

# Learning a Feasible and Stabilizing Explicit Model Predictive Control Law by Robust Optimization

Alexander Domahidi<sup>1</sup>, Melanie N. Zeilinger<sup>1</sup>, Manfred Morari<sup>1</sup> and Colin N. Jones<sup>2</sup>

**Abstract**—Fast model predictive control on embedded systems has been successfully applied to plants with microsecond sampling times employing a precomputed state-to-input map. However, the complexity of this so-called explicit MPC can be prohibitive even for low-dimensional systems. In this paper, we introduce a new synthesis method for low-complexity suboptimal MPC controllers based on function approximation from randomly chosen point-wise sample values. In addition to standard machine learning algorithms formulated as convex programs, we provide sufficient conditions on the learning algorithm in the form of tractable convex constraints that guarantee input and state constraint satisfaction, recursive feasibility and stability of the closed loop system. The resulting control law can be fully parallelized, which renders the approach particularly suitable for highly concurrent embedded platforms such as FPGAs. A numerical example shows the effectiveness of the proposed method.

## I. INTRODUCTION

In model predictive control (MPC), the control input is obtained by solving an optimization problem at each sampling time. It is well-known that for certain problem structures, the resulting quadratic or linear program can be solved offline using multiparametric programming [1]. The solution is a piece-wise affine (PWA) state feedback control law defined over polyhedral regions partitioning the state space. Thus, online evaluation of the control law reduces to a point location problem, which can be solved efficiently using a binary search tree [2], [3]. This so-called explicit MPC enables finite time optimal control of fast systems such as power electronics, see e.g. [4] and the references therein.

Explicit MPC is however limited mainly in two ways. First, the computation of the parametric solution may be computationally intractable, even for low dimensional systems. This stems from the fact that the solution to multiparametric programs can grow exponentially in the problem size. Second, if a PWA state feedback map has been obtained, generation of the associated search tree may fail or lead to lookup tables of sizes prohibitive for embedded control hardware.

In this work, we propose a new synthesis method for suboptimal explicit MPC based on function approximation which yields low-complexity control laws with a low degree of suboptimality, enhancing applicability of MPC on resource constrained platforms. Our method is based on approximating the optimal PWA control law from point-wise sample

values by a weighted sum of nonlinear basis functions. As the samples are chosen randomly, we call this process *learning*, and the learning problem consists of finding a suitable set of weights by means of convex optimization.

While standard methods from machine learning, e.g. soft margin support vector regression [5] or more general kernel methods [6], may work well in practice, no system theoretic guarantees are provided when learning MPC control laws. The main contribution of this work is to present a learning problem that provides guarantees for input and state constraint satisfaction, recursive feasibility and stability of the closed loop system with only mild requirements on the class of basis functions. This is accomplished by deriving suitable tractable convex constraints which can be added to any of the standard learning setups relying on convex optimization. An observation from our numerical experiments is that when learning with such system theoretic constraints, the resulting weight matrix is often sparse, resulting in low storage requirements and fast online evaluation.

A vast number of other approximate explicit MPC schemes have been proposed in the literature [7]. While geometric methods (e.g. [8], [9], [10], [11], [12]) are based on the polyhedral partitioning of the state space, our approach belongs to the class of function approximation. An early work in this context is [13], employing neural networks and a gradient-based search to find the associated weights through non-convex optimization. Function approximation by PWA basis functions is considered in [14], where the offline problem is a convex quadratic or linear program. However, the exact solution is needed to compute the approximation, and closed-loop stability is verified a-posteriori by construction of a piece-wise linear (PWL) Lyapunov function, which is generally difficult to obtain. Both the exact solution and a PWL Lyapunov function are also prerequisites for stability guarantees given in [15], which is based on a global polynomial approximation and is restricted to  $1, \infty$ -norm cost functions. Barycentric interpolations on a hierarchical grid are computed in [16] by a series of convex optimization problems.

In contrast, the method presented in this paper permits to use any differentiable function as basis function, which gives freedom to the control designer to trade off approximation accuracy vs. memory and computing power limitations. As the basis functions can be evaluated fully in parallel, the resulting state feedback controller is particularly suited for highly concurrent embedded platforms such as FPGAs or VLIW processors. Moreover, our algorithm relies solely on point-wise values of the optimal solution and is hence

<sup>1</sup>Automatic Control Laboratory, Department of Information Technology and Electrical Engineering, ETH Zurich, 8092 Zurich, Switzerland. domahidi|zeilinger|morari@control.ee.ethz.ch

<sup>2</sup>Automatic Control Laboratory, École Polytechnique Fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland. colin.jones@epfl.ch

potentially suited for MPC problems of high complexity. We approximate only the first control move of the input trajectory, which is sufficient for closed loop control in a receding horizon implementation, thereby significantly reducing the problem size.

The outline of the paper is as follows: In Section II we introduce the MPC formulation under consideration and the general function approximation setting, followed by a specialization to learning MPC controllers in Section III. After stating the semi-infinite learning problem, we subsequently derive finite tractable convex inner approximations of the infinite constraints. Our method is applied to an example system in Section IV, showing the effectiveness of learning optimal controllers under system theoretic constraints.

## NOTATION

$I_n$  is the  $n \times n$  identity matrix.  $\mathbb{R}_{\geq 0}$  ( $\mathbb{R}_+$ ) is the set of nonnegative (strictly positive) reals. The operators  $\otimes$  and  $\oplus$  denote the Kronecker product and the Minkowski sum, respectively.  $\text{vec}(A)$  is the vector obtained by stacking the columns of matrix  $A$ . A *polyhedron* is an intersection of a finite number of closed half-spaces in  $\mathbb{R}^n$ , and a *polytope* is a bounded polyhedron.  $\text{extr}P$  denotes the set of vertices of the polytope  $P$ . All inequalities are component-wise if applied to vectors.

## II. PRELIMINARIES

In this work, we consider linear discrete time dynamical systems of the form

$$x^+ = Ax + Bu, \quad (1)$$

where  $x \in \mathbb{R}^{n_x}$  is the current state,  $u \in \mathbb{R}^{n_u}$  the current control input and  $x^+$  the successor state. If (1) is controlled by the control law  $u = \kappa(x)$ , the closed loop system is defined as

$$x^+ = Ax + B\kappa(x). \quad (2)$$

**Definition II.1 (Positively invariant (PI) set)** A set  $\mathcal{X} \subseteq \mathbb{R}^{n_x}$  is a *Positively Invariant (PI) set* of system (2), if  $Ax + B\kappa(x) \in \mathcal{X} \forall x \in \mathcal{X}$ .

**Definition II.2 (K-class function)** A real-valued function  $\alpha : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  belongs to class  $K$  if it is continuous, strictly increasing and  $\alpha(0) = 0$ .

**Definition II.3 (Lyapunov function)** Let  $\mathcal{X}$  be a PI set for system (2) containing a neighborhood  $\mathcal{N}$  of the origin in its interior and let  $\underline{\alpha}(\cdot)$ ,  $\bar{\alpha}(\cdot)$  and  $\beta(\cdot)$  be  $K$ -class functions. A non-negative function  $V : \mathcal{X} \rightarrow \mathbb{R}_{\geq 0}$  with  $V(0) = 0$  is called a *Lyapunov function* in  $\mathcal{X}$  if:

$$V(x) \geq \underline{\alpha}(\|x\|) \quad \forall x \in \mathcal{X}, \quad (3)$$

$$V(x) \leq \bar{\alpha}(\|x\|) \quad \forall x \in \mathcal{N}, \quad (4)$$

$$V(x^+) - V(x) \leq -\beta(\|x\|) \quad \forall x \in \mathcal{X}. \quad (5)$$

**Theorem II.4 (Asymptotic stability [17])** If system (2) admits a Lyapunov function in  $\mathcal{X}$ , then the equilibrium point at the origin is asymptotically stable with region of attraction  $\mathcal{X}$ .

## A. MPC Problem Formulation

In model predictive control, a cost function reflecting various control tradeoffs is minimized over a finite time horizon with respect to the control inputs  $\mathbf{u} = [u_0, \dots, u_{N-1}]$ :

$$V_N^*(x) = \min_{\mathbf{u}} V_N(x, \mathbf{u}) = \sum_{n=0}^{N-1} l(x_n, u_n) + V_f(x_N) \quad (6a)$$

$$\text{s.t. } x_0 = x, \quad (6b)$$

$$x_{n+1} = Ax_n + Bu_n \quad n = 0, \dots, N-1, \quad (6c)$$

$$(x_n, u_n) \in \mathbb{X} \times \mathbb{U} \quad n = 0, \dots, N-1, \quad (6d)$$

$$x_N \in \mathcal{X}_f. \quad (6e)$$

The sets  $\mathbb{X} = \{x \in \mathbb{R}^{n_x} | H_X x \leq h_X\}$  and  $\mathbb{U} = \{u \in \mathbb{R}^{n_u} | H_u u \leq h_u\}$  are assumed to be closed and convex polytopes, containing the origin in their interior. We define  $\mathcal{X}_N$  as the set of  $x$  for which (6) is feasible. The stage costs are of the form  $l(x_i, u_i) = x_i^T Q x_i + u_i^T R u_i$  where  $Q \in \mathbb{R}^{n_x \times n_x}$  is a positive semi-definite and  $R \in \mathbb{R}^{n_u \times n_u}$  a positive definite weight matrix. The terminal penalty function is given by  $V_f(x_N) = x_N^T P x_N$  for some  $P \in \mathbb{R}^{n_x \times n_x}$  positive definite. The terminal target set  $\mathcal{X}_f \subseteq \mathbb{X}$  is a compact convex polytope containing the origin in its interior. We denote the minimizer of (6) by  $\mathbf{u}^*(x)$ , and the corresponding state feedback control law is the first element of the optimal input sequence,  $\kappa(x) = u_0^*(x)$ , such that the closed loop system dynamics under receding horizon control are given by

$$x^+ = Ax + Bu_0^*(x). \quad (7)$$

**Assumption II.5** In the following, we assume that  $\mathcal{X}_f$  is a PI set under the control law  $\kappa_f(x)$  and that  $V_f(\cdot)$  is a Lyapunov function in  $\mathcal{X}_f$ :

$$A1: Ax + B\kappa_f(x) \in \mathcal{X}_f \text{ and } \kappa_f(x) \in \mathbb{U} \forall x \in \mathcal{X}_f,$$

$$A2: V_f(Ax + B\kappa_f(x)) - V_f(x) \leq -l(x, \kappa_f(x)) \forall x \in \mathcal{X}_f.$$

**Theorem II.6 (Lyapunov function for (7) [18])** Under Assumption II.5,  $V_N^*(\cdot)$  is a Lyapunov function for the closed loop system (7).

## B. Kernel Regression

In classical nonlinear regression, one seeks to approximate a function  $f : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$  in some function space  $\mathcal{F}$ , by a function  $\tilde{f} : \Omega \rightarrow \mathbb{R}$  in some  $\tilde{\mathcal{F}} \subseteq \mathcal{F}$ , such that the approximation error  $\|f - \tilde{f}\|_{\mathcal{F}}$ , where  $\|\cdot\|_{\mathcal{F}}$  is a norm defined on  $\mathcal{F}$ , is minimized:

$$\min_{\tilde{f} \in \tilde{\mathcal{F}}} \|f - \tilde{f}\|_{\mathcal{F}}.$$

Since the minimization problem is carried out over the space of functions  $\tilde{\mathcal{F}}$ , it is in general infinite dimensional and computationally intractable. A common approach is to parametrize  $\tilde{\mathcal{F}}$  by a weighted sum of  $L$  nonlinear basis functions  $k_i : \Omega \rightarrow \mathbb{R}$  and to minimize the *empirical error* instead. This involves drawing  $M$  samples  $\chi_i$  according to a distribution  $\mathcal{P} : \Omega \rightarrow \mathbb{R}_{\geq 0}$ . At the samples,  $f(\cdot)$  is evaluated to yield  $f_i := f(\chi_i)$ . With

$$k(x) := [k_1(x), \dots, k_L(x)]^T, \quad (8)$$

the finite dimensional problem of learning  $f$  is then given by

$$\theta^* := \arg \min_{\theta} \sum_{i=1}^M \mathcal{L}(r_i) \quad (9a)$$

$$\text{s.t. } r_i = f(\chi_i) - \tilde{f}(\chi_i) \quad i = 1, \dots, M \quad (9b)$$

$$\tilde{f}(\chi_i) = \theta^T k(\chi_i) \quad i = 1, \dots, M \quad (9c)$$

where  $\theta^* \in \mathbb{R}^L$  is the set of weights that minimize the empirical error under loss function  $\mathcal{L} : \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}$ . For some common choices of basis functions and the relation to kernel methods see e.g. [5] and [19].

### III. LEARNING AN MPC CONTROLLER

#### A. Sampling

To generate data for learning, we define  $\Omega = \mathcal{X}_N$  as the domain of the basis functions  $k_i(\cdot)$ , which ensures that the samples  $\chi_i$  are feasible for the original MPC problem (6).

#### B. The primary learning problem

First, note that the learning problem in (9) has to be adjusted in order to fit our purpose of learning an MPC control law.

**Definition III.1 (Suboptimal control law)** *The suboptimal state feedback control law employs a dual-mode strategy [18], [20] and is given by*

$$\tilde{u}(x) := \begin{cases} \Theta k(x) & x \notin \mathcal{X}_f \\ \kappa_f(x) & x \in \mathcal{X}_f \end{cases}, \quad (10)$$

where  $\Theta \in \mathbb{R}^{n_u \times L}$  is a set of weights and  $k(x)$  is a column vector of basis functions evaluated at  $x$  as defined in (8). The suboptimal closed loop dynamics are given by

$$x^+ = Ax + B\tilde{u}(x). \quad (11)$$

We first state a rather general result for this class of suboptimal controllers. By incorporating input and state constraint satisfaction, recursive feasibility and stability constraints into the learning problem (9), we impose system theoretic constraints on (10) and (11). Thus the following theorem is key to our method.

**Theorem III.2 (Primary learning problem)** *Let  $\tilde{\mathcal{X}} \subseteq \mathcal{X}_N$  be a polytopic set containing the origin in its interior. Furthermore, let  $\chi_i \in \mathcal{X}_N$ ,  $i = 1, \dots, M$  denote randomly drawn samples. If the semi-infinite optimization problem over the weights  $\Theta$  of (10),*

$$\min_{\Theta} \sum_{i=1}^M \mathcal{L}(r_i) \quad (12a)$$

$$\text{s.t. } r_i = u_0^*(\chi_i) - \tilde{u}(\chi_i), \quad i = 1, \dots, M, \quad (12b)$$

$$\tilde{u}(x) \in \mathbb{U} \quad \forall x \in \tilde{\mathcal{X}}, \quad (12c)$$

$$Ax + B\tilde{u}(x) \in \tilde{\mathcal{X}} \quad \forall x \in \tilde{\mathcal{X}}, \quad (12d)$$

$$V_N^*(x^+) - V_N^*(x) \leq -\varepsilon l(x, 0) \quad \forall x \in \tilde{\mathcal{X}}, \quad (12e)$$

with  $x^+$  as given in (11),  $r_i \in \mathbb{R}^{n_u}$ ,  $\mathcal{L} : \mathbb{R}^{n_u} \rightarrow \mathbb{R}_{\geq 0}$  and  $\varepsilon \in \mathbb{R}_+$  is feasible for some fixed  $\varepsilon > 0$ , then  $\tilde{\mathcal{X}}$  is a

PI set for the closed loop system (11). Furthermore, (11) is asymptotically stable with region of attraction  $\tilde{\mathcal{X}}$ .

**Remark III.3** *As the suboptimal control law employs a dual-mode strategy, it suffices to impose (12c), (12d) and (12e) for all  $x$  in  $\tilde{\mathcal{X}} \setminus \mathcal{X}_f$ , as the terminal controller  $\kappa_f(x)$  satisfies these conditions by Assumption II.5.*

The primary learning problem (12) is merely theoretical, as the optimization problem has an infinite number of constraints. In the next section, we will derive the main result of the paper, which is a tractable convex inner approximation of (12) with a finite number of constraints.

#### C. Derivation of a tractable learning problem

We will first derive a tractable formulation of the input constraints (12c), by making use of a robust optimization reformulation technique, see e.g. [21]. A similar reformulation is also employed in the subsequent derivation of convex conditions sufficient for (12d). Finally, we give sufficient finite convex constraints for (12e).

Some mild assumptions are required for the basis functions and the involved sets, which are met by all standard basis functions used in learning [22].

**Assumption III.4** *The basis functions  $k_i(\cdot)$  are continuously differentiable on  $\tilde{\mathcal{X}}$ .*

**Assumption III.5** *Let the set  $\tilde{\mathcal{X}} \subseteq \mathbb{X}$  be compact and convex with non-empty interior and implicit representation, i.e.  $\tilde{\mathcal{X}} = \{x \in \mathbb{R}^{n_x} \mid \exists \mathbf{u} : \mathcal{G}x + \mathcal{H}\mathbf{u} \leq b\}$ , with  $\mathcal{G} \in \mathbb{R}^{m \times n_x}$ ,  $\mathcal{H} \in \mathbb{R}^{m \times N \cdot n_u}$ ,  $b \in \mathbb{R}^m$  and  $\mathbf{u}$  is the input vector. For details on how to derive such a representation see e.g. [23].*

**Lemma III.6 (First order Taylor approximation)** *Let  $\mathcal{R} \subseteq \tilde{\mathcal{X}}$ , where  $\mathcal{R} = \{x \in \tilde{\mathcal{X}} \mid H_r x \leq h_r\}$  is defined by a finite number of linear inequalities with  $h_r \in \mathbb{R}^{N_r}$ . The first order approximation of  $k(x)$  around a point  $x_c \in \mathcal{R}$  allows for a representation*

$$k(x) \in J_k(x_c)(x - x_c) + k(x_c) \oplus \mathbb{W}, \quad x \in \mathcal{R},$$

where  $J_k(x_c) := \nabla_x k(x)|_{x=x_c} \in \mathbb{R}^{L \times n_x}$  is the Jacobian of  $k(\cdot)$  evaluated at  $x_c$  and  $\mathbb{W} = \{w \in \mathbb{R}^L \mid H_w w \leq h_w\}$  is a compact convex polytope defined by  $H_w \in \mathbb{R}^{N_w \times L}$  and  $h_w \in \mathbb{R}^{N_w}$  overbounding the remainder of the Taylor expansion.

*Proof:* Since  $k(\cdot)$  is differentiable on  $\mathbb{X}$  by Assumption III.4, the Jacobian exists. Since  $\tilde{\mathcal{X}} \subseteq \mathbb{X}$  is bounded, the remainder  $w(x) := k(x) - J_k(x_c)(x - x_c) - k(x_c)$  must be bounded for all  $x \in \mathcal{R}$ . ■

**Remark III.7** *In general, the map of  $\mathcal{R}$  under  $w(\cdot)$  as defined in the proof of Lemma III.6 is nonlinear, but can be overbounded by interval arithmetic methods [24], yielding the polytope  $\mathbb{W}$ .*

**Assumption III.8** *The set  $\mathbb{W}$  has a non-empty interior.*

#### D. Satisfying input constraints

The following theorem states the tractable counterpart of invariance condition (12c).

**Theorem III.9 (Tractable input constraints)** Define  $\mathcal{R}$  and  $\mathbb{W}$  as in Lemma III.6 and fix  $x_c \in \mathcal{R}$ . Furthermore, define

$$\mathcal{A} := \begin{bmatrix} \mathcal{G} & 0 & \mathcal{H} \\ 0 & H_w & 0 \\ H_r & 0 & 0 \end{bmatrix}, \quad \mathcal{B} := \begin{bmatrix} b \\ h_w \\ h_r \end{bmatrix},$$

$$\mathcal{C}(\Theta, x_c) := [H_u \Theta J_k(x_c) \quad H_u \Theta \quad 0],$$

$$\mathcal{D}(x_c) := h_u + H_u \Theta (J_k(x_c) x_c - k(x_c)),$$

and let  $N_u$  be the number of rows of  $H_u$ . Then, the existence of  $\Theta \in \mathbb{R}^{n_u \times L}$  and  $Y \in \mathbb{R}^{N_u(m+N_w+N_r)}$  such that

$$(I_{N_u} \otimes \mathcal{B})^T Y \leq \mathcal{D}(x_c), \quad (13a)$$

$$(I_{N_u} \otimes \mathcal{A})^T Y = \text{vec}(\mathcal{C}(\Theta, x_c)^T), \quad (13b)$$

$$Y \geq 0, \quad (13c)$$

is a sufficient condition for (12c) on  $\mathcal{R}$ , i.e. (13) implies  $\tilde{u}(x) \in \mathbb{U} \forall x \in \mathcal{R}$ .

*Proof:* Using Lemma III.6, a sufficient condition for  $\tilde{u}(x) \in \mathbb{U} \forall x \in \mathcal{R}$  is given by

$$H_u \Theta (J_k(x_c)(x - x_c) + k(x_c) + w) \leq h_u$$

$$\forall x \in \mathcal{R}, \forall w \in \mathbb{W} \quad (14a)$$

$$\Leftrightarrow H_u \Theta J_k(x_c) x + H_u \Theta w \leq h_u + H_u \Theta (J_k(x_c) x_c - k(x_c))$$

$$\forall x \in \mathcal{R}, \forall w \in \mathbb{W}. \quad (14b)$$

Using the matrices introduced in the theorem, we rewrite the infinite condition as constraints on the row-wise maxima of the LHS of (14b) as follows:

$$\max_{x \in \mathcal{R}, w \in \mathbb{W}} \mathcal{C}_i(\Theta, x_c) \begin{bmatrix} x \\ w \\ 0 \end{bmatrix} \leq \mathcal{D}_i(x_c), \quad (15a)$$

$$\Leftrightarrow \max_{x, w, \mathbf{u}} \mathcal{C}_i(\Theta, x_c) \begin{bmatrix} x \\ w \\ \mathbf{u} \end{bmatrix} \leq \mathcal{D}_i(x_c), \quad (15b)$$

$$\text{s.t.} \quad \mathcal{A} \begin{bmatrix} x \\ w \\ \mathbf{u} \end{bmatrix} \leq \mathcal{B} \quad (15c)$$

for  $i = 1, \dots, N_u$ , where  $\mathcal{C}_i(\Theta, x_c)$ ,  $\mathcal{D}_i(x_c)$  denote the  $i^{\text{th}}$  row of the corresponding matrices. Since both  $\tilde{\mathcal{X}}$  and  $\mathbb{W}$  are compact convex sets with nonempty interior (Assumption III.5 and Assumption III.8), strong duality for LPs holds and we then have that

$$\max_{x, w, \mathbf{u}} \mathcal{C}_i(\Theta, x_c) \begin{bmatrix} x \\ w \\ \mathbf{u} \end{bmatrix} = \min_{y_i} \mathcal{B}^T y_i \leq \mathcal{D}_i(x_c) \quad (16)$$

$$\text{s.t.} \quad \mathcal{A}^T y_i = \mathcal{C}_i(\Theta, x_c)^T,$$

$$y_i \geq 0,$$

$$\text{s.t.} \quad \mathcal{A} \begin{bmatrix} x \\ w \\ \mathbf{u} \end{bmatrix} \leq \mathcal{B}$$

$i = 1, \dots, N_u$ . Since the existence of  $y_i$  satisfying (16) is sufficient, the minimization in (16) can be dropped. Repeating this construction for every row of  $\mathcal{C}(\Theta, x_c)$  and stacking the corresponding variables to  $Y := [y_1^T, \dots, y_{N_u}^T]^T$  yields the claim. ■

**Remark III.10 (Input feasibility for union of regions)**

Let  $R$  regions  $\mathcal{R}^i$ ,  $i = 1, \dots, R$  correspond to a covering of  $\tilde{\mathcal{X}}$ , i.e.  $\cup_{i=1}^R \mathcal{R}^i = \tilde{\mathcal{X}}$  and let Theorem III.9 be satisfied for all  $R$  regions with separate  $Y^i$  and one  $\Theta$ . Then Theorem III.9 holds on  $\tilde{\mathcal{X}}$ .

**Remark III.11 (Tractability of (13))** Both the number of additional variables  $Y$  and the number of constraints introduced in Theorem III.9 is polynomial in the dimensions involved. We have precisely  $N_u(1 + m + N_w + N_r)$  inequality and  $N_u(n_x + L + N n_u)$  equality constraints in  $N_u(m + N_w + N_r)$  variables.

**E. Recursive feasibility**

The following theorem states the tractable counterpart of invariance condition (12d).

**Theorem III.12 (Tractable invariance constraint)** Let  $\tilde{\mathcal{X}}$  be given by an explicit polytopic representation, i.e.  $\tilde{\mathcal{X}} = \{x \in \mathbb{R}^{n_x} \mid H_x x \leq h_x\}$  with  $h_x \in \mathbb{R}^{N_x}$ . Let furthermore  $x_c \in \mathcal{R} \subseteq \tilde{\mathcal{X}}$  be fixed and  $\mathcal{R}$ ,  $\mathcal{A}$ ,  $\mathcal{B}$  and  $\mathbb{W}$  be defined as in Theorem III.9. Define furthermore

$$\mathcal{E}(\Theta, x_c) := [H_x(A + B\Theta J_k(x_c)) \quad H_x B \Theta \quad 0],$$

$$\mathcal{F}(\Theta, x_c) := h_x + H_x B \Theta (J_k(x_c) x_c - k(x_c)).$$

Then a sufficient condition for the semi-infinite constraint

$$x \in \mathcal{R} \Rightarrow Ax + B\tilde{u}(x) \in \tilde{\mathcal{X}}$$

is the existence of weights  $\Theta \in \mathbb{R}^{n_u \times L}$  and a column vector  $\mathcal{Y} \in \mathbb{R}^{N_x(m+N_w+N_r)}$  such that

$$(I_{N_x} \otimes \mathcal{B})^T \mathcal{Y} \leq \mathcal{F}(\Theta, x_c), \quad (17a)$$

$$(I_{N_x} \otimes \mathcal{A})^T \mathcal{Y} = \text{vec}(\mathcal{E}(\Theta, x_c)^T), \quad (17b)$$

$$\mathcal{Y} \geq 0. \quad (17c)$$

*Proof:* By linearizing  $\tilde{u}(x)$  according to Lemma III.6, using the polytopic representation of  $\tilde{\mathcal{X}}$  and reordering the constraints yields a structure equivalent to (15). Dualizing, dropping the minimum and stacking the dual variables, yields the claim. ■

**Remark III.13** Any convex inner approximation of  $\tilde{\mathcal{X}}$ , defined as  $\tilde{\mathcal{X}} := \{x \in \mathbb{R}^{n_x} \mid H_x x \leq h_x\} \subset \tilde{\mathcal{X}}$ , can be used in Theorem III.12 to impose  $x \in \mathcal{R} \Rightarrow Ax + B\Theta k(x) \in \tilde{\mathcal{X}}$ , yielding a sufficient condition for invariance. This is important in practice, where  $\tilde{\mathcal{X}}$  is usually defined only implicitly.

**Remark III.14 (Invariance for union of regions)** Let  $R$  regions  $\mathcal{R}^i$ ,  $i = 1, \dots, R$  correspond to a covering of  $\tilde{\mathcal{X}}$  as in Remark III.14 and let Theorem III.12 be satisfied for all  $R$  regions with separate  $\mathcal{Y}^i$  and one  $\Theta$ . Then Theorem III.12 holds on  $\tilde{\mathcal{X}}$  and subsequently  $\tilde{\mathcal{X}}$  is an invariant set for (11).

**Remark III.15 (Tractability of (17))** Both the number of additional variables  $\mathcal{Y}$  and the number of constraints introduced in Theorem III.12 is polynomial in the dimensions involved. We have precisely  $N_x(1 + m + N_w + N_r)$  inequality and  $N_x(n_x + L + N n_u)$  equality constraints in  $N_x(m + N_w + N_r)$  variables.

## F. Stability

We will now derive sufficient conditions for (12e). We state convex constraints which can be added to the learning problem to impose that the optimal value function  $V_N^*(\cdot)$ , which, by Lemma II.6, is a Lyapunov function for (7), is also a Lyapunov function for (11). The following results will be useful for the derivation of this result in Theorem III.18.

**Lemma III.16** *Let  $x_i^+(x)$  denote the  $i^{\text{th}}$  component of the successor state of  $x$  satisfying (11). Let furthermore  $a_i$  and  $b_i$  denote the  $i^{\text{th}}$  row of  $A$  and  $B$ , respectively. Then there exist  $\underline{y}_i \in \mathbb{R}_{\geq 0}^{N_w}$  and  $\bar{y}_i \in \mathbb{R}_{\geq 0}^{N_w}$  such that*

$$\underline{x}_i^+(x, \Theta, \underline{y}_i) \leq x_i^+(x) \leq \bar{x}_i^+(x, \Theta, \bar{y}_i) \quad \forall x \in \tilde{\mathcal{X}}, \quad (18a)$$

$$-H_w^T \underline{y}_i = \Theta^T b_i^T, \quad H_w^T \bar{y}_i = \Theta^T b_i^T, \quad (18b)$$

where

$$\begin{aligned} \underline{x}_i^+(x, \Theta, \underline{y}_i) &:= x_i^{+,L}(x, \Theta, x_c) - h_w^T \underline{y}_i, \\ \bar{x}_i^+(x, \Theta, \bar{y}_i) &:= x_i^{+,L}(x, \Theta, x_c) + h_w^T \bar{y}_i, \\ x_i^{+,L}(x, \Theta, x_c) &:= a_i x + b_i \Theta (J_k(x - x_c) + k(x_c)). \end{aligned}$$

*Proof:* We first linearize the control law according to Lemma III.6. Writing an underbound on the  $i^{\text{th}}$  component of the successor state, we have

$$\begin{aligned} \underline{x}_i^+(x, \Theta) &= \min_{w \in \mathbb{W}} a_i x + b_i \Theta (J_k(x - x_c) + k(x_c) + w) \\ &= \underbrace{a_i x + b_i \Theta (J_k(x - x_c) + k(x_c))}_{x_i^{+,L}(x, \Theta, x_c)} + \min_{w \in \mathbb{W}} b_i \Theta w. \end{aligned}$$

Since  $\mathbb{W}$  is a nonempty compact convex set, the minimum exists and is attained. Hence, strong duality of LPs yields

$$\begin{aligned} \min_w b_i \Theta w &= \max_{\underline{y}_i} -h_w^T \underline{y}_i & (19) \\ \text{s.t. } H_w w \leq h_w & \quad \text{s.t. } -H_w^T \underline{y}_i = \Theta^T b_i^T, \\ & \quad \underline{y}_i \geq 0. \end{aligned}$$

Since some lower bound is sufficient, we can drop the maximization in (19), yielding the left inequality in (18a). By a similar construction, the overbound on  $x_i^+$  is obtained by maximizing  $x_i^+(x)$  w.r.t. to  $w$ , dualizing and dropping the minimum. ■

**Corollary III.17 (Set of successor states)** *The set of successor states obtained by overbounding the linearization of the suboptimal control law,*

$$X^+(x, \Theta) := \{x^+ \in \mathbb{R}^{n_x} \mid \underline{x}_j^+ \leq x_j^+ \leq \bar{x}_j^+\},$$

with  $\underline{x}_j^+ \equiv \underline{x}_j^+(x, \Theta, \underline{y}_j)$  and  $\bar{x}_j^+ \equiv \bar{x}_j^+(x, \Theta, \bar{y}_j)$  satisfying Lemma III.16, is a convex outer bound of the true successor state, i.e. we always have that

$$x^+(x, \Theta) := Ax + B\Theta k(x) \in X^+(x, \Theta) \quad \forall x \in \tilde{\mathcal{X}}, \forall \Theta.$$

*Proof:* Follows from the definition of  $X^+(x, \Theta)$ . ■

**Theorem III.18 (Finite convex stability constraints)** *Let  $\tilde{\mathcal{X}}$  be given as in Assumption III.5,  $x_c \in \mathcal{R} \subseteq \tilde{\mathcal{X}}$  be fixed*

and  $\mathcal{R}$  be defined as in Theorem III.9. A sufficient condition for (12e) is the existence of weights  $\Theta \in \mathbb{R}^{n_u \times L}$  such that

$$V_N^*(x^+(x, \Theta)) - V_N^{*,L}(x, x_c) \leq -\varepsilon l(x, 0) \quad (20a)$$

$$\forall x \in \text{extr}(\mathcal{R}), \quad x^+(x, \Theta) \in \text{extr}(X^+(x, \Theta)), \quad (20b)$$

where

$$V_N^{*,L}(x, x_c) := \nabla_x V_N^*(x)|_{x=x_c} (x - x_c) + V_N^*(x_c)$$

is the optimal value function linearized in  $x_c$ .

*Proof:* By linearizing  $V_N^*(x)$  around  $x_c$ , we obtain a sufficient condition for (12e)

$$\begin{aligned} V_N^*(x^+(x, \Theta, w)) - V_N^{*,L}(x, x_c) + \varepsilon l(x, 0) &\leq 0 & (21) \\ \forall x \in \mathcal{R}, \quad \forall w \in \mathbb{W}. \end{aligned}$$

The LHS of (21) is convex in  $x$  and  $w$ , as  $x^+(x, \Theta, w)$  is affine in  $x$  and  $w$  when applying the linearized control law. Maximizing the LHS of (21) over all  $x \in \tilde{\mathcal{X}}$  and  $w \in \mathbb{W}$  yields the worst Lyapunov decrease which should be nonpositive. Note that  $X^+(x, \Theta)$  is a box, hence a compact convex polytope. The maximum of a convex function over a convex polytope is attained at one of the vertices of the polytope, hence it suffices to check (21) at the vertices of  $\mathcal{R}$  and the corresponding vertices of  $X^+(x, \Theta)$ . ■

## G. Tractable loss functions

In order to yield a convex problem, convex loss functions  $\mathcal{L}(\cdot)$  should be used. A common loss function is the squared 2-norm of the residuals  $r_i$ , i.e.  $\mathcal{L}(r_i) := \|r_i\|_2^2$ , yielding a constrained least-squares problem. Many other convex loss functions exist [5]. In particular, regularization terms of type  $\lambda \|\text{vec}(\Theta)\|_1$ ,  $\lambda \in \mathbb{R}_+$ , are often added to facilitate sparse solutions. Here  $\lambda$  allows to trade off sparsity vs. approximation accuracy.

**Assumption III.19 (Convex loss function)** *The loss function  $\mathcal{L}(\cdot)$  in (12a) is chosen such that it is convex on  $\mathbb{U}$ .*

We are now ready to state a tractable formulation of (12).

**Theorem III.20 (Tractable learning problem)** *Let  $\tilde{\mathcal{X}} \subseteq \mathcal{X}_N$  be a polytopic set containing the origin in its interior. Furthermore, let  $\chi_i \in \mathcal{X}_N$ ,  $i = 1, \dots, M$  denote randomly drawn samples. If, for some fixed  $\varepsilon > 0$ , the convex optimization problem*

$$\min_{\Theta, \mathcal{Y}, \underline{y}, \bar{y}} \sum_{i=1}^M \mathcal{L}(r_i) \quad (22a)$$

$$\text{s.t. } (12b), (13), (17), (20), \quad (22b)$$

is feasible, returning an optimal set of weights  $\Theta^*$ , then the closed loop system (11) under the proposed suboptimal control law in (10) with weights  $\Theta^*$  is asymptotically stable with region of attraction  $\tilde{\mathcal{X}}$ .

**Remark III.21 (A posteriori verification)** *Conditions (13), (17) and (20) can also be used for verification of the corresponding system theoretic guarantees for (11) when using the standard learning approach in (9).*

**Remark III.22 (Stability implies invariance)** Feasibility of (21) for  $x_0 = x^+(x, \Theta)$  for fixed  $\Theta$  and  $\forall x \in \tilde{\mathcal{X}}$  implies that  $\tilde{\mathcal{X}}$  is an invariant set for (11). Hence, (17) may also be dropped from the tractable learning problem.

**Remark III.23** The tractable learning problem (22) is a quadratically constrained quadratic program (QCQP). In case the MPC problem is formulated with a linear (1 or  $\infty$ ) norm as the stage and terminal cost, the resulting learning problem is a QP.

We have derived a tractable convex inner approximation of the semi-infinite learning problem (12). If it is feasible, the resulting controller is equipped with the desired system theoretic properties. In case of infeasibility of (12), the control designer can select different or more basis functions, or refine the regions  $\mathcal{R}^i$  where necessary. A good indicator where to refine the tessellation is the size of the linearization error sets  $\mathbb{W}^i$ . Once a feasible solution to (12) has been obtained, more samples can be added to increase approximation accuracy.

In the following, we apply our method of learning MPC controllers to an example system.

#### IV. NUMERICAL EXAMPLE

In this section, we present the results for learning the optimal finite horizon controller of a double integrator system. We employ the LQR state feedback law as terminal controller, i.e.  $\kappa_f(x) = K_{\text{LQR}}x$ . The terminal set  $\mathcal{X}_f$  was chosen as the maximum invariant set under LQR control along with the terminal weight  $V_f(x) = x^T P x$ . The problem data is given by

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix},$$

$$Q = Q_{\text{LQR}} = I_2, \quad R = R_{\text{LQR}} = 1, \quad N = 5,$$

$$P = \begin{bmatrix} 1.8085 & 0.231 \\ 0.231 & 2.6489 \end{bmatrix}, \quad \|x\|_\infty \leq 5, \quad |u| \leq 1.$$

We use the following basis function for the learning:

$$k_i(x) = \left(1 - \frac{\|x - \chi_i\|_2}{\sigma}\right)^3,$$

with  $\sigma = 20$ . This function has been investigated on a 1-dimensional domain e.g. in [6], [19]. We draw  $M = 200$  samples by sampling from a uniform distribution, yielding  $L = M = 200$  basis functions centered at the sample points. We have chosen  $\tilde{\mathcal{X}} := 0.9\mathcal{X}_N$  and a tessellation into 274 regions, out of which 60 lie strictly inside  $\mathcal{X}_f$  (Fig. 1). In the learning problem (22), we use a quadratic loss function without regularization,  $\mathcal{L}(r) = \|r\|_2^2$ , and impose input constraints (13) and invariance constraints (17). The problem has been formulated with YALMIP [25] and the resulting QP is solved by CPLEX in 43s on an Intel Xeon 2.53 GHz machine with 8 cores. Stability has been verified a-posteriori using (20).

Figure 2 shows the feasible set  $\mathcal{X}_N$  of the nominal MPC controller with the random sample points used for learning. We have added the vertices of  $\mathcal{X}_N$  to the sample set. The samples not encircled were assigned an almost zero

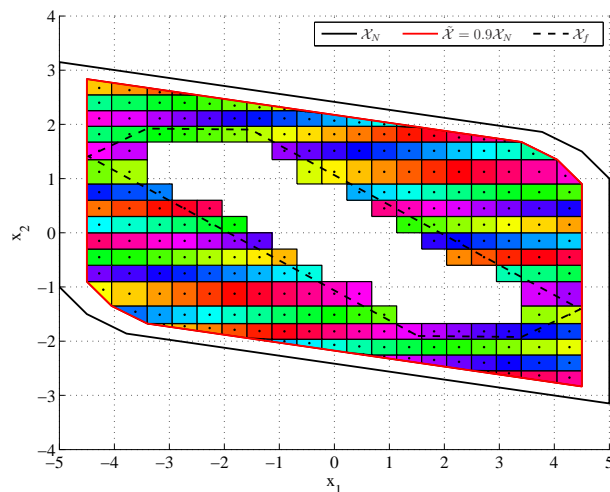


Fig. 1. Tessellation of region of interest,  $\tilde{\mathcal{X}} = 0.9\mathcal{X}_N$ , into 274 regions, 60 of which are strictly inside the terminal set  $\mathcal{X}_f$  and are not shown here. The black dots depict the linearization points  $x_c$ .

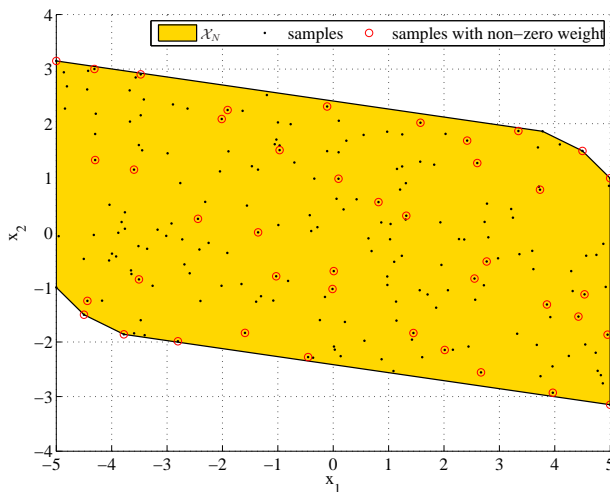


Fig. 2. Samples used for learning. Only the encircled 42 samples out of 200 have been assigned weights with a magnitude bigger than  $10^{-6}$ , i.e. the resulting weight vector is sparse, which significantly speeds up online evaluation of the control law.

weight (less than  $10^{-6}$  in magnitude), which shows that the constrained learning is sparse even without regularization. In this example, only 42 out of the 200 samples suffice to evaluate the nonlinear control law. This result suggests that oversampling does not necessarily increase the complexity of the resulting control law.

The learned controller is a good approximation of the optimal PWA control law, as can be seen in Fig. 3. Due to the use of nonlinear basis functions, the global behavior of the PWA law is captured well, although local approximation errors can be high. This is particularly the case near the hinges of the PWA law. The largest error on a grid of 7480 points is 0.339 at  $x = [2.5 \ -0.315]^T$  with  $u_0^*(x) = -1$  and  $\tilde{u}(x) = -0.661$ .

To investigate the closed loop performance under the approximate control law, trajectories have been simulated

## REFERENCES

- [1] A. Bemporad, F. Borrelli, and M. Morari, "Model predictive control based on linear programming - the explicit solution," *IEEE Trans. Automatic Control*, vol. 47, no. 12, pp. 1974 – 1985, Dec. 2002.
- [2] P. Tondel, T. Johansen, and A. Bemporad, "Computation and approximation of piecewise affine control laws via binary search trees," in *IEEE Conf. Decision and Control*, vol. 3, Dec. 2002, pp. 3144 – 3149.
- [3] C. N. Jones, P. Grieder, and S. Rakovic, "A logarithmic-time solution to the point location problem for parametric linear programming," *Automatica*, vol. 42, no. 12, pp. 2215–2218, Dec. 2006.
- [4] S. Mariethoz, A. Domahidi, and M. Morari, "Sensorless explicit model predictive control of permanent magnet synchronous motors," in *Electric Machines and Drives Conference, 2009. IEMDC '09. IEEE International*, May 2009, pp. 1250–1257.
- [5] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, pp. 199–222, 2004.
- [6] F. Cucker and D.-X. Zhou, *Learning Theory: an Approximation Theory Viewpoint*. Cambridge: Cambridge University Press, 2007.
- [7] A. Alessio and A. Bemporad, "A survey on explicit model predictive control," in *Nonlinear Model Predictive Control*. Springer Berlin / Heidelberg, 2009, vol. 384, pp. 345–369.
- [8] G. Pannochia, J. B. Rawlings, and S. J. Wright, "Fast, large-scale model predictive control by partial enumeration," *Automatica*, vol. 43, pp. 852–860, 2007.
- [9] A. Bemporad and C. Filippi, "Suboptimal explicit mpc via approximate multiparametric quadratic programming," in *IEEE Conf. Decision and Control*, vol. 5, 2001, pp. 4851–4856 vol.5.
- [10] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, January 2002.
- [11] P. Grieder and M. Morari, "Complexity reduction of receding horizon control," in *42nd IEEE Conference on Decision and Control*, Maui (USA), 2003.
- [12] C. N. Jones and M. Morari, "The double description method for the approximation of explicit MPC control laws," in *IEEE Conf. Decision and Control*, Cancun, Mexico, Dec. 2008.
- [13] T. Parisini and R. Zoppoli, "A receding-horizon regulator for nonlinear systems and a neural approximation," *Automatica*, vol. 31, no. 10, pp. 1443 – 1451, 1995.
- [14] A. Bemporad, A. Oliveri, T. Poggi, and M. Storace, "Synthesis of stabilizing model predictive controllers via canonical piecewise affine approximations," in *Decision and Control (CDC), 2010 49th IEEE Conference on*, dec. 2010, pp. 5296–5301.
- [15] M. Kvasnica, J. Löfberg, M. Herceg, L. Cirka, and M. Fikar, "Low-complexity polynomial approximation of explicit MPC via linear programming," in *American Control Conference (ACC), 2010*, 30 2010-july 2 2010, pp. 4713–4718.
- [16] S. Summers, C. N. Jones, J. Lygeros, and M. Morari, "A multiscale approximation scheme for explicit model predictive control with stability, feasibility, and performance guarantees," in *IEEE Conf. on Decision and Control*, Shanghai, China, Dec. 2009.
- [17] M. Vidyasagar, *Nonlinear Systems Analysis*. Prentice Hall, 1993.
- [18] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789 – 814, 2000.
- [19] M. G. Genton, "Classes of kernels for machine learning: A statistics perspective," *Journal of Machine Learning Research*, vol. 2, pp. 299–312, Dec. 2001.
- [20] H. Michalska and D. Mayne, "Moving horizon observers and observer-based control," *IEEE Trans. Automatic Control*, vol. 40, no. 6, pp. 995–1006, Jun. 1995.
- [21] A. Ben-Tal, L. E. Ghaoui, and A. Nemirovski, *Robust Optimization*. Princeton University Press, 2009.
- [22] L. Hermes and J. Buhmann, "Feature selection for support vector machines," in *Int. Conf. Pattern Recognition*, vol. 2, 2000, pp. 712–715.
- [23] F. Borrelli, *Constrained Optimal Control of Hybrid Systems*. Springer, 2003, vol. 290.
- [24] R. B. Kearfott, *Rigorous Global Search: Continuous Problems*. Kluwer Academic Publishers, 1996.
- [25] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. [Online]. Available: <http://users.isy.liu.se/johanl/yalmip>

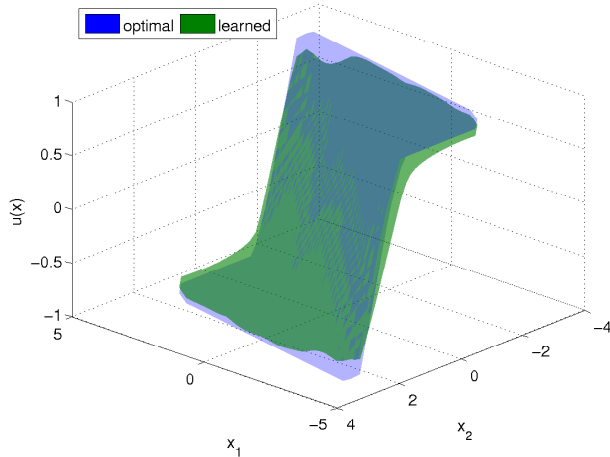


Fig. 3. Optimal (in blue) vs. learned (in green) controller.

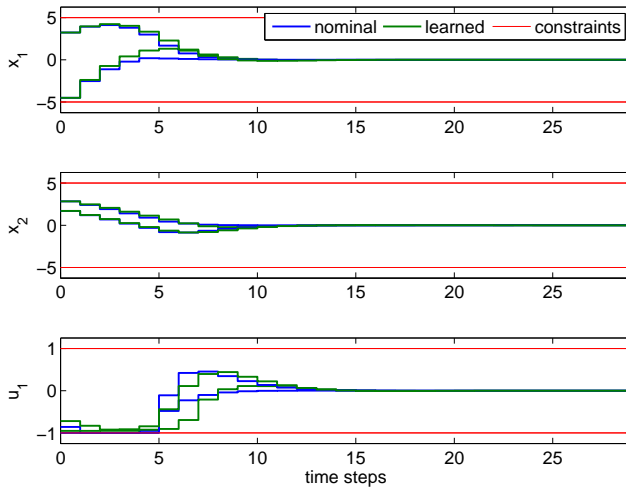


Fig. 4. State and input trajectories for two different initial states.

from 7480 initial conditions (the same points as used above to evaluate the approximation error). Figure 4 depicts two selected trajectories from different initial conditions. The closed loop system is stable, which has been verified a-posteriori with (20). The same result would have been obtained by including the constraints into the learning problem. No state constraint violations occurred for the trajectories. The number of steps needed to reach the terminal region increases for about 10% of the trajectories by one step as compared to the nominal controller.

## V. ACKNOWLEDGEMENTS

The first author gratefully thanks Debasish Chatterjee for insightful discussions at an early stage of this work.

The research leading to these results has received funding from the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement number FP7-ICT-2009-4 248940.