Robust Model Predictive Control for Non-Linear Systems with Input and State Constraints Via Feedback Linearization

Yash Vardhan Pant, Houssam Abbas, Rahul Mangharam

Abstract— **Robust predictive control of non-linear systems under state estimation errors and input and state constraints is a challenging problem, and solutions to it have generally involved solving computationally hard non-linear optimizations. Feedback linearization has reduced the computational burden, but has not yet been solved for robust model predictive control under estimation errors and constraints. In this paper, we solve this problem of robust control of a non-linear system under bounded state estimation errors and input and state constraints using feedback linearization. We do so by developing robust constraints on the feedback linearized system such that the non-linear system respects its constraints. These constraints are computed at run-time using online reachability, and are linear in the optimization variables, resulting in a Quadratic Program with linear constraints. We also provide robust feasibility, recursive feasibility and stability results for our control algorithm. We evaluate our approach on two systems to show its applicability and performance.**

I. Introduction

In this paper we are concerned with the problem of controlling nonlinear dynamical systems S of the form $\dot{x} = f(x) + G(x)u$ under state and input constraints, and subject to errors in the state estimate. This problem is formulated as

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & l(\mathbf{x},\mathbf{u}) \\ \text{s.t. } \dot{x} &= f(x) + G(x)u \\ x &\in X, u \in U \end{aligned} \tag{1}$$

where $l(\mathbf{x},\mathbf{u})$ is a cost function whose minimization over the state and input trajectories $\mathbf{x}$ and $\mathbf{u}$ ensures stability of the system. Sets $X \subset \mathbb{R}^{n_x}$ and $U \subset \mathbb{R}^{n_u}$ encode constraints on the state (e.g., safety) and the input. The input $u = u(\hat{x})$ is a function of a state estimate that in general differs from the true state of the system.

Whereas Model Predictive Control (MPC) is a widely used technique for controlling linear systems with constraints, its application to nonlinear systems involves the repeated solution of generally non-quadratic, non-convex optimizations. Various approaches for solving (or approximately solving) the optimizations and their trade-offs are reviewed in [1]. Another approach is to first *feedback linearize* the system S [2]: namely, the applied control $u = u(x, v)$ is designed in such a way that the resulting closed-loop dynamics S_{fl} are now *linear*:

$$S_{fl} : \dot{z} = Az + Bv$$

The input v to the linearized dynamics can now be computed so as to optimize system performance and ensure stability. The state z of the linearized system S_{fl} is related to the state x of the nonlinear system S via a (system-specific) function T: $z = T(x)$.

Previous work on nonlinear MPC with feedback linearization assumed the state $x(t)$ is perfectly known to the controller at any moment in time [3]. In practice, only a state estimate $\hat{x}(t)$ is available, and $\hat{x}(t) \neq x(t)$. Thus a controller designed to work optimally when operating on the true state x is in general sub-optimal when operating on $\hat{x}$ (any may even lead to instability). Robust MPC (RMPC) has been investigated as a way of handling state estimation errors for linear [4] and nonlinear systems [5], [6], but not via feedback linearization. In particular, for non-linear systems, [5] develops a non-linear MPC with tube like constraints for robust feasibility, but involves solving two (non-convex) optimal control problems. In [6], the authors solve a non-linear Robust MPC through a bi-level optimization that involves solving a non-linear, non-smooth optimization which is challenging. [6] also guarantees a weaker form of recursive feasibility than [4] and what we guarantee in this work. In [7] the authors approximate the non-linear dynamics of a quadrotor by linearizing it around hover and apply the RMPC of [4] to the linearized dynamics. This differs significantly from our approach, where we formulate the RMPC on the *feedback* linearized dynamics directly, and not on the dynamics obtained via Jacobian linearization of the non-linear system. Existing work on MPC via feedback linearization and input/state constraints has also assumed that either T is the identity, which simplifies the subsequent stability and performance analysis [3], or, in the case of uncertainties in the parameters, that there are no state constraints [8]. A non-identity T is problematic when the state is not perfectly known, since the state estimation error $e = \hat{x} - x$ maps to the linearized dynamics via T in non-trivial ways greatly complicating the analysis. In particular, the error bounds for the state estimate in z-space now depend on the current nonlinear state x. One of the complications introduced by feedback linearization is that the bounds on the input ($u \in U$) may become a non-convex state-dependent constraint on the input v to S_{fl}: $V = \{\underline{v}(x, U) \leq v \leq \mathbf{v}(x, U)\}$. In [3] forward reachability is used to provide inner convex approximations to the input set V. A non-identity T increases the computational burden since the non-linear reach set must

*This work was supported by STARnet a Semiconductor Research Corporation program sponsored by MARCO and DARPA, NSF MRI-0923518 and the US Department of Transportation University Transportation Center Program

The Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, U.S.A. {yashpant,habbas,rahulm}@seas.upenn.edu

Preprint submitted to 55th IEEE Conference on Decision and Control.
Received March 15, 2016.

be computed (with an identity T, the feedback linearized reach set is sufficient).

Contributions: In this paper we develop a feedback linearization solution to the above control problem, with state estimation errors, input and state constraints, and non-identity T. To the best of our knowledge, this is the first feedback linearization solution to this problem. The resulting control problem is solved by RMPC with time-varying linear constraint sets.

The paper is organized as follows: in the next section we formulate the feedback linearized control problem. In Sec. III, we describe the RMPC algorithm we use to solve it, and prove that it stabilizes the nonlinear system. Sec. IV shows how to compute the various constraint sets involved in the RMPC formulation, and Sec. V applies our approach to two examples. Sec. VI concludes the paper. An online technical report [9] contains the proof of Thm. 1.

II. Problem Formulation

A common method for control of nonlinear systems is Feedback linearization [2]. Briefly, in feedback linearization, one applies the feedback law $u(x,v) = R(x)^{-1}(-b(x)+v)$ to (1), so that the resulting dynamics, expressed in terms of the transformed state $z = T(x)$, are linear time-invariant:

$$S_{fl} : \dot{z} = A_c z + B_c v \tag{2}$$

By using the remaining control authority in v to control S_{fl}, we can effectively control the non-linear system for, say, stability or reference tracking. T is a diffeomorphism [2] over a domain $D \subset X$. The original and transformed states, x and z, have the same dimension, as do u and v, i.e. $n_x = n_z$ and $n_u = n_v$. Because we are controlling the feedback linearized system, we must find constraint sets Z and V for the state z and input v, respectively, such that $(z,v) \in Z \times V \implies (T^{-1}(z), u(T^{-1}(z), v)) \in X \times U$. This is done in Sec. IV-D. Without loss of generality we assume that the system (1) has no zero dynamics and all states are controllable [2].

For feedback linearizing (1) and for controlling (1), only a periodic state estimate $\hat{x}$ of x is available. This estimate is available periodically every τ time units, so we may write $\hat{x}_k := \hat{x}(k\tau) = x_k + e_k$, where x_k and e_k are sampled state and error respectively. We assume that e_k is in a bounded set E for all k. This implies that the feedback linearized system can be represented in discrete-time: $z_{k+1} = Az_k + Bz_k$.

The corresponding z-space estimate $\hat{z}_k$ is given by $\hat{z}_k = T(\hat{x}_k)$. In general the z-space error $\tilde{e}_k := T(\hat{x}_k) - T(x_k)$ is bounded for every k but does not necessarily lie in E. Let $\widetilde{E}_k$ be the set containing $\tilde{e}_k$: in Sec. IV-C we show how to compute it.

Because the linearizing control operates on the state estimate and not x_k, we add a process noise term to the linearized, discrete-time dynamics. Our system model is therefore

$$z_{k+1} = Az_k + Bv_k + w_k \tag{3}$$

where the noise term w_k lies in the bounded set W for all k. An estimate of W can be obtained using the techniques of this paper. The control problem (1) is therefore replaced by:

$$\begin{aligned} \min_{\mathbf{z},\mathbf{v}} \quad & \sum_{k=0}^{\infty} z_k^T Q z_k + v_k^T R v_k \\ \text{s.t. } z_{k+1} &= Az_k + Bv_k + w_k \\ z_k &\in Z, v_k \in V, w_k \in W \end{aligned} \tag{4}$$

(In Thm. 2, we show that minimizing this cost function implies stability of the system).

It is easy to derive the dynamics of the state estimate $\hat{z}_k$:

$$\begin{aligned} \hat{z}_{k+1} &= z_{k+1} + \tilde{e}_{k+1} \\ &= Az_k + Bv_k + w_k + \tilde{e}_{k+1} \\ &= A\hat{z}_k + Bv_k + (w_k + \tilde{e}_{k+1} - A\tilde{e}_k) \\ &= A\hat{z}_k + Bv_k + \widehat{w}_{k+1} \end{aligned} \tag{5}$$

where $\widehat{w}_{k+1} = w_k + \tilde{e}_{k+1} - A\tilde{e}_k$, and lies in the set $\widehat{W}_{k+1} := W \oplus \widetilde{E}_{k+1} \oplus (-A\widetilde{E}_k)$.

Example 1: Consider the 2D system

$$\dot{x}_1 = \sin(x_2), \dot{x}_2 = -x_1^2 + u \tag{6}$$

The safe set for x is given as $X = \{|x_1| \leq \pi/2, |x_2| \leq \pi/3\}$, and the input set is $U = [-2.75, 2.75]$. For the measurement $y = h(x) = x_1$, the system can be feedback linearized on the domain $D = \{x | \cos(x_2) \neq 0\}$, where it has a relative degree of $\rho = 2$. The corresponding linearizing feedback input is $u = -\tan(x_2) + (\cos(x_2))v$. The feedback linearized system is $\dot{z}_1 = z_2 \, , \dot{z}_2 = v$, where T is given by $z = T((x_1, x_2)) = (x_1, \sin(x_2))$. We can analytically compute the safe set in z-space as $Z = T(X) = \{|z_1| \leq \pi/2, |z_2| \leq 0.8660\}$. □

For a more complicated T, it is not possible to obtain analytical expressions for Z. The computation of Z in this more general case is addressed in Sec. IV-D.

Notation. Given two subsets A, B of $\mathbb{R}^n$, their *Minkowski sum* is $A \oplus B := \{a + b \mid a \in A, b \in B\}$. Their *Pontryagin difference* is $A \ominus B = \{c \in \mathbb{R}^n \mid c + b \in A \; \forall b \in B\}$. Given integers $n \leq m$, $[n:m] := \{n, n+1, \ldots, m\}$.

Assumption. The approach we use applies when X, U, E and W are arbitrary convex polytopes (i.e. bounded intersections of half-spaces). For the sake of simplicity, in this paper we assume they are all hyper-rectangles that contain the origin, i.e. sets of the form $[\underline{a}_1, \overline{a}_1] \times \ldots \times [\underline{a}_n, \overline{a}_n]$, $\underline{a}_i \leq 0 \leq \overline{a}_i \; \forall i$.

III. Robust MPC for the feedback linearized system

Following [4], [10], we formulate a Robust MPC (RMPC) controller of (4) via *constraint restriction.* We outline the idea before providing the technical details. The key idea is to move the effects of estimation error $\tilde{e}_k$ and process noise w_k (the 'disturbances') to the constraints, and work with the nominal (i.e., disturbance-free) dynamics: $\bar{z}_{k+1} = A\bar{z}_k + Bv_k$, $\bar{z}_0 = \hat{z}_0$. Because we would be optimizing over disturbance-free states, we must account for the noise in the constraints. Specifically, rather than require the next (nominal) state $\bar{z}_{k+1}$ to be in Z, we require it to be in the shrunk set $Z \ominus \widehat{W}_{k+1|k} \ominus \widetilde{E}_{k+1|k}$: by definition of Pontryagin difference, this implies that whatever the actual value of the noise $\widehat{w}_{k+1} \in \widehat{W}_{k+1|k}$ and of the estimation error $\tilde{e}_{k+1} \in \widetilde{E}_{k+1|k}$, the actual state z_{k+1} will be in Z. This is repeated over the entire MPC prediction horizon $j = 1, \ldots, N$, with further shrinking at every step. For further steps ($j > 1$), the process noise $\widehat{w}_{k+j|k}$ is propagated through the dynamics, so the shrinking term $\widehat{W}$ is shaped by a stabilizing feedback controller $\bar{z} \mapsto K\bar{z}$. At the final step ($j = N+1$), a terminal constraint is derived using the worst case estimation error set $\widetilde{E}_{max}$ and a global inner approximation for the input constraints, $V_{inner-global}$.

Through this successive constraint tightening we ensure robust safety and feasibility of the feedback linearized system (and hence of the non-linear system). Since we use just the nominal dynamics, and show that the tightened constraints are linear in the state and inputs, we still solve a Quadratic Program (QP) for the RMPC optimization. The difficulty of applying RMPC in our setting is that the amounts by which the various sets are shrunk vary with time because of the time-varying state estimation error, are state-dependent, and involve set computations with the non-convexity preserving mapping T. One of our contributions in this paper is to establish recursive feasibility of RMPC with time-varying constraint sets.

The RMPC optimization $\mathbb{P}_k(\hat{z}_k)$ for solving (4) is:

$$\begin{aligned}
J^*(\bar{z}_k) = \min_{\bar{\mathbf{z}},\mathbf{u}} \sum_{j=0}^{N} &\{\bar{z}_{k+j|k}^T Q \bar{z}_{k+j|k} + v_{k+j|k}^T R v_{k+j|k}\} \\
&+ \bar{z}_{k+N+1|k}^T Q_f \bar{z}_{k+N+1|k} && \text{(7a)} \\
\bar{z}_{k|k} &= \hat{z}_k && \text{(7b)} \\
\bar{z}_{k+j+1|k} &= A\bar{z}_{k+j|k} + Bv_{k+j|k}, j = 0, \ldots, N && \text{(7c)} \\
\bar{z}_{k+j|k} &\in \overline{Z}_{k+j|k},\ j = 0, \ldots, N && \text{(7d)} \\
v_{k+j|k} &\in V_{k+j|k},\ j = 0, \ldots, N-1 && \text{(7e)} \\
p_{N+1} &= [z_{k+N+1|k}, v_{k+N|k}]^T \in P_f && \text{(7f)}
\end{aligned}$$

Here, $\bar{z}$ is the state of the nominal feedback linearized system. The cost and constraints of the optimization are explained below:

- Eq. (7a) shows a cost quadratic in $\bar{z}$ and v, where as usual Q is positive definite and R is positive semi-definite. In the terminal cost term, Q_f is the solution of the Lyapunov equation $Q_f - (A+BK)^T Q_f (A+BK) = Q + K^T R K$. This choice guarantees that the terminal cost equals the infinite horizon cost under a linear feedback control $\bar{z} \mapsto K\bar{z}$ [11].
- Eq. (7b) initializes the nominal state with the current state estimate.
- Eq. (7c) gives the nominal dynamics of the discretized linearized system.
- Eq. (7d) tightens the admissible set of the nominal state by a sequence of shrinking sets.
- Eq. (7e) constrains $v_{k+j|k}$ such that the corresponding $u(x, v)$ is admissible, and the RMPC is recursively feasible.
- Eq. (7f) constrains the final input and nominal state to be within a terminal set P_f.

The details of these sets' definitions and computations are given in Sec. IV.

A. State and Input Constraints for the Robust MPC

The state and input constraints for the RMPC are defined as follows:

The state constraints $\overline{Z}_{k+j|k}$: The tightened state constraints are functions of the error sets $\widetilde{E}_{k+j|k}$ and disturbance sets $\widehat{W}_{k+j|k}$, and defined $\forall\, j = 0, \ldots, N$

$$\overline{Z}_{k+j|k} = Z \ominus_{i=0}^{j-1} (L_i \widehat{W}_{k+(j-i)|k}) \ominus (-\widetilde{E}_{k+j|k}) \qquad (8)$$

(Recall Z is a subset of $T(X)$, $\widehat{W}_{k+j|k}$ and $\widetilde{E}_{k+j|k}$ bound the estimation error and noise, resp., and are formally defined in Sec. IV). The state transition matrix L_j, $\forall j = 0, \ldots, N$ is defined as $L_0 = \mathbb{I}, L_{j+1} = (A+BK)L_j$. The intuition behind this construction was given at the start of this section.

The input constraints $V_{k+j|k}$: $\forall j = 0, ..., N-1$

$$V_{k+j|k} = \underline{V}_{k+j|k} \ominus_{i=0}^{j-1} KL_i \widehat{W}_{k+(j-i)|k} \qquad (9)$$

where $\underline{V}_{k+j|k}$ is an inner-approximation of the set of admissible inputs v at prediction step $j+k|k$, as defined in Sec. IV-B. The intuition behind this construction is similar to that of $\overline{Z}_{k+j|j}$: given the inner approximation $\underline{V}_{k|k}$, it is further shrunk at each prediction step j by propagating forward the noise $\widehat{w}_k$ through the dynamics, and shaped according to the stabilizing feedback law K, following [4].

The terminal constraint P_f: This constrains the extended state $p_k = [\bar{z}_k, v_{k-1}]^T$, and is given by

$$P_f = C_p \ominus \begin{bmatrix} (A+BK)^N \\ K(A+BK)^{N-1} \end{bmatrix} \widehat{W}_{max} \qquad (10)$$

where $\widehat{W}_{max} \subset \mathbb{R}^{n_z}$ is a bounding set on the worst-case disturbance (we show how it's computed in Sec. IV-C), and $C_p \subset \mathbb{R}^{n_z} \times \mathbb{R}^{n_v}$ is an invariant set of the nominal dynamics subject to the stabilizing controller $\bar{z} \mapsto K\bar{z}$, naturally extended to the extended state p: that is, there exists a feedback control law $p \mapsto \widehat{K}p$, such that $\forall p \in C_p$

$$\widehat{A}p + \widehat{B}\widehat{K}p + \widehat{L}_N[\widehat{w}^T, 0^T]^T \in C_p, \forall \widehat{w} \in \widehat{W}_{max} \qquad (11)$$

with $\widehat{A} = \begin{bmatrix} A & 0_{n\times m} \\ 0_{m\times n} & 0_{m\times m} \end{bmatrix}$, $\widehat{B} = \begin{bmatrix} B \\ \mathbb{I}_{m\times m} \end{bmatrix}$, $\widehat{K} = \begin{bmatrix} K & 0_{m\times m} \end{bmatrix}$, $\widehat{L}_N = (\widehat{A} + \widehat{B}\widehat{K})^N$. It is important to note the following:

- The set P_f can be computed offline since it depends on $\widehat{W}_{max}$, $\widetilde{E}_{max}$ and the global inner approximation for the constraints on v, $V_{inner-global}$, all of which can be computed offline.
- If P_f is non-empty, then all intermediate sets that appear in (7) are also non-empty, since P_f shrinks the state and input sets by the maximum disturbances $\widehat{W}_{max}$ and $\widetilde{E}_{max}$. Thus we can tell, before running the system, whether RMPC might be faced with empty constraint sets (and thus infeasible optimizations).
- Note that all constraints are linear.

B. The Control Algorithm

We can now describe the algorithm used for solving (7) by robust receding horizon control.

Algorithm 1 RMPC via feedback linearization

```
Require: System model, $X$, $U$, $E$, $W$
  Offline, compute:
    Initial safe sets $X_0$ and $Z$                                  ▷ Sec. IV-D
    $\widetilde{E}_{max}$, $\widehat{W}_{max}$                        ▷ Sec. IV-C
    $C_p$, $P_f$                                                      ▷ Sec. III-A
  Online:
  if $P_f = \emptyset$ then
      Quit
  else
      for $k = 1, 2, \ldots$ do
          Get estimate $\hat{x}_k$, compute $\hat{z}_k = T(\hat{x}_k)$
          Compute $\underline{V}_{k+j|k}$, $\widetilde{E}_{k+j|k}$, $\widehat{W}_{k+j|k}$   ▷ Sec. IV-B
          Compute $\overline{Z}_{k+j|k}$, $V_{k+j|k}$                    ▷ Sec. III-A
          $(v^*_{k|k}, \ldots, v^*_{k+N|k})$ = Solution of $\mathbb{P}_k(\hat{z}_k)$
          $v_k = v^*_{k|k}$
          Apply $u_k = R(\hat{x}_k)^{-1}[b(\hat{x}_k) + v_k]$ to plant
      end for
  end if
```

C. Robust Feasibility and Stability

We are now ready to state the main result of this paper: namely, that the RMPC of the feedback linearized system (7) is feasible at all time steps if it starts out feasible, and that it stabilizes the nonlinear system, for all possible values of the state estimation error and feedback linearization error.

Theorem 1 (Robust Feasibility): If at some time step $k_0 \geq 0$, the RMPC optimization $\mathbb{P}_{k_0}(\hat{z}_{k_0})$ is feasible, then all subsequent optimizations $\mathbb{P}_k(\hat{z}_k) k > k_0$ are also feasible. Moreover, the nonlinear system (1) controlled by algorithm 1 and subject to the disturbances (E, W) satisfies its state and input constraints at all times $k \geq k_0$.
The proof is in the online report [9].

Theorem 2 (Stability): Given an equilibrium point $x_e \in X_0 \subset T^{-1}(Z)$ of the nonlinear dynamics (1), Algorithm 1 stabilizes the nonlinear system to x_e.

Proof: Let T be the diffeomorphism mapping x to z from feedback linearization. By a change of variables $z' = z - T(x_e)$, stabilizing the linear dynamics (with state z') to 0 implies stabilizing the nonlinear dynamics to x_e. Recall that Q and Q_f of (7) are positive definite and that R is positive semi-definite, so that the optimal cost $J^*(\bar{z}_k)$ is a positive definite function of $\bar{z}_k$, and that the terminal weight in (7) is equivalent to the infinite horizon cost (by our choice of Q_f). Finally Thm. 1 guarantees that the tail of the input sequence computed at k is admissible at time $k+1$. Therefore it is a standard result that the optimal cost $J^*(\bar{z}_k)$ is non-increasing in k and that 0 is a stable equilibrium for the closed-loop linear system (e.g., see [11]). Moreover the system converges to 0 from anywhere in Z. Therefore x_k converges to x_e from anywhere in $X_0 \subset T^{-1}(Z)$. ■

IV. SET DEFINITIONS FOR THE RMPC

Algorithm 1 and the problem $\mathbb{P}_k(\hat{z}_k)$ (7) use a number of constraint sets to ensure recursive feasibility of the successive RMPC optimizations, namely: inner approximations of the admissible input sets $\underline{V}_{k+j|k}$, bounding sets for the (T-mapped) estimation error $\widetilde{E}_{k+j|k}$, bounding sets for the process noise $\widehat{W}_{k+j|k}$, and the largest error and noise sets $\widetilde{E}_{max}$ and $\widehat{W}_{max}$. In this section we show how these sets are defined and computed.

A. Approximating the reach set of the nonlinear system

First we show how to compute an outer-approximation of the j-step reach set of the nonlinear system, starting at time k, $\mathbf{X}_{k+j|k}$. This is needed for computing $\underline{V}_{k+j|k}$ and $\widetilde{E}_{k+j|k}$.

In all but the simplest systems, forward reachable sets cannot be computed exactly. To approximate them we may use a reachability tool for nonlinear systems like RTreach [12]. A reachability tool computes an outer-approximation of the reachable set of a system starting from some set $\mathcal{X} \subset X$, subject to inputs from a set U, for a duration $T \geq 0$. Denote this approximation by $RT_{=T}(\mathcal{X}, U)$, so $x(T) \in RT_{=T}(\mathcal{X}, U)$ for all T, $x(0) \in \mathcal{X}$ and $u : [0, T] \rightarrow U$.

At time k, the state estimate $\hat{x}_k$ is known. Therefore $x_k = \hat{x}_k - e_k \in \hat{x}_k \oplus (-E) := X_{k|k}$. Propagating $X_{k|k}$ forward one step through the continuous-time nonlinear dynamics yields $X_{k+1|k}$, which is outer-approximated by $RT_{=T}(X_{k|k}, U)$. The state estimate that the system will receive at time $k+1$ is therefore bound to be in the set $RT_{=T}(X_{k|k}, U) \oplus E$. Since $0 \in E$, we maintain $X_{k+1|k} \subset RT_{=T}(X_{k|k}, U) \oplus E$. We define the outer-approximate reach set at $k+1$, computed at time k, to be

$$\mathbf{X}_{k+1|k} := RT_{=T}(X_{k|k}, U) \oplus E \oplus (-E)$$

(The reason for adding the extra $-E$ term will be apparent in the proof to Thm. 1).

More generally, for $1 \leq j \leq N$, we define the j-step approximation computed at time k to be

$$\begin{aligned} \mathbf{X}_{k|k} &:= \hat{x}_k \oplus (-E) \\ \mathbf{X}_{k+j|k} &:= RT_{=T}(\mathbf{X}_{k+j-1|k}, U) \oplus E \oplus (-E) \quad (12) \end{aligned}$$

Fig. 1 shows a visualization of this approach. The following holds by construction:

Lemma 3: For any time k and step $j \geq 1$, $X_{k+j|k} \subset \mathbf{X}_{k+j|k}$.

This construction of $\mathbf{X}_{k+j|k}$ permits us to prove recursive feasibility of the RMPC controller, because it causes the constraints of the RMPC problem setup at time $k+1$ to be consistent with the constraints of the RMPC problem setup at time k.

B. Approximating the bounding sets for the input

Given $x \in X$, define the set $V(x) := \{v \in \mathbb{R}^{n_v} \mid u(x) = R^{-1}(x)[b(x) + v] \in U\}$. We assume that there exist functions $\underline{v}_i, \mathbf{v}_i : X \rightarrow \mathbb{R}$ s.t. for any x, $V(x) = \{[v_1, \ldots, v_{n_v}]^T \mid \underline{v}_i(x; U) \leq v_i \leq \mathbf{v_i}(x; U)\}$. Because in

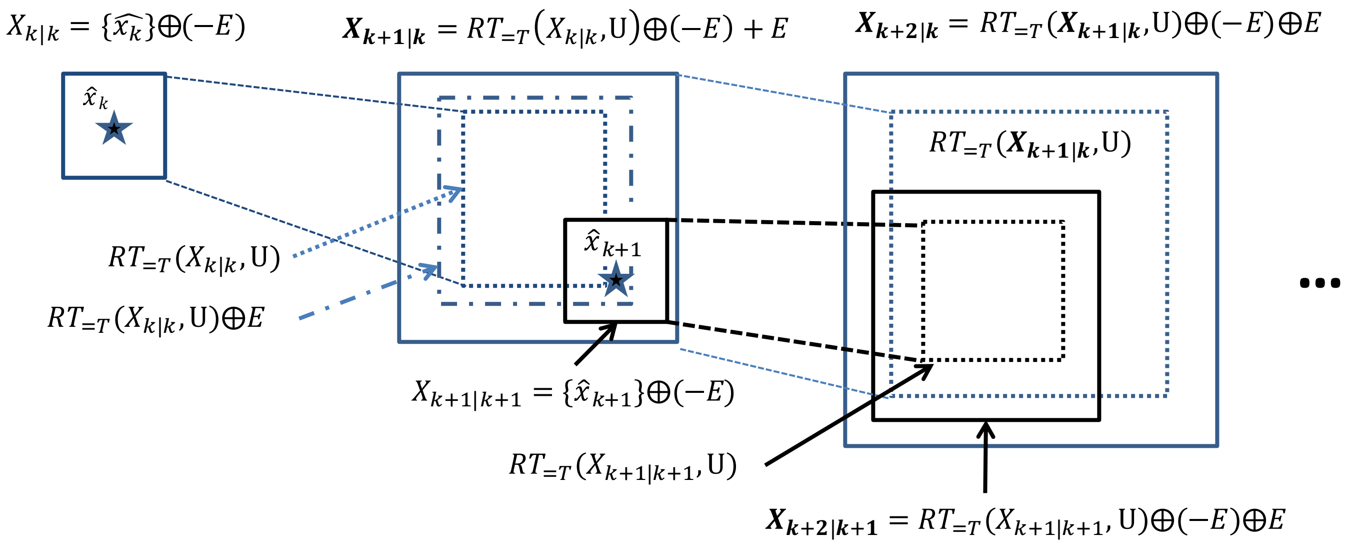
$X_{k|k} = \{\widehat{x_k}\} \oplus (-E)$
$\boldsymbol{X_{k+1|k}} = RT_{=T}(X_{k|k}, \mathrm{U}) \oplus (-E) + E$
$\boldsymbol{X_{k+2|k}} = RT_{=T}(\boldsymbol{X_{k+1|k}}, \mathrm{U}) \oplus (-E) \oplus E$
$\hat{x}_k$
$RT_{=T}(\boldsymbol{X_{k+1|k}}, \mathrm{U})$
$\hat{x}_{k+1}$
$RT_{=T}(X_{k|k}, \mathrm{U})$
$RT_{=T}(X_{k|k}, \mathrm{U}) \oplus E$
•••
$X_{k+1|k+1} = \{\hat{x}_{k+1}\} \oplus (-E)$
$RT_{=T}(X_{k+1|k+1}, \mathrm{U})$
$\boldsymbol{X_{k+2|k+1}} = RT_{=T}(X_{k+1|k+1}, \mathrm{U}) \oplus (-E) \oplus E$

Fig. 1. The outer-approximated reach sets for x_{k+j}, computed at time steps $k, k+1$, used to compute $\widetilde{E}_{k+j|k}$, $\underline{V}_{k+j|k}$.

general $V(x)$ is not a rectangle, we work with inner and outer rectangular approximations of $V(x)$. Specifically, let $\mathcal{X}$ be a subset of X. Define the inner and outer bounding rectangles, respectively

$$\underline{V}(\mathcal{X}) := \{[v_1, \ldots, v_{n_v}]^T \mid \max_{x \in \mathcal{X}} \underline{v}_i(x; U) \leq v_i \leq \min_{x \in \mathcal{X}} \mathbf{v}_i(x; U)\}$$

$$\mathbf{V}(\mathcal{X}) := \{[v_1, \ldots, v_{n_v}]^T \mid \min_{x \in \mathcal{X}} \underline{v}_i(x; U) \leq v_i \leq \max_{x \in \mathcal{X}} \mathbf{v}_i(x; U)\}$$

By construction, we have for any subset $\mathcal{X} \subset X$

$$\underline{V}(\mathcal{X}) \subseteq \cap_{x \in \mathcal{X}} V(x) \subset \mathbf{V}(\mathcal{X}) \qquad (13)$$

If two subsets of X satisfy $\mathcal{X}_1 \subset \mathcal{X}_2$, then it holds that

$$\underline{V}(\mathcal{X}_2) \subset \underline{V}(\mathcal{X}_1),\ \mathbf{V}(\mathcal{X}_1) \subset \mathbf{V}(\mathcal{X}_2) \qquad (14)$$

We can compute:

$$\underline{V}_{k+j|k} = \underline{V}(\mathbf{X}_{k+j|k}),\ \underline{V}_{inner-global} = \underline{V}(X) \qquad (15)$$

In practice we use interval arithmetic to compute these sets since $\mathbf{X}_{k+j|k}$ and U are hyper-intervals. Fig. 2 shows these sets for the running example.

C. Approximating the bounding sets for the disturbances

We will also need to define containing sets for the state estimation error in z space: recall that $\hat{z}_k = T(\hat{x}_k) = T(x_k + e_k)$. We use a Taylor expansion

$$\begin{aligned}\hat{z}_k &= T(x_k) + \underbrace{\frac{dT}{dx}(x_k)}_{M(x_k)} e_k + \underbrace{\frac{1}{2} e_k^T \frac{d^2T}{dx^2}(c) e_k}_{r_k(c)}, c \in x_k + E \\ &= T(x_k) + M(x_k)e_k + r_k(c), c \in x_k + E \\ &= T(x_k) + h_k + r_k(c), c \in x_k + E\end{aligned}$$

The remainder term $r_k(c)$ is bounded in the set $\cup_{c \in \{x_k\} \oplus E} \frac{1}{2} e^T \frac{d^2T}{dx^2}(c) e$. Thus when setting up $\mathbb{P}_k(\hat{z}_k)$, at the j^{th} step, $r_{k+j|k} \in D_{k+j|k} := \cup_{c \in \mathbf{X}_{k+j|k} \oplus E} \frac{1}{2} e^T \frac{d^2T}{dx^2}(c) e$, where $\mathbf{X}_{k+j|k}$ is the reach set computed in (12).

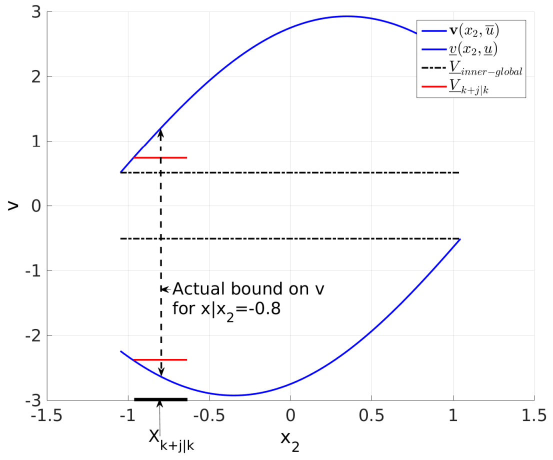

Fig. 2. Local and global inner approximations of input constraints for running example, with $\mathbf{X}_{k+j|k} = [-\pi/4, 0] \times [-0.9666, -0.6283]$ for some k, j and $U = [-2.75, 2.75]$. Color in online version.

The error h_k lives in $\cup_{x \in X_k, e \in E} M(x)e$. Thus when setting up $\mathbb{P}_k(\hat{z}_k)$, the error $h_{k+j|k}$ lives in $\cup_{x \in \mathbf{X}_{k+j|k}} M(x)E$. Finally the rectangular over-approximation of this set is

$$\begin{aligned}H_{k+j|k} = \{h \mid \sum_{\ell=1}^{n_x} \min_{x \in \mathbf{X}_{k+j|k}, e \in E} M_{i\ell}(x)e(\ell) \leq h(i) \\ \leq \sum_{\ell=1}^{n_x} \max_{x \in \mathbf{X}_{k+j|k}, e \in E} M_{i\ell}(x)e(\ell)\} \quad (16)\end{aligned}$$

where $M_{i\ell}$ is the $(i, \ell)^{th}$ element of matrix M and $h(\ell)$ is the ℓ^{th} element of h.

Therefore the state estimation error $h_{k+j|k} + r_{k+j|k}$ is bounded in the set $H_{k+j|k} \oplus D_{k+j|k}$. In the experiments we ignore the remainder term $D_{k+j|k}$ based on the observation

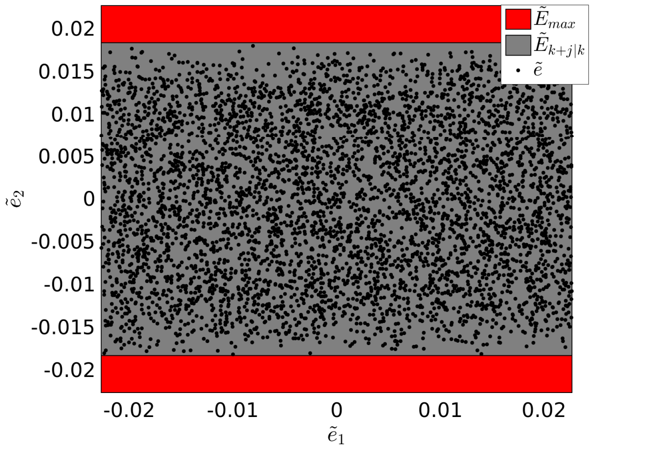

Fig. 3. The error sets $\tilde{E}_{max}$ and $\tilde{E}$ computed over an arbitrary $\mathbf{X}_{k+j|k}$. Also shown are realizations of $\tilde{e} := T(\hat{x}) - T(x)$ for randomly chosen $x \in \mathcal{X}$. Color in online version.

that e_k is small relative to the state x_k. Thus we use:

$$\widetilde{E}_{k+j|k} = H_{j+k|j} \tag{17}$$

Example 2: For the running example (6), we have $M = [1, 0; 0, \cos(x_2)]$. If the estimation error e (in radians) is bounded in $E = \{e|||e||_\infty \leq 0.0227\}$, then the relative linearization error, averaged over several realizations of the error, is less than $2 \cdot 10^{-3}$. □

We also need to calculate containing sets for the process noise $\widehat{w}$. Recall that for all k, j, $\hat{z}_{k+j+1} = A\hat{z}_{k+j} + Bv_k + \widehat{w}_{k+j+1}$. Therefore

$$\widehat{w}_{k+j+1} \in \widehat{W}_{k+j+1|k} := W \oplus \widetilde{E}_{k+j+1|k} \oplus (-A\widetilde{E}_{k+j|k}) \tag{18}$$

We also define the set $\widetilde{E}_{max}$, which is necessary for the terminal constraints of Eq. (10). $\tilde{E}_{max}$ represents the worst case bound on the estimation error $\tilde{e}_k$, and is computed similar to Eq. (17), but over the entire set X.

$$\sum_{\ell=1}^{n} \min_{x \in X, e \in E} M_{i\ell}(x)e(\ell) \leq \tilde{e}_k(i) \leq \sum_{\ell=1}^{n} \max_{x \in X, e \in E} M_{i\ell}(x)e(\ell) \tag{19}$$

$\widehat{W}_{max}$ is then defined as:

$$\widehat{W}_{max} = W \oplus \tilde{E}_{max} \oplus (-A\tilde{E}_{max}) \tag{20}$$

For the running example, Fig. 3 shows the set $\widetilde{E}_{max}$ and $\widetilde{E}_{k+j|k}$ computed by Eqs. (17) and (19). for an arbitrary $\mathbf{X}_{k+j|k} = [-\pi/4, 0] \times [-0.9666, -0.6283]$. It also shows 1000 randomly generated values for $T(\hat{x}) - x$ (for randomly generated $e \in E$ and $x \in \mathbf{X}_{k+j|k}$), and all fall inside $\tilde{E}_{k+j|k}$.

D. Transforming between x-space and z-space

Since we control the system in z-space, we need to compute a set $Z \subset \mathbb{R}^{n_z}$ s.t. $z \in Z \implies x = T^{-1}(z) \in X$, i.e. $Z \subset T(X)$. Thus keeping the state z of the linearized dynamics in Z implies the nonlinear system's state x remains in X. Moreover, to check feasibility at time 0 of the MPC optimization, and for stability of the nonlinear dynamics, we need a subset $X_0 \subset X$ s.t. $x \in X_0 \implies z = T(x) \in Z$, i.e. $X_0 \subset T^{-1}(Z)$. Because T can be an arbitrary diffeomorphism Z and X_0 have to computed numerically.

1) Let $Z_1 \subset \mathbb{R}^{n_z}$ be the rectangle with bounds in the i^{th} dimension $[\min_{x \in X} T_i(x), \max_{x \in X} T_i(x)]$, $i = 1, \ldots, n_x$. This over-approximates $T(X)$. Next we need to prune it so it under-approximates $T(X)$.
2) Define $z_{in} := \min\{\|z\|_0 \mid z \in Z_1, T^{-1}(z) \notin X\}$. z_{in} is the smallest-norm inadmissible z in Z_1. Thus all points in the ℓ_0-ball of radius $\|z_{in}\|$,$B_z(0, \|z_{in}\|)$, are admissible, i.e. their pre-images via T^{-1} are in X.
3) Let R_z be the largest inscribed rectangle in $B_z(0, \|z_{in}\|)$. Now we need to get the x-set that maps to R_z (or a subset of it).
4) Let $X_1 \subset X$ be the rectangle with bounds in the i^{th} dimension $[\min_{z \in R_z} T_i^{-1}(z), \max_{z \in R_z} T_i^{-1}(z)]$. Again, this is an over-approximation of $T^{-1}(R_z)$, so it needs to be pruned.
5) Define $x_{in} = \inf\{\|x\|_0 \mid x \in X_1, T(x) \notin R_z\}$. Then every point in the ℓ_0-ball $B_x(0, \|x_{in}\|) \subset X$ maps via T to R_z

Therefore we choose $Z = R_z$ and X_0 to be the largest inscribed rectangle in $B_x(0, \|x_{in}\|)$.

V. Experiments

We evaluate our approach on the running example, and on a 4D flexible joint manipulator. We implemented the RMPC controller of Alg. 1 in MATLAB The set computations were done using the MPT Toolbox [13], and the invariant set computations using the Matlab Invariant Set Toolbox [14]. The reachability computations for $\mathbf{X}_{k+j|k}$ were performed on the linear dynamics and mapped back to x-space as described in point 4) of Sec. IV-D. The RMPC optimizations were formulated in CVX [15] and solved by Gurobi [16].

A. Running example

For the running example of Eq. 6, we discretize the feedback linearized system at 10Hz and formulate the controller with a horizon of $N = 15$ steps. The cost function has parameters $Q = I$ and $R = 10^{-2}$, and $W = [-10^{-2}, 10^{-2}]^2$. The state trajectories (and estimates) for the nonlinear and linearized systems are shown in Fig. 4. Note that the states converge to the equilibrium 0. The input u is shown in Fig. 5, and it can be noted that $u_k \in U$ for all k.

B. Single link flexible joint manipulator

We consider the single link flexible manipulator system S, also used in [8] and [17], shown in Fig. 6, whose dynamics are given by:

$$S : \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} x_2 \\ -\frac{mgl}{I}\sin(x_1) - \frac{\sigma}{I}(x_1 - x_3) \\ x_4 \\ \frac{\sigma}{J}(x_1 - x_3) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J} \end{bmatrix} u$$

This models a system where a motor, with an angular moment of inertia $J = 1$, is coupled to a uniform thin bar of

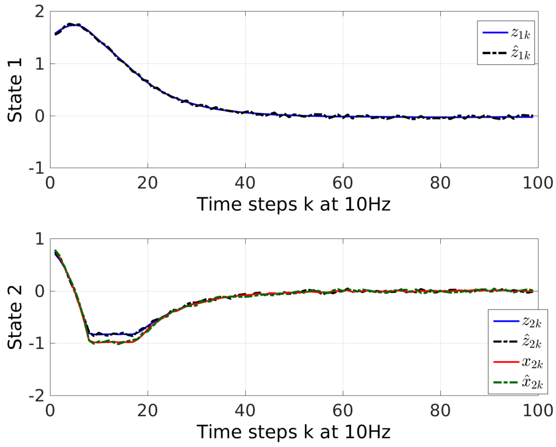

Fig. 4. The states and their estimates of the feedback linearized and non-linear running example. Recall that $z_1 = x_1$ therefore to reduce clutter, we only plot the first state only for the feedback linearized system. Color in online version.

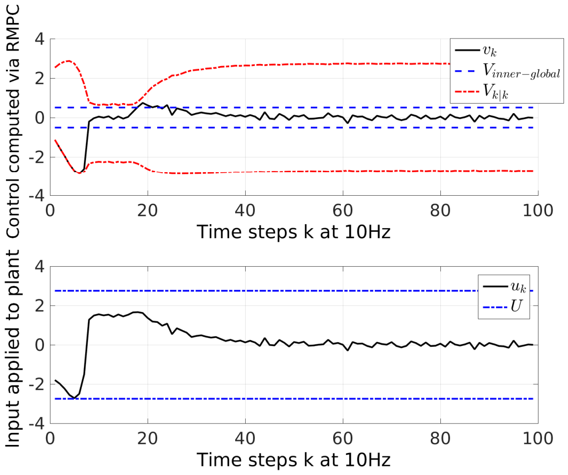

Fig. 5. Inputs v and u and their bounds for the running example. Color in online version.

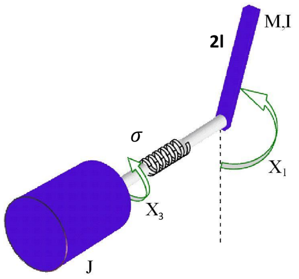

Fig. 6. Flexible joint manipulator. Figure adapted from [17].

mass $m = 1/g$, length $l = 1m$ and moment of inertia $I = 1$, through a flexible torsional string with stiffness $\sigma = 1$ and $g = 9.8ms^{-2}$. States x_1 and x_2 are the angles of the bar and motor shaft in radians, respectively, and x_3, x_4 are their respective rotational speeds in radians/sec. The safe set is the box $X = [-\pi/4, \pi/4] \times [-\pi/4, \pi/4] \times [-\pi, \pi] \times [-\pi, \pi]$. The input torque u is bounded in $U = [\underline{u}, \overline{u}] = [-10, 10]N \cdot m$. The estimation error $e = \hat{x} - x$ is bounded in $E = [-\pi/180, \pi/180]^4$ and $W = [-10^{-3}, 10^{-3}]^4$.

The diffeomorphism T is given by:

$$z = T(x) = \begin{bmatrix} x_1 \\ x_2 \\ -\frac{mgl}{I}\sin(x_1) - \frac{\sigma}{I}(x_1 - x_3) \\ \frac{mgl}{I}x_2\cos(x_1) - \frac{\sigma}{I}(x_2 - x_4) \end{bmatrix}$$

The input to the feedback linearized system is given by $v = \beta u + \alpha(x)$ where $\beta = \frac{\sigma}{IJ}$ and

$$\begin{aligned} \alpha(x) &= \frac{mgl}{I}x_2^2 sin(x_1) + \frac{\sigma^2}{IJ}(x_1 - x_3) \\ &- (\frac{mgl}{I}cos(x_1) - \frac{\sigma}{I})(\frac{mgl}{I}sin(x_1) + \frac{\sigma}{I}(x_1 - x_3)) \end{aligned}$$

The feedback linearized system S_{fl} has the dynamics: $\dot{z}_1 = z_2, \dot{z}_2 = z_3, \dot{z}_3 = z_4, \dot{z}_4 = v$.

A global inner approximation of the v input set is computed, via interval arithmetic, as $V_{inner-global} = [max_{x\in X}\alpha(x) + \beta\underline{u}, min_{x\in X}\alpha(x) + \beta\overline{u}]$. Similarly, the inner approximations $\underline{V}_{k+j|k}$ are computed online by interval arithmetic as $\underline{V}_{k+j|k} = [max_{x\in \mathbf{X}_{k+j|k}}\alpha(x) + \beta\underline{u}, min_{x\in \mathbf{X}_{k+j|k}}\alpha(x) + \beta\overline{u}]$. Using the procedure of Sec. IV-D the set of safe states for S_{fl} is given by $Z = [-0.5121, 0.5121]^2 \times [-2.5347, 2.5347] \times [-2.5603, 2.5603]$. Also $X_0 = [-0.4655, 0.4655]^2 \times [-2.7598, 2.7598] \times [-2.793, 2.793]$. Comparing it to the set X, it shows that we can stabilize the system starting from initial states in a significantly large region in X.

We applied our controller to the above system with a discretization rate of 10Hz and MPC horizon $N = 10$. Fig.7 show the states of the feedback linearized system S_{fl}. They converge to the origin in the presence of estimation error, while respecting all constraints. Fig. 7 also shows x_3 and x_4: they also converge to zero (and remember $x_1 = z_1$ and $x_2 = z_2$).

Fig. 8 shows the input v to S_{fl} along with the global inner approximation $V_{inner-global}$ and the x-dependent inner approximations at the instant when the control is applied, $\underline{V}_{k|k}$ computed online. Note that the bounds computed online allow for significantly more control action compared to the conservative global inner approximation. Finally, Fig. 8 also shows the input u applied to the non-linear system (and its bounds), which robustly respects its constraints $u \in U$.

VI. Discussion

In this paper we develop the first algorithm for robust control of a non-linear system with estimation errors and state and input constraints via feedback linearization and Robust MPC. We develop a method to compute linear

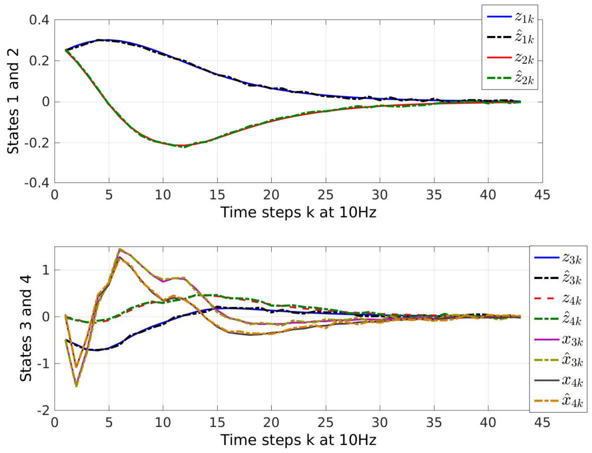

Fig. 7. The states and their estimates of the feedback linearized and non-linear manipulator. Recall that $z_1 = x_1$ and $z_2 = x_2$, therefore to reduce clutter, we only plot first two states only for the feedback linearized system. Color in online version.

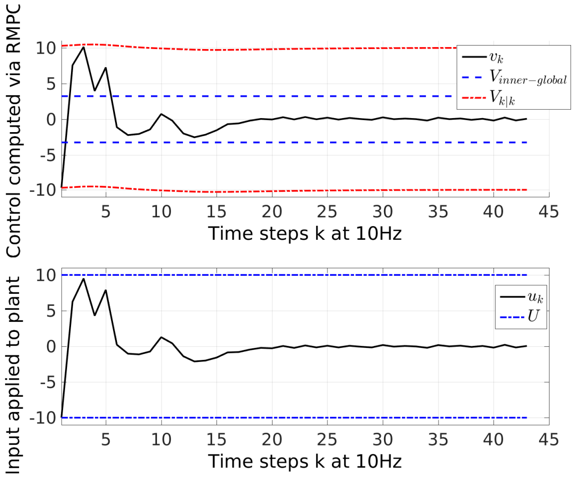

Fig. 8. Inputs v and u and their bounds for the manipulator example. Color in online version.

constraints on the state and inputs of the feedback linearized system such that the non-linear system respects its state and input constraints under bounded state estimation errors. We demonstrate the applicability of our approach on a planar system and a flexible link manipulator example. Results show that the control algorithm stabilizes the systems while ensuring robust constraint satisfaction.While we only evaluated our approach for single input systems, the formulation and set computations involved hold as is for multi-input systems as well.

Limitations of the approach mostly have to do with the numerical limitations involved in computing the constraint sets. E.g., in the manipulator example, the set of states X_0 from which we can control the system is a strict subset of the set of safe states X. Similarly in the computation of the input and error sets, over-conservatism is potentially a problem.

Ongoing work focuses on implementing this approach for evaluation on a $1/10^{th}$ scale autonomous car, running a low power embedded platform. Real-time online reachability [18], interval arithmetic, and support function based computations for $\bar{Z}_{k+j|k}$ should allow for fast enough computation of the linear constraints for the RMPC optimization. In [10], we have already shown that CVXGEN [19] is fast enough to solve Quadratic programs with linear constraints on low-powered embedded platforms at high enough sampling rates to allow for satisfactory control of a real system.

References

[1] M. Cannon, "Efficient nonlinear model predictive control algorithms," *Annual Reviews in Control*, vol. 28, no. 2, pp. 229 – 237, 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1367578804000355

[2] H. Khalil, "Nonlinear systems," in *Prentice Hall*, 2002.

[3] D. Simon, J. Lofberg, and T. Glad, "Nonlinear model predictive control using feedback linearization and local inner convex constraint approximations," in *Control Conference (ECC), 2013 European*, July 2013, pp. 2056–2061.

[4] A. Richards and J. How, "Robust model predictive control with imperfect information," in *American Control Conference, 2005. Proceedings of the 2005*, June 2005, pp. 268–273.

[5] D. Mayne and E. Kerrigan, "Tube-based Robust Nonlinear Model Predictive Control ," in *IFAC Symposium on Nonlinear Control Systems*, 2007.

[6] S. Streif, M. Kogel, T. Bathge, and R. Findeisen, "Robust Nonlinear Model Predictive Control with Constraint Satisfaction: A Relaxation-based Approach," in *IFAC World Congress*, 2014.

[7] W. Zhao and T. H. Go, "Quadcopter formation flight control combining mpc and robust feedback linearization," *Journal of the Franklin Institute*, vol. 351, no. 3, pp. 1335 – 1355, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S001600321300389X

[8] W. Son, J. Choi, and O. Kwon, "Robust control of feedback linearizable system with the parameter uncertainty and input constraint," in *Proceedings of the 40th SICE Annual Conference*, 2001.

[9] Y. V. Pant, H. Abbas, and R. Mangharam, "Tech report: Robust model predictive control for non-linear systems with input and state constraints via feedback linearization," March 2016. [Online]. Available: http://tinyurl.com/nlrmpcwisc

[10] Y. V. Pant, K. Mohta, H. Abbas, T. X. Nghiem, J. Devietti, and R. Mangharam, "Co-design of anytime computation and robust control," in *Real-Time Systems Symposium, 2015 IEEE*, Dec 2015, pp. 43–52.

[11] B. Kouvaritakis and M. Cannon, *Model Predictive Control: Classical, Robust and Stochastic*. Springer Verlag, 2015.

[12] T. T. Johnson, S. Bak, M. Caccamo, and L. Sha, "Real-time reachability for verified simplex design," *ACM Trans. Embed. Comput. Syst.*, vol. 15, no. 2, pp. 26:1–26:27, Feb. 2016. [Online]. Available: http://doi.acm.org/10.1145/2723871

[13] M. Herceg, M. Kvasnica, C. Jones, and M. Morari, "Multi-Parametric Toolbox 3.0," in *Proc. of the European Control Conference*, Zürich, Switzerland, July 17–19 2013, pp. 502–510, http://control.ee.ethz.ch/~mpt.

[14] E. Kerrigan, "Matlab invariant set toolbox version 0.10.5," 2016. [Online]. Available: {http://www-control.eng.cam.ac.uk/eck21/matlab/invsetbox}

[15] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.0 beta," http://cvxr.com/cvx, Sep. 2013.

[16] I. Gurobi Optimization, "Gurobi optimizer reference manual," 2015. [Online]. Available: http://www.gurobi.com

[17] M. Seidi, M. Hajiaghamemar, and B. Segee, "Fuzzy Control Systems: LMI-Based Design," *Fuzzy Controllers- Recent Advances in Theory and Applications, InTech*, 2012.

[18] S. Bak, T. T. Johnson, M. Caccamo, and L. Sha, "Real-time reachability for verified simplex design," in *Real-Time Systems Symposium (RTSS), 2014 IEEE*, Dec 2014, pp. 138–148.

[19] J. Mattingley and S. Boyd, "Cvxgen: a code generator for embedded convex optimization," *Optimization and Engineering*, 2012.