Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Learning Robustness with Bounded Failure: An Iterative MPC Approach

Monimoy Bujarbaruah† * Akhil Shetty† * Kameshwar Poolla * Francesco Borrelli *

* *University of California Berkeley, Berkeley, CA 94701 USA (e-mails: {monimoyb, shetty.akhil, poolla, fborrelli}@ berkeley.edu).*

Abstract: We propose an approach to design a Model Predictive Controller (MPC) for constrained Linear Time Invariant systems performing an iterative task. The system is subject to an additive disturbance, and the goal is to learn to satisfy state and input constraints robustly. Using disturbance measurements after each iteration, we construct *Confidence Support* sets, which contain the true support of the disturbance distribution with a given probability. As more data is collected, the Confidence Supports converge to the true support of the disturbance. This enables design of an MPC controller that avoids conservative estimate of the disturbance support, while simultaneously bounding the probability of constraint violation. The efficacy of the proposed approach is then demonstrated with a detailed numerical example.

Keywords: Predictive Control for Linear Systems, Iterative Predictive Control, Robust Convex Optimization, Confidence Intervals, Bootstrap.

1. INTRODUCTION

As data-driven decision making and control becomes ubiquitous (Tanaskovic et al. (2017); Rosolia et al. (2018)), system identification methods are being integrated with control algorithms for control of uncertain dynamical systems. The uncertainty in these systems can be typically attributed to two factors: (i) model uncertainty (eg. modeling mismatch and inaccuracies), and (ii) exogenous disturbances (eg. sensor noise). For such uncertain systems subject to state and input constraints, Model Predictive Control (MPC) (Mayne et al. (2000); Borrelli et al. (2017)) is a commonly used approach for ensuring robust constraint satisfaction.

The field of Adaptive MPC (Tanaskovic et al. (2014); Köhler et al. (2019)) deals with learning the model uncertainty to improve controller performance over time. These methods rely upon Set Membership approaches, which assume known set based bounds on the exogenous disturbances. As these disturbance supports are actually unknown in practice, conservative over-approximations are used for control design. This results in the controller either being infeasible, or incurring higher costs by following highly sub-optimal trajectories. This motivates learning the disturbance support over time in order to improve controller performance. In such cases, it is necessary to allow the possibility of *failure*, i.e, violation of imposed constraints. Such violations are acceptable for certain non safety critical robotic applications.

To that end, numerous works in MPC literature have considered constructing probabilistic approximations of both the model uncertainty and disturbance support (Zhang et al. (2013); Hewing and Zeilinger (2017); Soloperto et al. (2018)), allowing room for violations of imposed constraints with a certain probability. Methods such as (Hewing and Zeilinger (2017); Soloperto et al. (2018)), utilize Gaussian Process (GP) Regression to model and update the uncertainty in the system. However, they have no theoretical bounds for rate of constraint violations by the closed loop system over time.

Assuming the presence of *only* exogenous disturbances, (Zhang et al. (2013)) addresses this issue by constructing disturbance support sets offline using the scenario approach (Tempo et al., 2012, Chapter 12). This approach involves solving a scenario program with potentially large number of samples, which is computationally expensive. Moreover, the rate of constraint violation is dependent on the number of disturbance samples available offline. In certain settings (for eg., iterative tasks), it is often the case that one starts the controller having observed no samples apriori. While learning the disturbance support over time in such cases, it is desirable to have a user-specified upper bound for probability of failure over all time. The approach in (Zhang et al. (2013)) is unable to satisfy such an upper bound at all times, since the required number of samples could be unavailable during operation.

In this paper, we present an approach to design an MPC controller for constrained LTI systems performing an iterative task (Rosolia and Borrelli (2017)). Like (Zhang et al. (2013)) we consider an additive disturbance in the system, under no uncertainty in the system matrices. Instead of considering a conservative over-approximation of the disturbance support such as (Tanaskovic et al. (2014); Köhler et al. (2019)), we learn this set from observed

⋆ † Authors contributed equally to this work.
⋆⋆This work was partially funded by Office of Naval Research grant ONR-N00014-18-1-2833, National Science Foundation under grants EAGER-1549945 and CPS-1646612, and by the National Research Foundation of Singapore under a grant to the Berkeley Alliance for Research in Singapore.

Copyright lies with the authors

disturbance samples. While doing so, we guarantee a user-specified upper bound on the probability of failure over all iterations. Our main contributions can be summarized as:

- We introduce the notion of a *Confidence Support*, which is guaranteed to contain the true disturbance support with a specified probability. Constructing and updating the Confidence Supports after each iteration is computationally cheap, as opposed to (Zhang et al. (2013)).
- Using these Confidence Supports, we attempt robust MPC design and demonstrate satisfaction of desired upper bound on probability of failure in each iteration. For any value of user-specified upper bound on probability of failure, the controller is able to learn robust satisfaction of imposed constraints asymptotically, without suffering conservatism that is inherent to existing approaches (Tanaskovic et al. (2014); Lorenzen et al. (2019)).

2. PROBLEM FORMULATION

We consider uncertain linear time-invariant systems of the form:

$$x_{t+1} = Ax_t + Bu_t + w_t, \tag{1}$$

where $x_t \in \mathbb{R}^d$ is the state at time step t, $u_t \in \mathbb{R}^m$ is the input, and A and B are known system matrices of appropriate dimensions. At each time step t, the system is affected by an independently and identically distributed (i.i.d.) random disturbance $w_t \overset{\text{iid}}{\sim} \mathcal{P}$ with a convex and compact support $\mathbb{W} \subset \mathbb{R}^d$. We aim to satisfy state and input constraints on the system robustly. We define $H_x \in \mathbb{R}^{s\times d}$, $h_x \in \mathbb{R}^s$, $H_u \in \mathbb{R}^{o\times m}$ and $h_u \in \mathbb{R}^o$. We can then write the imposed constraints for all time steps $t \geq 0$ as:

$$\mathbb{Z} := \{(x,u) : H_x x \leq h_x,\ H_u u \leq h_u\}. \tag{2}$$

Throughout the paper, we assume that system (1) performs the same task repeatedly for J number of times. Each task execution is referred to as *iteration.* Our goal is to design a controller that, at each iteration j, solves the finite horizon robust optimal control problem:

$$\begin{aligned} & V^{j,\star}(x_S) = \\ & \min_{u_0^j, u_1^j(\cdot),\ldots} \quad \sum_{t=0}^{T-1} \ell\left(\bar{x}_t^j, u_t^j\left(\bar{x}_t^j\right)\right) \\ & \text{s.t.} \quad x_{t+1}^j = Ax_t^j + Bu_t^j(x_t^j) + w_t^j, \\ & \qquad \bar{x}_{t+1}^j = A\bar{x}_t^j + Bu_t^j(\bar{x}_t^j), \\ & \qquad H_x x_t^j \leq h_x, \\ & \qquad H_u u_t^j \leq h_u, \\ & \qquad \forall w_t^j \in \mathbb{W}, \\ & \qquad x_0^j = x_S,\ t = 0,1,\ldots,(T-1), \end{aligned} \tag{3}$$

where x_t^j, u_t^j and w_t^j denote the realized system state, control input and disturbance at time t of the j^{th} iteration respectively, and $(\bar{x}_t^j, u_t^j(\bar{x}_t^j))$ denote the disturbance-free nominal state and corresponding nominal input. Notice that (3) minimizes the nominal cost over a time horizon of length $T \gg 0$ in any j^{th} iteration with $j \in [J]$. Here we use $[J]$ to denote the set $\{1,2,\ldots,J\}$. We point out that, as system (1) is uncertain, the optimal control problem (3) consists of finding $[u_0^j, u_1^j(\cdot), u_2^j(\cdot), \ldots]$, where $u_t^j : \mathbb{R}^d \ni x_t^j \mapsto u_t^j = u_t^j(x_t^j) \in \mathbb{R}^m$ are state feedback policies. As task duration $T \gg 0$, for computational tractability we try to approximate a solution to the optimal control problem (3), by solving a simpler constrained optimal control problem with prediction horizon $N \ll T$ in a receding horizon fashion.

In this work, we consider the support $\mathbb{W}$ of disturbance w_t^j to be an unknown convex and compact set. We estimate $\mathbb{W}$ using observed disturbance samples. At the start of iteration j, the estimated support is denoted by $\hat{\mathbb{W}}^j$.

3. ITERATIVE MPC PROBLEM

The MPC controller solves a finite horizon optimal control problem at each time step t in the j^{th} iteration. Since the disturbance support $\mathbb{W}$ is unknown and is estimated with $\hat{\mathbb{W}}^j$ built from data, robust satisfaction of (2) along the iteration is not guaranteed. This implies that the closed loop task execution might fail. We will formally define this notion of *failure* after defining the closed loop controller in this section.

We attempt to design a robust MPC controller in the j^{th} iteration with our best estimate $\hat{\mathbb{W}}^j$ of disturbance support $\mathbb{W}$, by solving the following optimal control problem:

$$\begin{aligned} & V_{t\to t+N}^{\text{MPC},j}(x_t^j, \hat{\mathbb{W}}^j, \hat{\mathcal{X}}_N^j) := \\ & \min_{U_t^j(\cdot)} \quad \sum_{k=t}^{t+N-1} \ell(\bar{x}_{k|t}^j, v_{k|t}^j) + Q(\bar{x}_{t+N|t}^j) \\ & \text{s.t} \quad x_{k+1|t}^j = Ax_{k|t}^j + Bu_{k|t}^j + w_{k|t}^j, \\ & \qquad \bar{x}_{k+1|t}^j = A\bar{x}_{k|t}^j + Bv_{k|t}^j, \\ & \qquad u_{k|t}^j = \sum_{l=t}^{k-1} M_{k,l|t}^j w_{l|t}^j + v_{k|t}^j, \\ & \qquad H_x x_{k|t}^j \leq h_x, \\ & \qquad H_u u_{k|t}^j \leq h_u, \\ & \qquad x_{t+N|t}^j \in \hat{\mathcal{X}}_N^j, \\ & \qquad \forall w_{k|t}^j \in \hat{\mathbb{W}}^j, \\ & \qquad \forall k = \{t,\ldots,t+N-1\}, \\ & \qquad x_{t|t}^j = \bar{x}_{t|t}^j = x_t^j, \end{aligned} \tag{4}$$

where in the j^{th} iteration, x_t^j is the measured state at time t, $x_{k|t}^j$ is the prediction of state at time k, obtained by applying predicted input policies $U_t^j(\cdot) = [u_{t|t}^j, \ldots, u_{k-1|t}^j]$ to system (1) and $\{\bar{x}_{k|t}^j, v_{k|t}^j\}$ with $v_{k|t}^j = u_{k|t}^j(\bar{x}_{k|t}^j)$ denote the disturbance-free nominal state and corresponding input respectively. The MPC controller minimizes the cost over the predicted disturbance free nominal trajectory $\left\{\{\bar{x}_{k|t}^j, v_{k|t}^j\}_{k=t}^{t+N-1}, \bar{x}_{t+N|t}^j\right\}$, which comprises of the positive definite stage cost $\ell(\cdot,\cdot)$, and the terminal cost $Q(\cdot)$. Notice, the above uses affine disturbance feedback parametrization (Goulart et al. (2006)) of input policies. We use state feedback to construct terminal set $\hat{\mathcal{X}}_N^j = \{x \in \mathbb{R}^d : \hat{Y}^j x \leq \hat{z}^j,\ \hat{Y}^j \in \mathbb{R}^{r^j\times d},\ \hat{z}^j \in \mathbb{R}^{r^j}\}$, which is the $(T-N)$ step robust reachable set (Borrelli et al., 2017, Chapter 10) to set of state constraints in (2), obtained with a state feedback controller $u = Kx$, dynamics (1) and constraints (2). After solving (4), in closed loop, we apply

$$u_t^j = v_{t|t}^{j,\star} \tag{5}$$

to system (1). We then resolve the problem (4) again at the next $(t+1)$-th time step, yielding a receding horizon strategy.

Assumption 1. (Well Posedness). We assume that given an initial state x_S, optimization problem (4) is feasible at all times $0 \leq t \leq T-1$ with true uncertainty support $\hat{\mathbb{W}}^j = \mathbb{W}$ for all iterations $j \in [J]$.

Since $\mathbb{W}$ is unknown and is being estimated with $\hat{\mathbb{W}}^j$ in the j^{th} iteration, we might lose the feasibility of (4) during $0 \leq t \leq T-1$. We formalize this with the following definition:

Definition 2. (State Constraint Failure). A State Constraint Failure at time step t in iteration j is the event

$$[\text{SCF}]_t^j: \ H_x x_t^j > h_x. \tag{6}$$

That is, a State Constraint Failure implies the violation of imposed constraints (2) by system (1) in closed loop with MPC controller (5).

Remark 3. Let $T^j < T$ denote the time step in the j^{th} iteration when a State Constraint Failure occurs. In that case, problem (4) becomes infeasible at T^j. We then stop the j^{th} iteration and update $\hat{\mathbb{W}}^j \overset{\text{update}}{\longrightarrow} \hat{\mathbb{W}}^{j+1}$. When $T^j = T$, it denotes a successful iteration without any State Constraint Failure.

Our aim is not only to keep the probability of State Constraint Failure $[\text{SCF}]_t^j$ low along each iteration, but also to maintain satisfactory controller "performance" during successful iterations (as defined in Remark 3). Let the closed loop cost of a successful iteration j be denoted by

$$\hat{\mathcal{V}}^j(x_S, w^{1:j}) = \sum_{t=0}^{T-1} \ell(x_t^j, v_{t|t}^{j,\star}), \tag{7}$$

where notation $w^{1:j}$ denotes the set $\bigcup_{i=1}^{j} \bigcup_{t=0}^{T-1} w_t^i$. We use the average closed loop cost $\mathbb{E}[\hat{\mathcal{V}}^j(x_S, w^{1:j})]$ to quantify controller performance. The goal is to lower the *performance loss* defined as

$$[\text{PL}]^j = \mathbb{E}[\hat{\mathcal{V}}^j(x_S, w^{1:j})] - \mathbb{E}[\mathcal{V}^\star(x_S, w^{1:j})], \tag{8}$$

where $\mathbb{E}[\mathcal{V}^\star(x_S, w^{1:j})]$ denotes the average closed loop cost of the j^{th} iteration if $\mathbb{W}$ had been known, i.e., $\hat{\mathbb{W}}^j = \mathbb{W}$ for all $j \in [J]$.

In the next section, we introduce two design specifications (D1) and (D2) to formalize this joint focus on lowering probability of State Constraint Failure and maintaining satisfactory controller performance. We then show how the sets $\hat{\mathbb{W}}^j$ are constructed according to these specifications.

4. LEARNING ROBUSTNESS WITH BOUNDED FAILURE

We consider the following design specifications:

(D1) Closed loop MPC control law (5) ensures that system (1) in the j^{th} iteration satisfies a user specified upper bound α on probability of State Constraint Failure (Definition 2),

(D2) Minimize $[\text{PL}]^j$ (as defined in (8)) over all iterations $j \in [J]$ while satisfying (D1).

For satisfaction of (D1) we require,

$$\mathbb{P}(H_x x_t^j > h_x) \leq \alpha. \tag{9}$$

Since the above probability is difficult to compute, we consider an alternative notion of failure in order to upper bound the probability of State Constraint Failure.

Definition 4. (Disturbance Support Failure). We define a Disturbance Support Failure at time step t in iteration j as the event

$$[\text{DSF}]_t^j: \ w_t^j \notin \hat{\mathbb{W}}^j. \tag{10}$$

As the MPC controller (4) is robust to all $w_t^j \in \hat{\mathbb{W}}^j$, we have $[\text{SCF}]_t^j \subseteq [\text{DSF}]_t^j$. Therefore, probability of Disturbance Support Failure is an upper bound for probability of State Constraint Failure, i.e., $\mathbb{P}([\text{SCF}]_t^j) \leq \mathbb{P}([\text{DSF}]_t^j)$. Therefore, we focus on the following specification:

$$\mathbb{P}(w_t^j \notin \hat{\mathbb{W}}^j) \leq \alpha. \tag{11}$$

In the next few sections, we discuss how such sets $\hat{\mathbb{W}}^j$ can be constructed based on disturbance samples observed during the iterative task. For this purpose we make the following assumption.

Assumption 5. We assume that the unknown distribution $\mathcal{P}$ defined in Section 2 belongs to a finite dimensional parametric family $\{\mathcal{P}_\theta : \theta \in \Theta, \Theta \subseteq \mathbb{R}^l\}$.

We next explore how to construct the sets $\hat{\mathbb{W}}^j$ using Assumption 5, so that design specification (D1) is satisfied. For that purpose, we introduce the notion of Confidence Supports which are closely related to the notion of confidence intervals in classical statistics. Subsequently in Section 4.2 we present our algorithm.

4.1 Confidence Support of a Distribution

Consider i.i.d. samples $Z_{1:n} = (Z_1, \ldots, Z_n)$ from a distribution $\mathcal{P}_\theta$ parametrized by $\theta \in \mathbb{R}$, i.e., $Z_i \overset{\text{iid}}{\sim} \mathcal{P}_\theta$. In classical statistics, the notion of confidence interval provides a convenient way to characterize the uncertainty of parameter θ from the observed samples $Z_{1:n}$.

Definition 6. (Confidence Interval). A set $\mathcal{C}(Z_{1:n})$ is a $(1-\alpha)$-confidence interval for the parameter θ of distribution $\mathcal{P}_\theta$ if

$$\mathbb{P}(\theta \notin \mathcal{C}(Z_{1:n})) \leq \alpha. \tag{12}$$

If $\theta \in \mathbb{R}^d$, and $d > 1$, then the term *confidence region* is used for the set $\mathcal{C}(Z)$ as defined above.

Remark 7. Note that $\mathcal{C}(Z)$ is a random set as it is a function of the collection of random samples $Z_{1:n}$, whereas θ is an unknown deterministic parameter.

Definition 8. (Confidence Support). A set $\mathcal{S}(Z_{1:n})$ is a $(1-\alpha)$-Confidence Support of a distribution $\mathcal{P}_\theta$ with support $\mathbb{S}_\theta$ if

$$\mathbb{P}(\mathbb{S}_\theta \subseteq \mathcal{S}(Z_{1:n})) \geq 1 - \alpha, \tag{13}$$

i.e., $\mathcal{S}(Z_{1:n})$ contains the support $\mathbb{S}_\theta$ of $\mathcal{P}_\theta$ with probability greater than or equal to $(1-\alpha)$.

Using the above notion of Confidence Supports, we now demonstrate how the disturbance support estimates $\hat{\mathbb{W}}^j$ (as defined in iterative MPC problem (4)) can be computed based on observed disturbance samples.

4.2 Computing $\hat{\mathbb{W}}^j$

Consider i.i.d. disturbance samples $w_t^j \sim \mathcal{P}_\theta,\ \theta \in \mathbb{R}^l$ with support $\mathbb{W}$. Let $w_t^j(q)$ denote the q^{th} element of $w_t^j \in \mathbb{R}^d$. Recall that $[d]$ denotes the set $\{1, 2, \dots, d\}$. We make the following simplifying assumption:

Assumption 9. The elements of random vector $w_t^i \in \mathbb{R}^d$ are independently distributed,

$$w_t^j(q) \sim \mathcal{P}^q_{\theta_q},\ q \in [d], \tag{14}$$

where $\theta = (\theta_1, \dots, \theta_d)$ and $\{\mathcal{P}^q_{\theta_q} : \theta_q \in \Theta_q,\ \Theta_q \subset \mathbb{R}^{l/d}\}$ is the corresponding parametric family for the q^{th} element.

At the start of the j^{th} iteration, the collection of samples $w^{1:j-1}$ would have been observed. As the uncertainty distribution $\mathcal{P}_\theta$ is completely specified by θ, we can compute a $(1-\alpha)$-Confidence Support $\hat{\mathbb{W}}^j\big(w^{1:j-1}\big)$ by computing confidence regions for the individual parameters $(\theta_1, \dots, \theta_d)$. Note that the confidence regions and supports are functions of the observed disturbance samples $w^{1:j-1}$. For notational convenience, we represent such sets without explicitly showing this dependence.

Lemma 10. Let $\hat{\Theta}^j_q$ be a $(1-\alpha_q)$-confidence region for θ_q. Consider $\hat{\mathbb{W}}^j_q = \bigcup_{\bar{\theta}_q \in \hat{\Theta}^j_q} \text{Supp}(\mathcal{P}^q_{\bar{\theta}_q})$, where $\text{Supp}(\mathcal{P}^q_{\bar{\theta}_q})$ denotes the support of distribution $\mathcal{P}^q_{\bar{\theta}_q}$. Then, $\hat{\mathbb{W}}^j = \hat{\mathbb{W}}^j_1 \times \cdots \times \hat{\mathbb{W}}^j_d$ is a $(1 - \sum_q \alpha_q)$-Confidence Support of $\mathcal{P}_\theta$.

Proof. By definition, $\mathbb{W} = \text{Supp}(\mathcal{P}^1_{\theta_1}) \times \cdots \times \text{Supp}(\mathcal{P}^d_{\theta_d})$. As $\hat{\mathbb{W}}^j = \hat{\mathbb{W}}^j_1 \times \cdots \times \hat{\mathbb{W}}^j_d$, we have

$$\begin{aligned}
\mathbb{P}(\mathbb{W} \not\subseteq \hat{\mathbb{W}}^j) &= \mathbb{P}(\bigcup_{q=1}^{d} \text{Supp}(\mathcal{P}^q_{\theta_q}) \not\subseteq \hat{\mathbb{W}}^j_q) \\
&= \mathbb{P}(\bigcup_{q=1}^{d} \theta_q \notin \hat{\Theta}^j_q), \\
&\le \sum_{q=1}^{d} \mathbb{P}(\theta_q \notin \hat{\Theta}^j_q), \qquad (15) \\
&\le \sum_{q=1}^{d} \alpha_q, \qquad (16)
\end{aligned}$$

where (15) follows from the union bound and (16) follows from $\hat{\Theta}^j_q$ being a $(1-\alpha_q)$-confidence region for θ_q.

Thus, a $(1-\alpha)$-Confidence Support can be constructed using $(1-\alpha_q)$-confidence regions by setting $\alpha_q = \frac{\alpha}{d}$. We now show that such a Confidence Support has a bounded probability of Disturbance Support Failure, as defined in (10).

Proposition 11. Let $\hat{\mathbb{W}}^j$ be a $(1-\alpha)$-Confidence Support of $\mathcal{P}_\theta$ computed using samples $w^{1:j-1}$. Then, we have

$$\mathbb{P}(w_t^j \notin \hat{\mathbb{W}}^j) \le \alpha,\ 0 \le t \le T-1. \tag{17}$$

Proof. Note that both w_t^j and $\hat{\mathbb{W}}^j$ are random. Using the law of total probability, we have

$$\begin{aligned}
\mathbb{P}(w_t^j \notin \hat{\mathbb{W}}^j) &= \mathbb{P}(w_t^j \notin \hat{\mathbb{W}}^j | \mathbb{W} \subseteq \hat{\mathbb{W}}^j)\mathbb{P}(\mathbb{W} \subseteq \hat{\mathbb{W}}^j) \\
&\quad + \mathbb{P}(w_t^j \notin \hat{\mathbb{W}}^j | \mathbb{W} \not\subseteq \hat{\mathbb{W}}^j)\mathbb{P}(\mathbb{W} \not\subseteq \hat{\mathbb{W}}^j), \\
&= \mathbb{P}(w_t^j \notin \hat{\mathbb{W}}^j | \mathbb{W} \not\subseteq \hat{\mathbb{W}}^j)\mathbb{P}(\mathbb{W} \not\subseteq \hat{\mathbb{W}}^j), \\
&\le \mathbb{P}(\mathbb{W} \not\subseteq \hat{\mathbb{W}}^j), \qquad (18) \\
&\le \alpha, \qquad (19)
\end{aligned}$$

where (19) follows from the fact that $\hat{\mathbb{W}}^j$ is a $(1-\alpha)$-Confidence Support of $\mathcal{P}_\theta$.

Remark 12. As long as the confidence regions $\hat{\Theta}^j_q$ converge to the true parameter θ_q in probability, the Confidence Supports asymptotically converge to the true uncertainty support, i.e., $\hat{\mathbb{W}}^j \to \mathbb{W}$ in probability. The MPC controller (5) thus asymptotically learns to satisfy (2) *robustly.*

4.3 The LRBF Algorithm

We present our Learning Robustness from Bounded Failure (LRBF) algorithm which uses Confidence Supports $\hat{\mathbb{W}}^j$ from Section 4.2 in MPC optimization problem (4). This guarantees satisfaction of (9) (i.e., design requirement (D1)) by system (1) in closed loop with controller (5).

Algorithm 1 Learning Robustness with Bounded Failure (LRBF)

Inputs: $\mathbb{Z}, \hat{\mathbb{W}}^1, x_S$.
for $j = 2, \dots, J$ **do**
Computing Confidence Support $\hat{\mathbb{W}}^j$
  for $q = 1, \dots, d$ **do**
    Compute $(1 - \frac{\alpha}{d})$-confidence region $\hat{\Theta}^j_q$ for θ_q
    Compute $\hat{\mathbb{W}}^j_q = \cup_{\bar{\theta}_q \in \hat{\Theta}_q} \text{Supp}(\mathcal{P}^q_{\bar{\theta}_q})$
  Set $\hat{\mathbb{W}}^j = \hat{\mathbb{W}}^j_1 \times \cdots \times \hat{\mathbb{W}}^j_d$
Solving MPC problem (4) *using* $\hat{\mathbb{W}}^j$
  for $t = 0, 1, \dots, T-1$ **do**
    Apply $v^{j,\star}_{t|t}$ from (5) with $\hat{\mathbb{W}}^j$ as uncertainty set

We assume that for all iterations $j \in [J]$, at time step $t = 0$, MPC problem (4) is feasible with disturbance supports $\hat{\mathbb{W}}^j$ constructed in Algorithm 1. In case such an assumption is not satisfied, $\hat{\mathbb{W}}^j$ can be scaled down (for eg., by increasing α).

5. NUMERICAL SIMULATIONS

In this section we find approximate solutions to the following iterative optimal control problem in receding horizon:

$$\begin{aligned}
&V^{j,\star}(x_S) = \\
&\min_{u_0^j, u_1^j(\cdot), \dots} \sum_{t=0}^{T-1} 10 \left\| \bar{x}_t^j - x_{\text{ref}} \right\|_2^2 + 2 \left\| u_t^j(\bar{x}_t^j) \right\|_2^2 \\
&\text{s.t.} \\
&\qquad x_{t+1}^j = A x_t^j + B u_t^j(x_t^j) + w_t^j, \\
&\qquad \bar{x}_{t+1}^j = A \bar{x}_t^j + B u_t^j(\bar{x}_t^j), \\
&\qquad \begin{bmatrix} -30 \\ -30 \\ -40 \end{bmatrix} \le \begin{bmatrix} x_t^j \\ u_t^j(x_t^j) \end{bmatrix} \le \begin{bmatrix} 30 \\ 30 \\ 40 \end{bmatrix},\ \forall w_t^j \in \mathbb{W}, \\
&\qquad x_0^j = x_S,\ t = 0, 1, \dots, T-1.
\end{aligned}$$

We consider two parametric distributions:

$$\mathcal{P}^q_{\theta_q} = \text{Unif}(-3,3), \tag{20a}$$

$$\mathcal{P}^q_{\theta_q} = \mathcal{N}_{\text{trunc}}(0,1,3), \tag{20b}$$

with $q \in \{1,2\}$. In both cases, $\mathbb{W} = [-3,3] \times [-3,3]$. The methods to compute the Confidence Supports for these distributions are discussed in (Bujarbaruah et al., 2019, Section 4.5). System matrices $A = \begin{bmatrix} 1.2 & 1.3 \\ 0 & 1.5 \end{bmatrix}$ and $B = [0,1]^\top$ are known. We solve the above optimization problem with the initial state $x_S = [0,0]^\top$ and reference point $x_{\text{ref}} = [27,27]^\top$ for task duration $T = 20$ steps over $J = 30$ iterations. Algorithm 1 is implemented with a control horizon of $N = 4$, and the feedback gain K in (5) is chosen to be the optimal LQR gain for system $x^+ = (A + BK)x$ with parameters $Q_{\text{LQR}} = 10I_{2\times 2}$ and $R_{\text{LQR}} = 2$. The goal is to show:

- Design specification (D1) is satisfied. Consequently, a lower probability of Disturbance Support Failure across all iterations using support $\hat{\mathbb{W}}^j$ from Algorithm 1, compared to that from the convex hull support estimate $C^{\text{hull}}(w^{1:j-1})$.
- The performance loss $[\text{PL}]^j$ rapidly approaches 0 within the first few iterations. However, in the initial iterations, there is a significant trade-off between a desired upper bound α on probability of State Constraint Failure and average closed loop cost $\mathbb{E}[\hat{\mathcal{V}}^j(x_S, w^{1:j})]$ (defined in (7)). That is, lower the upper bound α, higher is the average closed loop cost in the initial iterations. This suggests the need for tailoring the confidence level $(1-\alpha)$ in Algorithm 1 according to the application at hand.

5.1 Bounding the Probability of Failure (D1)

In this section, we demonstrate satisfaction of design specification (D1) by Algorithm 1 and compare the probability of Disturbance Support Failure $\mathbb{P}(w^j_t \notin \hat{\mathbb{W}}^j)$ for any timestep t in the j^{th} iteration, with $\hat{\mathbb{W}}^j$ obtained using Algorithm 1 and $\hat{\mathbb{W}}^j = C^{\text{hull}}(w^{1:j-1})$. This probability is estimated by averaging over 100 Monte Carlo draws of disturbance samples $w^{1:J}$, i.e.,

$$\mathbb{P}(w^j_t \notin \mathbb{W}^j) \approx \frac{1}{100}\sum_{\tilde{m}=1}^{100} (\mathbf{1}_{\mathcal{F}}(w^j_t))^{\star\tilde{m}},$$

where

$$(\mathbf{1}_{\mathcal{F}}(w^j_t))^{\star\tilde{m}} = \begin{cases} 1, & \text{if } w^j_t \notin (\hat{\mathbb{W}}^j)^{\star\tilde{m}} | (w^{1:j-1})^{\star\tilde{m}}, \\ 0, & \text{otherwise}, \end{cases}$$

and $(\cdot)^{\star\tilde{m}}$ represents the $\tilde{m}^{\text{th}}$ Monte Carlo sample. Fig. 1 shows this comparison for uniformly distributed disturbance (20a). Using LRBF to construct Confidence Supports $\hat{\mathbb{W}}^j$ allows for lowering $\mathbb{P}(w^j_t \notin \hat{\mathbb{W}}^j)$, i.e., probability of $[\text{DSF}]^j_t$ as defined in (10) below a user specified bound α, as opposed to simply utilizing $\hat{\mathbb{W}}^j = C^{\text{hull}}(w^{1:j-1})$. We plot the probability of $[\text{DSF}]^j_t$ for 2 different values of $\alpha = 0.05$ and $\alpha = 0.70$. We see that for $\alpha = 0.05$ the probability of $[\text{DSF}]^j_t$ with LRBF is on average 94% smaller than that from the convex hull support estimate for all iterations $j \in [30]$. Similarly for $\alpha = 0.70$, the probability of $[\text{DSF}]^j_t$ is on average 61% lower than that with the convex hull support estimate across all $j \in [30]$.

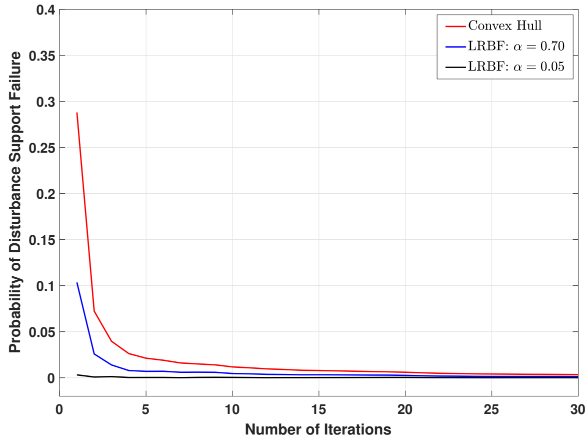

Fig. 1. Probability of Disturbance Support Failure vs Iteration Number for Uniformly Distributed Disturbance on $\mathbb{W}$.

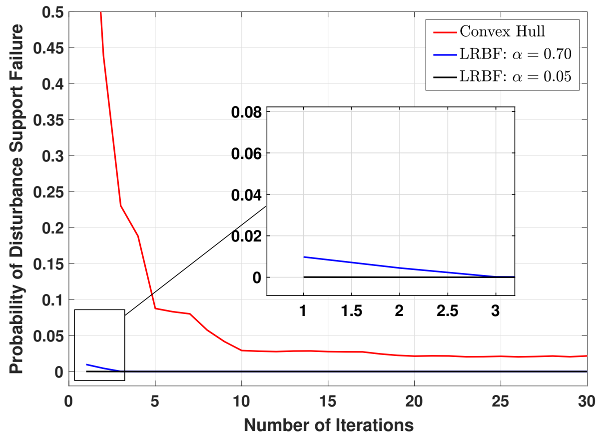

Fig. 2. Probability of Disturbance Support Failure vs Iteration Number for Truncated Normal Distribution of Disturbance on $\mathbb{W}$.

The same trend is seen in Fig. 2 for truncated normal distribution (20b), where probability of $[\text{DSF}]^j_t$ is at least 99% and 96% lower than convex hull support estimate for $\alpha = 0.05$ and $\alpha = 0.70$ respectively until iteration $j = 3$, and reaches a value of 0 for both values of α afterwards. The above trend in probability of $[\text{DSF}]^j_t$ is explained by Proposition 11, which relates the desired confidence $(1-\alpha)$ for support $\hat{\mathbb{W}}^j$ to the probability of $[\text{DSF}]^j_t$. Moreover, from Fig. 1 and Fig. 2 we see that in practice probability of $[\text{DSF}]^j_t$ is always at least $60\% - 80\%$ lower than corresponding chosen α. This highlights satisfaction of (D1) and also the conservatism in Proposition 11 arising from the upper bound in (18).

5.2 Performance Loss Reduction Over Iterations

In Fig. 3 and Fig. 4, we approximate the average closed loop cost $\mathbb{E}[\hat{\mathcal{V}}^j(x_S, w^{1:j})]$ of the j^{th} iteration by taking an empirical average over 100 Monte Carlo draws of $w^{1:J}$ as,

$$\mathbb{E}[\hat{\mathcal{V}}^j(x_S, w^{1:j})] \approx \frac{1}{100} \sum_{\tilde{m}=1}^{100} \hat{\mathcal{V}}^j(x_S, (w^{1:j})^{\star\tilde{m}}), \quad (21)$$

for $\alpha = 0.05$, and $\alpha = 0.70$. The cost values are normalized by $\mathcal{V}^\star(x_S)$, which denotes the empirical average closed loop cost of the j^{th} iteration if $\mathbb{W}$ had been known, i.e., $\hat{\mathbb{W}}^j = \mathbb{W}$. For both cases of α, we see that in Fig. 3 and Fig. 4 the average closed loop cost rapidly approaches $\mathcal{V}^\star(x_S)$. For (20a) in Fig. 3, cost (21) approaches to within 0.5% of $\mathcal{V}^\star(x_S)$ after just 5 iterations whereas for (20b) in Fig. 4, it is within 3% of $\mathcal{V}^\star(x_S)$ in the same duration.

However, the average closed loop cost incurred in earlier iterations has a trade-off with desired α. This trade-off is also highlighted in Fig. 3 and Fig. 4 for (20a) and (20b) respectively. We see from Fig. 3 and Fig. 4 that for lower value of probability of $[\text{SCF}]_t^j$ with $\alpha = 0.05$, we pay a maximum of 13% higher average closed loop cost for (20a), and a maximum of 10% higher average closed loop cost for (20b) compared to $\mathcal{V}^\star(x_S)$ until iteration $j = 5$. Allowing for higher probability of $[\text{SCF}]_t^j$ with $\alpha = 0.70$ proves to be cost-efficient, where we only pay a maximum of 0.3% higher average closed loop cost for (20a), and a maximum of 4% higher average closed loop cost for (20b) compared to $\mathcal{V}^\star(x_S)$ in the same duration. This essentially reflects the key trade-off between specifications (D1) and (D2) in the initial iterations. Thus, the upper bound α of $[\text{SCF}]_t^j$ must be chosen in an application-specific manner.

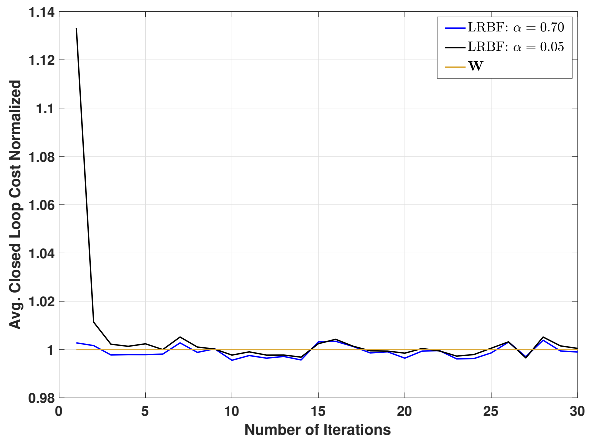

Fig. 3. Normalized Average Closed Loop Cost (21): Uniform Disturbance.

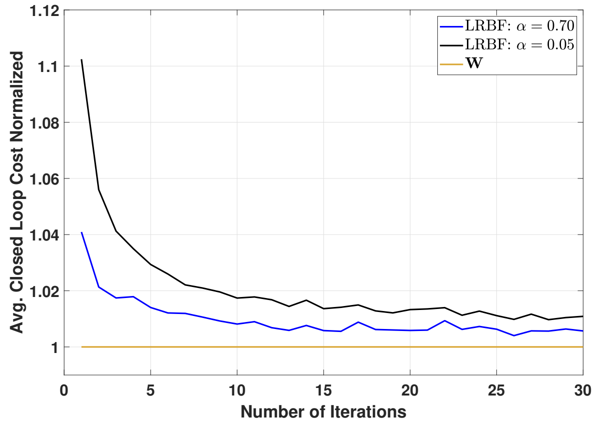

Fig. 4. Normalized Average Closed Loop Cost (21): Truncated Normal Disturbance.

6. ACKNOWLEDGEMENT

We thank Hamidreza Tavafoghi for helpful discussions and reviews.

REFERENCES

Borrelli, F., Bemporad, A., and Morari, M. (2017). *Predictive control for linear and hybrid systems*. Cambridge University Press.

Bujarbaruah, M., Shetty, A., Poolla, K., and Borrelli, F. (2019). Learning robustness with bounded failure: An iterative MPC approach. *arXiv preprint arXiv:1911.09910*.

Goulart, P.J., Kerrigan, E.C., and Maciejowski, J.M. (2006). Optimization over state feedback policies for robust control with constraints. *Automatica*, 42(4), 523–533.

Hewing, L. and Zeilinger, M.N. (2017). Cautious model predictive control using gaussian process regression. *arXiv preprint arXiv:1705.10702*.

Köhler, J., Andina, E., Soloperto, R., Müller, M.A., and Allgöwer, F. (2019). Linear robust adaptive model predictive control : Computational complexity and conservatism. *arXiv preprint arXiv:1909.01813*.

Lorenzen, M., Cannon, M., and Allgöwer, F. (2019). Robust MPC with recursive model update. *Automatica*, 103, 461 – 471.

Mayne, D.Q., Rawlings, J.B., Rao, C.V., and Scokaert, P.O. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, 36(6), 789–814.

Rosolia, U. and Borrelli, F. (2017). Learning model predictive control for iterative tasks. a data-driven control framework. *IEEE Transactions on Automatic Control*.

Rosolia, U., Zhang, X., and Borrelli, F. (2018). Data-driven predictive control for autonomous systems. *Annual Review of Control, Robotics, and Autonomous Systems*, 1, 259–286.

Soloperto, R., Müller, M.A., Trimpe, S., and Allgöwer, F. (2018). Learning-based robust model predictive control with state-dependent uncertainty. In *IFAC Conference on Nonlinear Model Predictive Control*. Madison, Wisconsin, USA.

Tanaskovic, M., Fagiano, L., Novara, C., and Morari, M. (2017). Data-driven control of nonlinear systems: An on-line direct approach. *Automatica*, 75, 1–10.

Tanaskovic, M., Fagiano, L., Smith, R., and Morari, M. (2014). Adaptive receding horizon control for constrained MIMO systems. *Automatica*, 50(12), 3019–3029.

Tempo, R., Calafiore, G., and Dabbene, F. (2012). *Randomized algorithms for analysis and control of uncertain systems: with applications*. Springer Science & Business Media.

Zhang, X., Margellos, K., Goulart, P., and Lygeros, J. (2013). Stochastic model predictive control using a combination of randomized and robust optimization. In *IEEE Conference on Decision and Control (CDC)*. Florence, Italy.