_{k+1} = \tilde{x}_k = \tilde{x}$. Therefore, every pair of steady states and inputs $\tilde{z}^s = [\tilde{x}^s, \tilde{u}^s]^\top$ satisfies $\begin{bmatrix} A - I_n & B \end{bmatrix} \begin{bmatrix} \tilde{x}^s & \tilde{u}^s \end{bmatrix}^\top = 0$, and thus is an element of the null space of matrix $\begin{bmatrix} A - I_n & B \end{bmatrix}$ (Limon et al., 2008). We assume that (A, B) is controllable and therefore the dimension of this null space is m. Consequently, there exists a matrix $M_\theta \in \mathbb{R}^{(n+m) \times m}$ such that

$$\tilde{z}^s = M_\theta \tilde{\theta} \quad \text{and} \quad \tilde{y}^s = N_\theta \tilde{\theta}, \tag{6}$$

for any $\theta \in \mathbb{R}^m$ and $N_\theta = [C \, D] \, M_\theta$. The goal of parametrization (6) is to enlarge the terminal invariant set for tracking when compared to regulation to a fixed target (Limon et al., 2008). Hence, the controller will exhibit a larger region of attraction (ROA), ensuring feasibility for target changes (Paulson et al., 2019).

3.2. Dual Mode Prediction

Dual mode prediction ensures that state constraints are satisfied over an infinite prediction horizon, while convergence properties of the controller are guaranteed. At time k, predictions of system (1) are denoted as

$$x_{j+1|k} = Ax_{j|k} + Bu_{j|k} + w(x_{j|k}, u_{j|k}), \tag{7a}$$

$$y_{j|k} = Cx_{j|k} + Du_{j|k}. \tag{7b}$$

The idea is to define a terminal feedback controller such that it drives the system as close as possible to a desired target $u_{j|k} = \tilde{u}_k^s + K \left(x_{j|k} - x_k^s \right)$, where superscript s denotes steady-state values and K is the controller gain, typically chosen to be the linear quadratic regulator gain. Using (6), a control law parametrization can be devised as [1]

$$u_{j|k} = \pi(x_{j|k}, c_{j|k}) = Kx_{j|k} + L\tilde{\theta}_k + c_{j|k}, \tag{8}$$

---

1. A linear feedback parametrization is typically chosen because of its simplicity and computational tractability.

where $c_{j|k}$ are the $N-1$ decision variables, N is the prediction horizon, and $L = [-K \; I_m] M_\theta$ is a known matrix (Paulson et al., 2019). Substituting the parametrized control law (8) into (7) yields the closed-loop dynamics

$$x_{j+1|k} = A_K x_{j|k} + B\left(L\tilde{\theta}_k + c_{j|k}\right) + w(x_{j|k}, u_{j|k}), \tag{9}$$

where $A_K = A + BK$ with eigenvalues strictly inside the unit circle. Moreover, the nominal closed-loop system predictions can be recovered by neglecting the uncertainty in (9)

$$\tilde{x}_{j|k} = A_K \tilde{x}_{j|k} + B\left(L\tilde{\theta}_k + c_{j|k}\right). \tag{10}$$

The idea behind dual mode prediction is to switch between a receding-horizon controller and a local linear controller, depending on whether the state is inside or outside the terminal region (Michalska and Mayne, 1993). In mode 1, which includes steps $k \in [0, N-1]$, the decision variables $c_{j|k}$ are free to vary. In mode 2, which includes $k \in [N, \infty)$, $c_{j|k}$ is set to 0, effectively switching to a linear controller of the form $u_{j|k} = u_k^s + K(x_{j|k} - x_k^s)$. In mode 2, constraints are enforced by ensuring that the system is within the terminal region $\mathbb{X}_f$ (Kouvaritakis et al., 2010). By definition, the terminal set is constructed such that when the system converges to $x_N \in \mathbb{X}_f$, the closed-loop dynamics under the terminal control law are guaranteed to stay within $\mathbb{X}_f$, i.e., $x_k \in \mathbb{X}_f \Rightarrow x_{k+1} = A_K x_k + BL\tilde{\theta}_k + w_k \in \mathbb{X}_f$.

3.3. State-Dependent Constraint Tightening

First, we tighten the hard input constraints using existing reachability analysis tools (Chisci et al., 2001). We denote the set that contains the worst-case state-dependent error between the real states (1) and the nominal predictions (10) j steps ahead as $\mathbb{W}_j(x)$. Then, the input constraints can be tightened as $\overline{\mathbb{U}}_j = \mathbb{U} \ominus K\mathbb{W}_j(x)$, where K is the LQR gain.

Figure 1: Schematic for backwards reachable sets.

The chance constraints (3) must be satisfied up to a pre-specified probability. As shown in (4), chance constraints can be converted to deterministic constraints using backoff parameters γ_i^*. Then, the deterministic constraint sets of nominal system (10), i.e., $\mathbb{X}^*$ and $\mathbb{U}$, are recursively tightened at each prediction time j to generate sets $\overline{\mathbb{X}}_j$ and $\overline{\mathbb{U}}_j$. Therefore, if the nominal system (10) lies within $\overline{\mathbb{X}}_j \times \overline{\mathbb{U}}_j = \overline{\mathbb{Z}}_j$, then the real system (9) is guaranteed to satisfy constraints $\mathbb{X}^*$ and $\mathbb{U}$. To obtain the tightened state constraint sets $\overline{\mathbb{X}}_j$ offline, we first need to compute the initial set $\overline{\mathbb{X}}_1 = \mathbb{X}^*$ by determining the backoff parameters γ_i^*. These can be calculated exactly using (4) if the probability distribution of $w(x_k, u_k)$ is known. Alternatively, γ_i^* can be determined using non-parametric statistical methods (Lorenzen et al., 2017; Owen, 2001; Tempo et al., 2012). From (4) it is evident that any $\hat{\gamma}_i \neq \gamma_i^*$ will satisfy $\hat{\gamma}_i \geq \gamma_i^*$. An estimate of $\hat{\gamma}_i$ can be obtained from the empirical cumulative distribution function (ECDF) of the uncertainties, as described in Santos et al. (2019).

The idea behind the proposed state-dependent constraint tightening approach is to compute backwards reachable sets for all states $x \in \mathbb{X}$ for N time instances. These sets are denoted by $\mathbb{H}_{-j}$. A schematic is shown in Figure 1. After computing backwards reachable sets (following
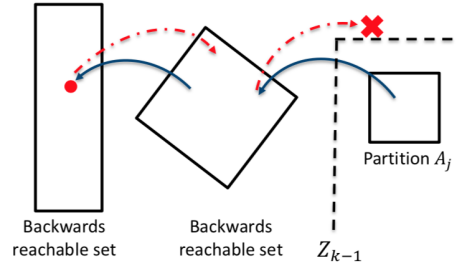
the solid blue arrows), we can estimate the maximum possible uncertainty at every individual state $x_i \in \mathbb{H}_{-j}$ by considering the worst-case uncertainty propagation from that state using the general uncertainty model $w(x_k, u_k)$ (following the red dotted arrows). Then, if the state-dependent worst-case system evolution results in a state outside of the previous tightened constraint, $\overline{\mathbb{X}}_{j-1}$, x_i is discarded, thus resulting in further constraint tightening in $\overline{\mathbb{X}}_j$. For computational tractability, the state space is partitioned into mutually exclusive and collectively exhaustive subsets $\mathbb{A}_i$ and the algorithm is applied to all subsets, instead of all states (Soloperto et al., 2018). In addition, we introduce the operator $G(\cdot)$ and the backwards reachable sets $\mathbb{H}_{-j}(\cdot)$ for system (10)

$$G(\mathbb{A}) := \cup_{x \in \mathbb{A}} \mathbb{W}(x), \ \mathbb{A} \subseteq \mathbb{X}, \quad \text{and} \quad \mathbb{H}_{-j}(\mathbb{A}_i) := A^{-j} \left(\{\mathbb{A}_i\} \oplus \left\{ \sum_{i=0}^{j-1} A^i B \overline{\mathbb{U}}_{j-1-i} \right\} \right) \cap \mathbb{X}.$$

The proposed constraint-tightening approach is summarized in Algorithm 1.

Algorithm 1: Chance Constraint Tightening Under State- and Input-dependent Uncertainty

1. Define the initial set $\overline{\mathbb{X}}_1 = \mathbb{X}^* \subseteq \mathbb{X}$ using (4) or the ECDF of the uncertainties.

2. Partition the set $\overline{\mathbb{X}}_1$ into $r > 0$ mutually exclusive and collectively exhaustive subsets $\mathbb{A}_i$, such that $\overline{\mathbb{X}}_1 = \cup \mathbb{A}_i$.
   Note that the uncertainty on any single subset $\mathbb{A}_i$ is the union of all of the uncertainties of the states contained in $\mathbb{A}_i$. That is, $\mathbb{W}(\mathbb{A}_i) = \cup_{x \in \mathbb{A}_i} \mathbb{W}(x) = G(\mathbb{A}_i)$.

3. **for** $j = 2 : N$ do

4.   **for** all $\mathbb{A}_i \subseteq \overline{\mathbb{X}}_{j-1}$ do

5.   Compute sets $\mathbb{H}_{-j}(\mathbb{A}_i)$ such that $\tilde{x}_{j|k} \in \mathbb{A}_i \Rightarrow \tilde{x}_{0|k} \in \mathbb{H}_{-j}(\mathbb{A}_i)$.

6.   The uncertainty $\mathbb{W}(\tilde{x}_{0|k})$ can be overbounded by considering the uncertainty acting on its backwards reachable sets, i.e., $G\left(\mathbb{H}_{-j}(\mathbb{A}_i)\right)$. Therefore,
   $\mathbb{W}(x_k) = \mathbb{W}(\tilde{x}_{0|k}) \subseteq G\left(\mathbb{H}_{-j}(\mathbb{A}_i)\right) =: \hat{\mathbb{W}}_{-j}(\mathbb{A}_i)$.

7.   Check if $\mathbb{A}_i$ can be included in $\overline{\mathbb{X}}_j$

         if $\mathbb{A}_i \oplus A_K^{j-1} \hat{\mathbb{W}}_{-j}(\tilde{x}_{j|k}) \subseteq \overline{\mathbb{X}}_{j-1}$:
               $\mathbb{A}_i \in \overline{\mathbb{X}}_j$.
        else:
               $\mathbb{A}_i \notin \overline{\mathbb{X}}_j$ (discard).
        end if
      end for
   end for

Remark 1 *Note that* $\max\{w(x_i, u)\} \leq \max\{w(x, u)\}$*, leading to less conservative results than assuming a constant (worst-case) uncertainty.*

Remark 2 *The sets* $\overline{\mathbb{X}}_j^*$ *resulting from Algorithm 1 may not be convex, even if the starting partitions* $\mathbb{A}_i$ *are chosen to be convex. To circumvent this, an inner approximation of* $\overline{\mathbb{X}}_j^*$ *may be constructed and used in the place of* $\overline{\mathbb{X}}_j^*$*.*

Remark 3 *For continuous state and input domains, partitioning* $\overline{\mathbb{X}}_1$ *into subsets* $\mathbb{A}_i$ *depends on the desired tradeoff between computational complexity and accuracy in the uncertainty characterization. The offline computational complexity of the constraint tightening scales as* $\mathcal{O}(n_1 n_2 \cdots n_N)$*,*

where n_i is the number of partitions in the ith dimension of x_k. Thus, as the number of partitions increases, the state dependence of the uncertainty is captured more accurately, but the offline computational complexity also increases. However, note that the constraint tightening is carried out offline, so the online computational complexity remains similar to that of nominal MPC.

4. Learning-based SMPC for Tracking

We use the constraint tightening approach described in Algorithm 1 to formulate the tracking LB-SMPC problem for system (1) with state-dependent uncertainty. The steady-state parametrization (6) results in an enlargement of the terminal set, which circumvents the need to redesign the terminal set every time the reference is changed. Furthermore, since the expensive constraint set computations are performed offline, this constraint-tightening approach features an online computational complexity similar to that of nominal MPC.

To steer the system to a desired target, we define the following cost function

$$V_N(\mathbf{c_k}, \tilde{\theta}_k, x_k, y_k^t) = \sum_{j=0}^{N-1} ||y_{j|k} - \tilde{y}_k^s||_Q^2 + ||\tilde{u}_{j|k} - \tilde{u}_k^s||_R^2 + y_N^\top P y_N + V_o\left(\tilde{y}_k^s - \tilde{y}_k^t\right),$$

where $(\tilde{x}_k^s, \tilde{u}_k^s) = M_\theta \tilde{\theta}_k$ and $\tilde{y}_k^s = N_\theta \tilde{\theta}_k$ are defined in (6), y_k^t is the desired target, and Q, R, P are suitably chosen weight matrices. To ensure convergence toward y_k^t, an offset cost $V_o(\cdot)$ is added to the cost function. This serves the purpose of penalizing deviations between the artificial reference and the desired target. Given a measured state x_k at sampling time k, the tracking LB-SMPC problem can be formulated as

$$V_N^*(x_k) = \min_{\mathbf{c_k}, \tilde{\theta}_k} V_N(\mathbf{c_k}, \tilde{\theta}_k, x_k, y_k^t) \tag{11a}$$

$$\text{s.t. } \tilde{x}_{0|k} = x_k, \quad x_{0|k} = x_k \tag{11b}$$

$$\tilde{x}_{j+1|k} = A\tilde{x}_{j|k} + B\tilde{u}_{j|k}, \tag{11c}$$

$$x_{j+1|k} = Ax_{j|k} + B\tilde{u}_{j|k} + w(x_{j-l:j}, \tilde{u}_{j-l:j}), \tag{11d}$$

$$\tilde{u}_{j|k} = Kx_{j|k} + L\tilde{\theta}_k + c_{j|k}, \tag{11e}$$

$$y_{j|k} = Cx_{j|k} + D\tilde{u}_{j|k} \tag{11f}$$

$$\left[\tilde{x}_{j|k}^\top, \ \tilde{u}_{j|k}^\top\right]^\top \in \overline{\mathbb{X}}_j \times \overline{\mathbb{U}}_j, \qquad j = 1, \ldots, N-1, \tag{11g}$$

$$\left[\tilde{x}_{N|k}^\top, \tilde{\theta}_k^\top\right]^\top \in \Omega_t^a, \tag{11h}$$

where $\mathbf{c}_k = \{c_{0|k}, ..., c_{N-1|k}\}$ and Ω_t^a is the augmented terminal set, which can be computed as described in Paulson et al. (2019). Two models are used to formulate the OCP (11). Constraints (11g) are enforced with respect to the nominal dynamics (11c), while the objective function is minimized with respect to the predictions corrected by the learned model (11d). The optimal solution to (11) is denoted by $\mathbf{c}_k^* = \{c_{0|k}^*, ..., c_{N-1|k}^*\}$. At each sampling time k, the control law (8) takes the form $u_{j|k} = Kx_k + L\tilde{\theta}_k^* + c_{0|k}^*$ due to the receding-horizon implementation. Recursive feasibility, stability, and convergence properties of the proposed tracking LB-SMPC strategy are discussed in Bonzanini et al. (2020).

5. Case Study: Predictive Control of a Safety-Critical Plasma Jet

The proposed tracking LB-SMPC strategy is demonstrated in a simulation case study of an atmospheric pressure plasma jet (APPJ) with applications in processing of (bio)materials. The APPJ under consideration is a multivariable process with two inputs and two outputs; see Gidon et al. (2018) for a detailed description of the APPJ. The inputs are He flowrate (q) and applied power (P); and the outputs are surface temperature (T) and plasma intensity (I). The deviation output and input variables are defined as $y = [T - T^s, \ 0.1(I - I^s)]^\top$ and $u = [q - q^s, \ P - P^s]^\top$, respectively, where the superscript s denotes steady state.

The control objective is to track a time-varying setpoint for surface temperature and plasma intensity, as well as to satisfy a safety-critical constraint on surface temperature, while the dynamics of plasma-surface interactions undergo unmodeled variations. This mimics a scenario in which a medical professional operates the plasma device on a tissue with spatially varying properties, while ensuring that safety-critical system constraints are not violated such that no harm is inflicted to the patient. A linear state-space model is identified via sub-space identification (Gidon et al., 2018) augmented with an additive state-dependent uncertainty, which is learned from the closed-loop data using Gaussian Process (GP) regression (Rasmussen and Williams, 2006).

Figure 2 shows the plasma operation begins by tracking the setpoint $[T^s, I^s]^\top = [40\,°C, 110\,\text{a.u.}]^\top$ on the insulating surface, which is the nominal operating surface. At time $t = 115$ s, the plant model is switched to treating a conductive surface, which represents a significant change in the system dynamics. To examine whether the controller can guarantee state constraint satisfaction, a setpoint change to $[T^s, I^s]^\top = [42.5\,°C, 110\,\text{a.u.}]^\top$ is introduced at $t = 260$ s to drive surface temperature close to its upper constraint $43\,°C$.
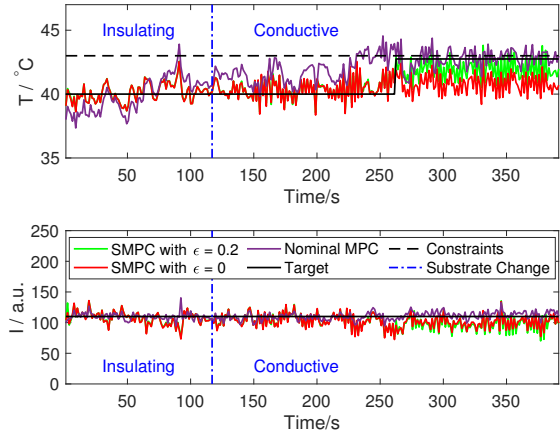

Figure 2: Closed-loop profiles of surface temperature and plasma intensity for the LB-SMPC strategy with $\varepsilon = 0$ and $\varepsilon = 0.2$ in comparison with nominal MPC with no learning. The surface dynamics undergo a significant change at time 115 s.
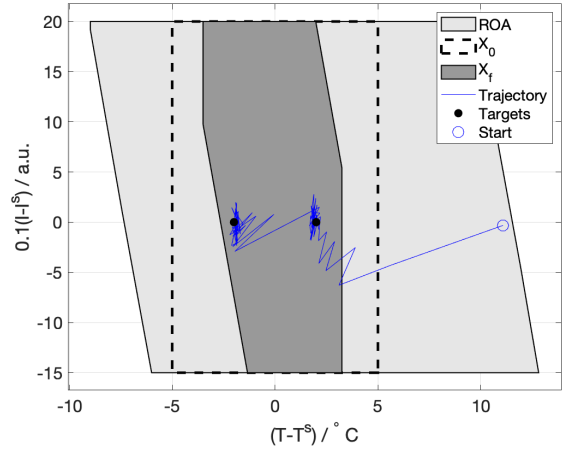
Figure 3: Phase plot of closed-loop surface temperature and plasma intensity for the mean of 100 closed-loop runs of the LB-SMPC strategy with $\varepsilon = 0.2$. Initial constraints (X_0) are shown in dashed lines, region of attraction (ROA) in light gray, and terminal set (X_f) in darker gray.

We present the performance of the proposed LB-SMPC strategy under two different levels of allowed constraint violation (i.e., $\varepsilon = 0$ and $\varepsilon = 0.2$), and compare its performance to that of nominal MPC. 100 closed-loop simulations are run with different uncertainty realizations, and one sample closed-loop profile from each case is shown in Figure 2. On the insulating surface, LB-SMPC performs slightly better than nominal MPC due to learning of the uncertainty. This performance difference becomes more evident once the surface is changed at $t = 115$ s. The performance of nominal MPC worsens, since the plant-model mismatch becomes larger. On the other hand, the LB-SMPC controller is effective in predicting the plant-model mismatch, leading to better control performance. Furthermore, up until $t = 260$ s, the closed-loop profiles are far enough from the constraint such that constraint tightening does not affect them, leading to similar performance between the controllers with $\varepsilon = 0$ and $\varepsilon = 0.2$. However, when the setpoint is changed from $40\,°C$ to $42.5\,°C$ (at $t = 260$ s), the two controllers perform differently. When $\varepsilon = 0$, the controller undershoots the setpoint but does not violate the temperature constraint. The 100 simulations of the closed-loop system indicate that this is the case for various uncertainty realizations, with very few exceptions that occur because 99% confidence intervals are used as the uncertainty bounds derived from the Gaussian process model. As expected, increasing the value of ε to 0.2 results in a better tracking performance, on average, at the expense of violating the constraint. From the 100 closed-loop simulations, we observe the constraint is violated 19.1% of the time, which is very close to the specified upper bound, suggesting that the proposed chance constraint tightening approach for the case of state-dependent uncertainty is not conservative.

Furthermore, to investigate the effect of changing ε on the region of attraction (ROA) of the controller, we run 100 closed-loop simulations starting from a point outside the original constraint set. Figure 3 illustrates that the proposed tracking LB-SMPC strategy can smoothly handle setpoint changes while also enlarging the ROA. This is due to two main reasons. Firstly, since the chance-constrained formulation of SMPC allows for some constraint violation, the controller can start from a point outside the original constraints and still steer the system to the terminal set while staying below the constraint violation threshold. In contrast, this is not allowed in robust MPC with hard constraints, where any starting point outside the original constraint set is by definition considered infeasible. And secondly, the enlarged terminal set for tracking compared to that for regulation allows further increase in the size of the ROA, since there are more allowable points to which the controller has to drive the system to at the end of the prediction horizon.

6. Conclusions and Future Work

This paper presents a learning-based MPC strategy for chance-constrained stochastic systems with state-dependent model uncertainty. It is shown that online adaptation of the state-dependent model uncertainty can improve control performance, while ensuring constraint satisfaction to a desired probability level. In addition, the enlarged terminal region for tracking will allow for expanding the work space of the controller. Future work will investigate how to fully leverage the learning of the state-dependent uncertainty for online adaptation of the tightened constraint sets.

References

Anil Aswani, Humberto Gonzalez, Shankar Sastry, and Claire Tomlin. Provably safe and robust learning-based model predictive control. *Automatica*, 49(5):1216–1226, 2013.

Angelo D. Bonzanini, David B. Graves, and Ali Mesbah. Learning-based stochastic model predictive control for reference tracking with state-dependent uncertainty: An application to cold atmospheric plasmas. *IEEE Transactions on Control Systems Technology*, 2020. Submitted.

Luigi Chisci, John Anthony Rossiter, and Giovanni Zappa. Systems with persistent disturbances: Predictive control with restricted constraints. *Automatica*, 37:1019–1028, 2001.

Dogan Gidon, David B Graves, and Ali Mesbah. Effective dose delivery in atmospheric pressure plasma jets for plasma medicine: A model predictive control approach. *Plasma Sources Science and Technology*, 26(8):085005, 2017.

Dogan Gidon, Brandon Curtis, Joel A. Paulson, David B. Graves, and Ali Mesbah. Model-based feedback control of a kHz-excited atmospheric pressure plasma jet. *IEEE Transactions on Radiation and Plasma Medical Sciences*, 2(2):129–137, 2018.

Lukas Hewing, Kim P. Wabersich, Marcel Menner, and Melanie N. Zeilinger. Learning-based model predictive control: Toward safe learning in control. *Annual Review of Control, Robotics, and Autonomous Systems*, 3:10.1–10.28, 2019.

Juš Kocijan, Roderick Murray-Smith, Carl E. Rasmussen, and Agathe Girard. Gaussian process model based predictive control. In *Proceedings of the American Control Conference*, pages 2214–2219, Boston, MA, 2004.

Basil Kouvaritakis, Mark Cannon, Sasa V. Rakovic, and Qifeng Cheng. Explicit use of probabilistic distributions in linear predictive control. In *Proceedings of the UKACC International Conference on Control*, pages 559–564, Coventry, 2010.

Bojan Likar and Juš Kocijan. Predictive control of a gas–liquid separation plant based on a Gaussian process model. *Computers & Chemical Engineering*, 31(3):142–152, 2007.

Daniel Limon, Ignacio Alvarado, Teodoro Alamo, and Eduardo F. Camacho. MPC for tracking piecewise constant references for constrained linear systems. *Automatica*, 44:2382–2387, 2008.

Daniel Limon, Jan Peter Calliess, and Jan Marian Maciejowski. Learning-based nonlinear model predictive control. In *Proceedings of the IFAC World Congress*, volume 50, pages 7769–7776, Toulouse, 2017.

Matthias Lorenzen, Fabrizio Dabbene, Roberto Tempo, and Frank Allgöwer. Constraint-tightening and stability in stochastic model predictive control. *IEEE Transactions on Automatic Control*, 62:3165–3177, 2017.

David Q Mayne. Model predictive control: Recent developments and future promise. *Automatica*, 50(12):2967–2986, 2014.

Hanna Michalska and David Q. Mayne. Robust receding horizon control of constrained nonlinear systems. *IEEE Transactions on Automatic Control*, 38:1623–1633, 1993.

Chris J. Ostafew, Angela P. Schoellig, and Timothy D. Barfoot. Learning-based nonlinear model predictive control to improve vision-based mobile robot path-tracking in challenging outdoor environments. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4029–4036, Hong Kong, 2014.

Art B. Owen. *Empirical likelihood*. Chapman and Hall/CRC, Boca Raton, Florida, 2001.

Joel A. Paulson, Tito L.M. Santos, and Ali Mesbah. Mixed stochastic-deterministic tube MPC for offset-free tracking in the presence of plant-model mismatch. *Journal of Process Control*, 83: 102–120, 2019.

Carl E. Rasmussen and Chris K. I. Williams. *Gaussian processes for machine learning.* MIT Press, Cambridge, 2006.

Tito L. M. Santos, Angelo D. Bonzanini, and Ali Mesbah. A constraint-tightening approach to nonlinear model predictive control with chance constraints for stochastic systems. In *Proceedings of the American Control Conference*, pages 1641–1647, Philadelphia, PA, 2019.

Raffaele Soloperto, Matthias A. Müller, Sebastian Trimpe, and Frank Allgöwer. Learning-based robust model predictive control with state-dependent uncertainty. In *Proceedings of the IFAC Conference on Nonlinear Model Predictive Control*, pages 442–447, Madison, WI, 2018.

Roberto Tempo, Giuseppe Calafiore, and Fabrizio Dabbene. *Randomized algorithms for analysis and control of uncertain systems: with applications*. Springer Science & Business Media, London, UK, 2012.