Approximate Optimal Controller Synthesis for Cart-Poles and Quadrotors via Sums-of-Squares

Lujie Yang[1], Hongkai Dai[2], Alexandre Amice[1], Russ Tedrake[1,2]

Abstract— **Sums-of-squares (SOS) optimization is a promising tool to synthesize certifiable controllers for nonlinear dynamical systems. Building upon prior works [1], [2], we demonstrate that SOS can synthesize dynamic controllers with bounded suboptimal performance for various underactuated robotic systems by finding good approximations of the value function. We summarize a unified SOS framework to synthesize both *under-* and *over-* approximations of the value function for continuous-time, control-affine systems, use these approximations to generate approximate optimal controllers, and perform regional analysis on the closed-loop system driven by these controllers. We then extend the formulation to handle hybrid systems with contacts. We demonstrate that our method can generate tight under- and over- approximations of the value function with low-degree polynomials, which are used to provide stabilizing controllers for continuous-time systems including the inverted pendulum, the cart-pole, and the quadrotor as well as a hybrid system, the planar pusher. To the best of our knowledge, this is the first time that a SOS-based time-invariant controller can swing up and stabilize a cart-pole, and push the planar slider to the desired pose. Videos at https://youtu.be/QQR_pPNPeyg; demo code at https://deepnote.com/workspace/lujieyang/project/hjb-sos.**

I. INTRODUCTION

Many interesting robotic tasks, including running, flying, and manipulation are naturally formulated as optimal control problems. Dynamic programming [3], [4] and the Hamilton-Jacobi-Bellman (HJB) equation provide a sound theoretical framework to study the solution to such problems and obtain optimal controllers to the nonlinear dynamical systems. However, the HJB equation is challenging to solve and most methods suffer from an exponential growth of computational complexity, widely recognized as the "curse of dimensionality" [5]. This challenge has motivated the study of numerically tractable methods to provide approximate solutions to the HJB equation [6], [7].

Reinforcement learning has demonstrated great empirical success in finding approximate solutions to the HJB equation which in turn generate highly dynamic controllers. Despite their great empirical success, such methods are rarely, if ever, able to bound the suboptimality of the resulting controllers or the approximation quality of the true cost-to-go function due to their reliance on finite sampling and function approximators.

These drawbacks have motivated the study of a variety of methods that provide such guarantees. In [8], [9], the authors lower bound the value function (the viscosity solution [10],

[1] Massachusetts Institute of Technology
[2] Toyota Research Institute
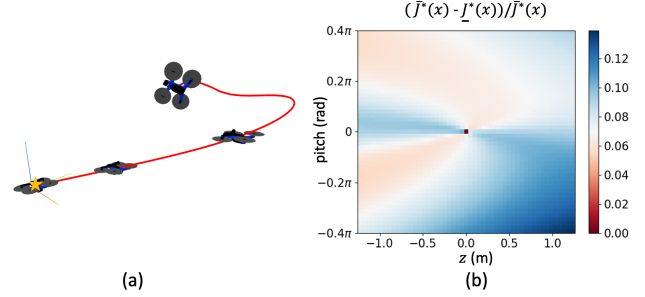Emails: {lujie, amice, russt}@mit.edu, hongkai.dai@tri.global

Fig. 1. Quadrotor: (a) snapshots of the simulation using the stabilizing controller synthesized from value function approximations. The red curve is the quadrotor's trajectory to balance around the origin. Here is an animation. (b) (z, pitch) slice of the relative approximation error of the value function under- and over- estimates ($\underline{J}^*$ and $\bar{J}^*$ respectively).

[11] to the HJB partial differential equation) by relaxing the HJB equation to an inequality and only consider discrete-time systems. Other works [1] consider continuous-time systems and provide under-approximation of the value function based on the moment-SOS hierarchy; this forms the basis of our approach. We show that with careful attention to a few details, their results can be extended to the interesting dynamic regimes. On the other hand, [2] relaxes the optimal control problem to find a global over-approximator on the value function using a policy iteration method. We extend this work by explicitly removing the assumption of an initial, globally stabilizing, polynomial control law, which may not exist even for polynomial dynamics.

In this work, we provide a unified framework to obtain both an under- and over- approximation of the value function within a compact region for continuous-time nonlinear systems. We synthesize tight value function approximations (see Fig. 1) satisfying relaxed HJB conditions using convex optimization, specifically sums-of-squares (SOS) programming [12], which can be solved in polynomial time. The approximation is generated locally over regions of interest for various underactuated robotic systems, which improves the approximation quality when compared to many works seeking global approximators. This is due to the fact that the SOS conditions are stronger [13] over a compact set than over unbounded sets.

While a significant body of work in the controls literature [14], [15] has adopted SOS-based methods for verifying and analyzing the stability and safety of nonlinear systems, their focus has primarily been on Lyapunov or barrier certificates. SOS has been used in the robotics community to cover the state-space with stabilizing [16], [17] and safe [18], [19] controllers. These works emphasize stability or safety rather than

the optimality of the local controller and consider dynamics locally approximated by their Taylor expansion. In this work, we show that using the original robot dynamics without local approximations and incorporating a notion of optimality can dramatically increase the size of the regions over which the resulting SOS-based controllers stabilize the system. Moreover, these prior works require a locally stabilizing initial controller for non-autonomous systems. In contrast, our approach does not require any initial guess on the controller to synthesize a value function under-approximator, which can be used to derive a stabilizing controller for synthesizing a value function over-approximator in all of our experiments.

Our contributions are summarized as follows: 1) we present a strengthened numerical relaxation of the programs from [2], [7] for computing value function estimates that approximately satisfy the HJB over a compact domain. 2) We analyze the local performance of our value approximations by computing inner approximations of both the closed-loop system's region of attraction and the region in which the controllers synthesized from our value function approximations are guaranteed to perform well. 3) We apply our approach to continuous robotic systems such as the inverted pendulum, cart-pole, and quadrotor. We showcase that tight under- and over- estimates of the value function can be synthesized and the corresponding controller derived from these approximators can stabilize the systems for a large region of the state space. 4) We extend the under-approximation formulation to hybrid systems with contacts and validate the framework on the hybrid planar-pusher system. To the best of our knowledge, our method provides the first time-invariant polynomial controllers synthesized with SOS to achieve full cart-pole swing-up and accomplish the planar-pushing task.

II. PROBLEM STATEMENT

We consider a continuous-time control-affine system:

$$\dot{x}(t) = f(x(t), u(t)) = f_1(x(t)) + f_2(x(t))u(t), \quad (1)$$

where f_1 and f_2 are polynomial functions of $x(t) \in \mathbb{R}^{n_x}$ and $u(t) \in \mathcal{U} = \{u | u_{\min} \leq u \leq u_{\max}\} \subset \mathbb{R}^{n_u}$ is the control input.

We consider an optimal control problem of regulating the system to a desired set point $x_d \in \mathbb{R}^{n_x}$ incurring the instantaneous cost

$$l(x, u) = q(x - x_d) + u^T R u,$$

where $q(\bullet)$ is a positive-definite polynomial and R is a diagonal matrix with positive entries. Under the system dynamics (1), the optimal cost-to-go function is positive definite and defined as

$$J^*(x_0) = \min_{u \in \mathcal{U}} \int_0^\infty l(x(t), u(t)) dt, \ x(0) = x_0, \quad (2)$$

while the cost-to-go under a policy π is defined as

$$J^\pi(x_0) = \int_0^\infty l(x(t), \pi(x(t))) \, dt, \ x(0) = x_0. \quad (3)$$

The Hamilton-Jacobi-Bellman equation defines a necessary and sufficient condition for a function J^* to be the optimal cost-to-go [11]:

$$\forall x, \ \min_{u \in \mathcal{U}} \left[l(x, u) + \frac{\partial J^*}{\partial x} f(x, u) \right] = 0. \quad (4)$$

Using (4), the optimal controller can be given in closed-form for the case of diagonal R (proof in Appendix VII-A):

$$\pi^*(x) = \tilde{\pi}(x; J^*) \quad (5)$$

$$\tilde{\pi}(x; J) := \operatorname*{argmin}_{u \in \mathcal{U}} \left[l(x, u) + \frac{\partial J}{\partial x} f(x, u) \right] \quad (6)$$

$$= \operatorname{clamp} \left(-\frac{1}{2} R^{-1} f_2(x)^T \frac{\partial J}{\partial x}^T, u_{\min}, u_{\max} \right), \quad (7)$$

where clamp function is defined as $\operatorname{clamp}(u, u_{\min}, u_{\max}) = \min(\max(u, u_{\min}), u_{\max})$ elementwise.

In general, it is intractable to find an analytic solution $J^*(x)$ to the HJB equation [20], so we aim to find good approximations of the value function, by providing both under- and over-approximators for $J^*(x)$.

A. Global value function under-approximation

It is shown in [21] that $\underline{J}$ is a *global* under-estimate of the value function if it satisfies:

$$\forall x, \ \min_{u \in \mathcal{U}} \left[l(x, u) + \frac{\partial \underline{J}}{\partial x} f(x, u) \right] \geq 0. \quad (8)$$

This under-estimate HJB inequality is equivalent to

$$\forall x, \forall u \in \mathcal{U}, \ l(x, u) + \frac{\partial \underline{J}}{\partial x} f(x, u) \geq 0. \quad (9)$$

A good under-approximator can be found using the optimization program:

$$\max \int_{\mathcal{X}} \underline{J}(x) \, dx \quad (10a)$$

$$\text{s.t.} \ l(x, u) + \frac{\partial \underline{J}}{\partial x} f(x, u) \geq 0, \quad \forall x, \forall u \in \mathcal{U}, \quad (10b)$$

where $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ is a compact region where the value function approximation is encouraged to be accurate. The integral objective is to push up the under-estimate $\underline{J}$.

B. Global value function over-approximation

Similarly, a function $\bar{J}$ is a *global* over-estimate of the value function [2] if it satisfies:

$$\forall x, \ \min_{u \in \mathcal{U}} \left[l(x, u) + \frac{\partial \bar{J}}{\partial x} f(x, u) \right] \leq 0. \quad (11)$$

The over-estimate HJB inequality is equivalent to

$$\forall x, \exists u \in \mathcal{U}, \ l(x, u) + \frac{\partial \bar{J}}{\partial x} f(x, u) \leq 0. \quad (12)$$

Handling the $\exists$ quantifier is much more challenging in general. However, for a fixed control law $u = \pi(x) \in \mathcal{U}$ that globally asymptotically stabilizes x_d, a sufficient condition that $\bar{J}$ is an over-approximator is

$$\forall x, \pi(x) \in \mathcal{U}, \ l(x, \pi(x)) + \frac{\partial \bar{J}}{\partial x} f(x, \pi(x)) \leq 0. \quad (13)$$

A good over-approximator satisfying (12) can be found in the manner of policy evaluation using the optimization program:

$$\min \int_{\mathcal{X}} \bar{J}(x)\,dx \tag{14a}$$

$$\text{s.t. } l(x, \pi(x)) + \frac{\partial \bar{J}}{\partial x} f(x, \pi(x)) \le 0, \quad \forall \pi(x) \in \mathcal{U}. \tag{14b}$$

Similar to the under-estimate program (10), the integral of $\bar{J}(x)$ over $\mathcal{X}$ is used to push down the over-estimate.

It is often impractical to find tight *global* value function under-/over- approximators. In this paper, we restrict to a *local* compact region $\mathcal{X}^{\text{h}} \subset \mathbb{R}^{n_x}$ (h stands for HJB) where low-degree polynomials provide tight approximations and synthesize approximate optimal controllers that generate dynamic behavior for complicated robotic systems.

III. METHOD

The programs (10) and (14) search over the space of all functions and are therefore infinite dimensional. In this section, we discuss strong, finite-dimensional formulations of convex restrictions to (10) and (14) which can find good local under- and over-approximations of the value function over a compact domain using SOS, a convex optimization technique that searches over a parameterized family of polynomials through semidefinite programming [12]. We then discuss how to formulate the programs for rigid-body systems.

A. Regions of interest

When working with compact subsets of a control system, the notion of an *invariant set* of the closed-loop system is important.

Definition 1. Given a controller π and the dynamical system $\dot{x} = f(x, \pi(x))$, a set $\mathcal{I}_\pi$ is an *invariant set* if

$$x(0) \in \mathcal{I}_\pi \Rightarrow x(t) \in \mathcal{I}_\pi, \forall\, t \in [0, \infty).$$

The largest invariant set is the union of all invariant sets.

We now define several regions which will be crucial for formulating the optimization programs to search for value function approximators. Their relationship is illustrated in Fig. 2:

- $\mathcal{X}^{\text{h}}$: the region over which the HJB inequalities (9),(13) are enforced.
- $\mathcal{B}_\pi^{\text{h}}$: the largest invariant set of the closed-loop system driven by a controller π that is contained within $\mathcal{X}^{\text{h}}$.
- $\mathcal{X}$: the region over which we are interested in approximating the value function. This region must be chosen such that $\mathcal{X} \subseteq \mathcal{B}_{\pi^*}^{\text{h}} \subseteq \mathcal{X}^{\text{h}}$.
- $\mathcal{B}_\pi$: the largest invariant set of the closed-loop system driven by a controller π that is contained within $\mathcal{X}$.

B. Local value function under-approximation

We parametrize $\underline{J}$ as a polynomial and require the HJB inequality (9) for the under-estimate to hold on the compact region $\mathcal{X}^{\text{h}}$. With polynomial $\underline{J}$, the nonnegativity constraint

(9) can be enforced by SOS. We can then synthesize a tight under-approximator via the polynomial optimization program

$$\underline{J}^* := \operatorname*{argmax}_{\underline{J}(x) \succeq 0}(10\text{a}) \text{ s.t. } (10\text{b}) \text{ for } x \in \mathcal{X}^{\text{h}}, \tag{15}$$

where $\underline{J}(x) \succeq 0$ is defined as $\underline{J}(x) \ge 0$ for all $x \in \mathcal{X}^{\text{h}}$ and $\underline{J}(x_d) = 0$. This positive-semidefinite requirement on $\underline{J}(x)$ is added because we know that the value function satisfies this condition, and the resulting $\underline{J}^*$ might be able to act as a Lyapunov function to certify the region of attraction for the closed-loop system using program (24). One can easily cast (15) (and the subsequent nonnegativity constraints for x in a semialgebraic set) as a SOS program as shown in Appendix VII-B. An approximate optimal controller can be generated from $\underline{J}^*(x)$:

$$\underline{\pi}(x) = \tilde{\pi}(x; \underline{J}^*) \tag{16}$$

Proposition 1. *The function $\underline{J}^*(x)$ is an under-estimate of the value function for $x \in \mathcal{B}_{\pi^*}^{\text{h}}$.*

The proof of Proposition 1 is in Appendix VII-C.

C. Local value function over-approximation

We similarly seek to enforce the HJB inequality (13) for the over-approximation on $\mathcal{X}^{\text{h}}$. Provided a polynomial control law $\pi_0(x)$ that asymptotically stabilizes $x \in \mathcal{X}^h$ to x_d, we can plug in π_0 as π to search for a good over-estimate using the optimization program :

$$\bar{J}^* := \operatorname*{argmin}_{\bar{J}(x) \succeq 0}(14\text{a}) \text{ s.t. } (14\text{b}) \text{ for } x \in \mathcal{X}^{\text{h}}. \tag{17}$$

An approximate optimal controller derived from $\bar{J}^*(x)$ is

$$\bar{\pi}(x) = \tilde{\pi}(x; \bar{J}^*) \tag{18}$$

Proposition 2. *The function $\bar{J}^*(x)$ is an over-estimate of the value function for $x \in (\mathcal{B}_{\pi_0}^{\text{h}} \cup \mathcal{B}_{\bar{\pi}}^{\text{h}}) \cap \mathcal{B}_{\pi^*}^{\text{h}}$.*

Proof: Starting from $x(0) \in \mathcal{B}_{\pi_0}^{\text{h}}$, integrate the inequality (13) along the trajectory $(x, \pi_0(x))$

$$0 \ge \int_0^\infty [l(x, \pi_0(x)) + \dot{\bar{J}}^*(x)]\,dt$$
$$= \int_0^\infty l(x, \pi_0(x))\,dt + \bar{J}^*(x(\infty)) - \bar{J}^*(x(0))$$
$$\bar{J}^*(x(0)) \ge \int_0^\infty l(x, \pi_0(x))\,dt = J^{\pi_0}(x(0)),$$

where the stabilizing controller π_0 leads to $x(\infty) = x_d$ and $\bar{J}^*(x(\infty)) = 0$. Therefore, we have $\forall x \in \mathcal{B}_{\pi_0}^{\text{h}}, J^{\pi_0}(x) \le \bar{J}^*(x)$. Moreover, since $\bar{\pi}$ is the minimizer of (11), we have that $\forall x \in \mathcal{X}^{\text{h}}, l(x, \bar{\pi}(x)) + \frac{\partial \bar{J}^*}{\partial x} f(x, \bar{\pi}(x)) \le 0$. We can then prove similarly $\forall x \in \mathcal{B}_{\bar{\pi}}^{\text{h}}, J^{\bar{\pi}}(x) \le \bar{J}^*(x)$. Note that $\mathcal{B}_{\pi_0}^{\text{h}} \cup \mathcal{B}_{\bar{\pi}}^{\text{h}}$ is a control invariant set [22].
$\forall x \in \mathcal{B}_{\pi^*}^{\text{h}}, J^*(x) \le J^{\pi_0}(x), J^*(x) \le J^{\bar{\pi}}(x)$ since by definition, the value function leads to the optimal policy and thus incurs smaller cost than any other policies. ∎

If $\pi_0(x) = \text{clamp}(\pi_{\text{poly}}(x), u_{\min}, u_{\max})$ is the saturation of a polynomial control law $\pi_{\text{poly}}(x)$, a piecewise analysis following the approach proposed in [16] should be performed

to impose input limits for the over-approximation. In particular, we enforce the following conditions on the polynomial optimization program (17):

$$
\pi_{\text{poly}}(x) \geq u_{\max}, x \in \mathcal{X}^{\text{h}} \Rightarrow l(x, u_{\max}) + \frac{\partial \bar{J}}{\partial x} f(x, u_{\max}) \leq 0
$$

$$
\pi_{\text{poly}}(x) \leq u_{\min}, x \in \mathcal{X}^{\text{h}} \Rightarrow l(x, u_{\min}) + \frac{\partial \bar{J}}{\partial x} f(x, u_{\min}) \leq 0
$$

$$
u_{\min} \leq \pi_{\text{poly}}(x) \leq u_{\max}, x \in \mathcal{X}^{\text{h}} \Rightarrow
$$
$$
l(x, \pi_{\text{poly}}(x)) + \frac{\partial \bar{J}}{\partial x} f(x, \pi_{\text{poly}}(x)) \leq 0, \tag{19}
$$

which can be incorporated as SOS conditions with additional multipliers.

Remark 1. *A piecewise-polynomial control law can be obtained from a clamped LQR controller around the desired set point or the under-approximation controller $\underline{\pi}(x)$. The stabilizing region of the initial controller affects the maximum allowable size of $\mathcal{X}^h$.*

Remark 2. *$\bar{J}^*$ is a control Lyapunov function [23] on $\mathcal{B}^h_{\pi_0} \cup \mathcal{B}^h_{\bar{\pi}}$ since*

$$
\frac{\partial \bar{J}^*}{\partial x} f(x, \pi_0(x)) \leq -l(x, \pi_0(x)) < 0, \ \forall x \in \mathcal{B}^h_{\pi_0} \backslash \{x_d\}
$$
$$
\frac{\partial \bar{J}^*}{\partial x} f(x, \bar{\pi}(x)) \leq -l(x, \bar{\pi}(x)) < 0, \ \forall x \in \mathcal{B}^h_{\bar{\pi}} \backslash \{x_d\}.
$$

D. Formulation - Rigid-body systems

The dynamics of rigid-body systems can be written as polynomial functions through a change of state variables, see [24], [25] for a complete description. Here we use the simple pendulum as an example.

Example 1. *By denoting the new coordinate variable $s \equiv \sin\theta$, $c \equiv \cos\theta$, the simple pendulum dynamics given in [7][Chapter 2, (1)] with mass m, length l, damping ratio b, and gravity g can be converted to polynomial dynamics*

$$
x = \begin{bmatrix} s \\ c \\ \dot{\theta} \end{bmatrix}, \ \dot{x} = \begin{bmatrix} c\dot{\theta} \\ -s\dot{\theta} \\ -\frac{1}{ml^2}(b\dot{\theta} + mgls) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{ml^2} \end{bmatrix} u. \tag{20}
$$

with the compact domain of x being $\mathcal{X}^h = \{x \in \mathbb{R}^{n_x} | s^2 + c^2 = 1, \dot{\theta}^h_{\min} \leq \dot{\theta} \leq \dot{\theta}^h_{\max}\}$ and $\mathcal{X} = \{x \in \mathbb{R}^{n_x} | s^2 + c^2 = 1, \dot{\theta}_{\min} \leq \dot{\theta} \leq \dot{\theta}_{\max}\}$. The integration in (10a) can be written as

$$
\int_{\dot{\theta}_{\min}}^{\dot{\theta}_{\max}} \int_{s^2 + c^2 = 1} \underline{J}(s, c, \dot{\theta}) \ dx.
$$

IV. REGIONAL ANALYSIS

In this section, we introduce various regions over which we evaluate the quality of our value function approximations and the resulting synthesized controllers. Fig. 2 illustrates an example of the containment relationship between these various sets whose defining properties are summarized in Table I.
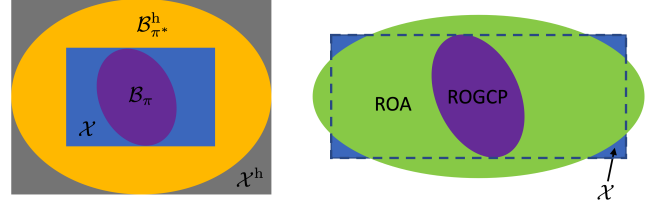

Fig. 2. Visualization of different regions. $\mathcal{X}^h$ is the region where the HJB inequalities are satisfied. $\mathcal{B}^h_{\pi^*}$ is the largest invariant set of the closed-loop system driven by π^* contained within $\mathcal{X}^h$. $\mathcal{X}$ is the region where the value function approximation is encouraged to be accurate. ROGCP illustrates the invariant region where the synthesized controllers have stability and suboptimality guarantees. ROGCP is strictly contained within $\mathcal{X}$ and lies within the closed-loop system's ROA. In general, the closed loop system's ROA and $\mathcal{X}$ have no containment relationship.

A. Region of guaranteed controller performance

Although our value function approximations satisfy the respective HJB inequalities on $\mathcal{X}^h$, the corresponding controllers might not be able to certifiably stabilize the entire region since $\mathcal{X}^h$ is not necessarily an invariant set for the resulting closed-loop system. In addition, the value function over-approximation is on the set $(\mathcal{B}^h_{\pi_0} \cup \mathcal{B}^h_{\bar{\pi}}) \cap \mathcal{B}^h_{\pi^*}$ while the under-approximation is on $\mathcal{B}^h_{\pi^*}$. Unfortunately, the invariant set $\mathcal{B}^h_{\pi^*}$ is difficult to compute exactly. However, since $\mathcal{X} \subseteq \mathcal{B}^h_{\pi^*}$, we can restrict our calculation to $\mathcal{X}$ and try to find good inner approximations of the largest invariant set $\mathcal{B}_{\bar{\pi}} \subseteq \mathcal{X}$ (resp. $\mathcal{B}_{\underline{\pi}}$) where the HJB inequality (13) (resp. (9)) is always satisfied along the trajectories driven by the synthesized controller $\bar{\pi}$ (resp. $\underline{\pi}$).

Definition 2. The region of guaranteed controller performance (ROGCP) for the closed-loop dynamics under $\bar{\pi}$ (resp. $\underline{\pi}$) is defined as the largest invariant set contained in $\mathcal{X}$.

1) Approximation of $\bar{\pi}$'s ROGCP: The sublevel set of a Lyapunov function is commonly used to characterize an invariant set. We can search for an inner approximation of the largest invariant set strictly contained within the compact region $\mathcal{X}$ using a $\bar{\rho}$-sublevel set of $\bar{J}^*$:

$$
\bar{\rho} := \min_{x \in \partial \mathcal{X}} \bar{J}^*(x) \tag{21}
$$
$$
\hat{\mathcal{B}}^{\bar{\rho}}_{\bar{\pi}} := \{x \in \mathcal{X} | \bar{J}^*(x) < \bar{\rho}\}, \tag{22}
$$

where $\partial \mathcal{X}$ is the boundary of $\mathcal{X}$.

Proposition 3. $\hat{\mathcal{B}}^{\bar{\rho}}_{\bar{\pi}}$ is a bounded, connected, invariant set.

We prove Proposition 3 in Appendix VII-D.
Within $\bar{\pi}$'s ROGCP, the following cost-to-go inequalities hold.

Proposition 4. $\forall x \in \hat{\mathcal{B}}^{\bar{\rho}}_{\bar{\pi}}, \ J^*(x) \leq J^{\bar{\pi}}(x) \leq \bar{J}^*(x)$.

The proof of Proposition 4 is similar to Proposition 2.
2) Approximation of $\underline{\pi}$'s ROGCP: Unlike $\bar{J}^*$, $\underline{J}^*$ is not a Lyapunov function by construction and its sublevel set

$$
\hat{\mathcal{B}}^{\rho}_{\underline{\pi}} := \{x \in \mathcal{X} | \underline{J}^*(x) < \rho\}
$$

is not guaranteed to be invariant. Nonetheless, we can search for the region where $\underline{J}^*$ acts as a Lyapunov function by

	HJB inequality satisfied	Invariant
$\mathcal{X}$	yes	not guaranteed
ROGCP	yes	yes
ROA	not guaranteed	yes

TABLE I. HJB inequality satisfaction and set invariance of different regions.

	n_x	$\underline{J}^*$ deg	$\underline{J}^*$ time (s)	$\bar{J}^*$ deg	$\bar{J}^*$ time (s)
pendulum	3	2	0.044	2	0.286
cart-pole	5	6	540	2	6.4
quadrotor	13	2	30	2	1100
pusher	6	2	1410	—	—

TABLE II. Degree and computation time of value function approximations to obtain a stabilizing controller for each task.
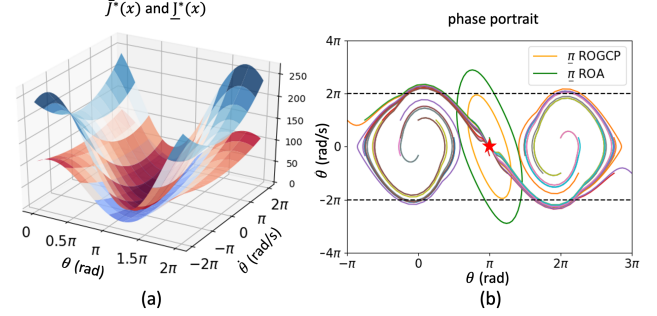

Fig. 3. Inverted pendulum: (a) quadratic value function under- and over-estimate with unconstrained control input; (b) inner approximation of region of guaranteed controller performance/region of attraction and phase portrait of simulating the pendulum using the under-estimate controller with input limits. The orange contour is saturated $\underline{\pi}(x)$'s ROGCP and the green curve displays the closed-loop system's ROA driven by $\underline{\pi}(x)$. The dashed black line is the region of interest $\mathcal{X}$.

enforcing the condition that

$$\frac{\partial \underline{J}^*}{\partial x} f(x, \underline{\pi}(x)) \leq -\epsilon(m(x) - m(x_d))^T (m(x) - m(x_d)) \text{ for } x \in \hat{\mathcal{B}}^\rho_{\underline{\pi}},$$

where ϵ is a small positive constant and $m(x)$ is a vector of monomials in x. The largest $\hat{\mathcal{B}}^\rho_{\underline{\pi}}$ can be obtained with bisection search on ρ in the optimization program:

$$\underline{\rho} := \max_{\rho, \lambda(x)} \rho$$

$$\text{s.t. } \underline{J}^*(x) - \rho \geq 0 \text{ for } x \in \partial\mathcal{X}$$

$$-\frac{\partial \underline{J}^*}{\partial x} f(x, \underline{\pi}(x)) - \epsilon(m(x) - m(x_d))^T (m(x) - m(x_d)) +$$

$$\lambda(x)(\underline{J}^*(x) - \rho) \geq 0 \text{ for } x \in \mathcal{X}, \underline{\pi}(x) \in \mathcal{U}$$

$$\lambda(x) \geq 0 \text{ for } x \in \mathbb{R}^{n_x}.$$

$$(23)$$

For piecewise polynomial $\underline{\pi}$, we can perform the piecewise analysis similar to (19) on program (23).

The following propositions are the direct analogues of Proposition 3 and 4 for $\hat{\mathcal{B}}^\rho_{\underline{\pi}}, \underline{J}^*$ and $J^{\underline{\pi}}$.

Proposition 5. $\hat{\mathcal{B}}^\rho_{\underline{\pi}}$ *is a bounded, connected, invariant set.*

Proposition 6. $\forall x \in \hat{\mathcal{B}}^\rho_{\underline{\pi}}, \ \underline{J}^*(x) \leq J^*(x) \leq J^{\underline{\pi}}(x).$

B. Region of attraction

Notice that $\bar{J}^*$ satisfies the Lyapunov condition on $\mathcal{X}^h$, that is $\dot{\bar{J}}^*(x) < 0$ for all $x \in \mathcal{X}^h \backslash \{x_d\}$. We can find an inner approximation of the region of attraction (ROA) for the closed-loop system driven by $\bar{\pi}$ and verified by $\bar{J}^*$ as a Lyapunov function using the optimization program studied in [25]:

$$\rho^* := \max_{\rho, \lambda(x)} \rho$$

$$\text{s.t. } ((x - x_d)^T (x - x_d))^d (\bar{J}^*(x) - \rho) \quad (24)$$

$$+\lambda(x)\frac{\partial \bar{J}^*}{\partial x} f(x, \bar{\pi}(x)) \geq 0 \text{ for } \bar{\pi}(x) \in \mathcal{U},$$

where d is a positive integer and $\lambda(x)$ is a free polynomial.

Remark 3. *Define the sublevel set* $\mathcal{C}_{\rho^*} := \{x | \bar{J}^*(x) < \rho^*\}$. *Note that the optimization program* (24) *ensures that* $\dot{\bar{J}}^*(x) < 0, \forall x \in \mathcal{C}_{\rho^*} \backslash \{x_d\}$. *Therefore, the connected component of* $\mathcal{C}_{\rho^*}$ *that includes the origin is an invariant set and an inner approximation of the true region of attraction for the closed-loop system under* $\bar{\pi}$.

V. EXPERIMENTAL RESULTS

We synthesize value function approximations and their corresponding controllers for the inverted pendulum, cart-pole, and quadrotor as examples of continuous-time, rigid-body systems. A value function under-estimate and its corresponding controller are obtained for a planar pushing task

with hybrid dynamics. All programs are run on an Intel Core i9-7900X CPU and solved using Mosek. Table II records the function degree and computation time to synthesize a controller that stabilizes a large region of the state space for each task.

A. Inverted pendulum

We synthesize an under- and over-approximation to the value function and their corresponding stabilizing controllers of an inverted pendulum. We aim to swing up and stabilize the pendulum at the upright equilibrium $\theta = \pi, \dot{\theta} = 0$. The region of interest $\mathcal{X}$ is $[s, c, \dot{\theta}] \in [\pm 1, \pm 1, \pm 2\pi]$ and the HJB inequalities are verified in the region $[s, c, \dot{\theta}] \in [\pm 1, \pm 1, \pm 3\pi]$ as $\mathcal{X}^h$. A quadratic value function under-estimate (over-estimate) $\underline{J}^*(x)$ ($\bar{J}^*(x)$) with unconstrained control input is visualized in Fig. 3a. The phase portrait in Fig. 3b demonstrates that starting from any initial state, the system can always converge to the upright equilibrium with input limits 1.8 N $\cdot$ m (the gravity torque is $mgl = 4.9$ N $\cdot$ m). The spiraling trajectories demonstrate that the controller performs nontrivial pumping to swing up the pendulum.

We can search for the inner approximation of ROA where $\underline{J}^*(x)$ serves as a Lyapunov function using the polynomial optimization program (24) by replacing $\bar{J}^*(x)$ and $\bar{\pi}(x)$ with $\underline{J}^*(x)$ and $\underline{\pi}(x)$. In Fig. 3b, the orange contour displays saturated $\underline{\pi}$'s ROGCP and the green curve demonstrates the inner approximation of the closed-loop system ROA, driven by the under-estimate controller and verified by $\underline{J}^*(x)$.

B. Cart-pole

We apply our method to swing up and balance a cart-pole with five states and one actuator [7] around the unstable equi-
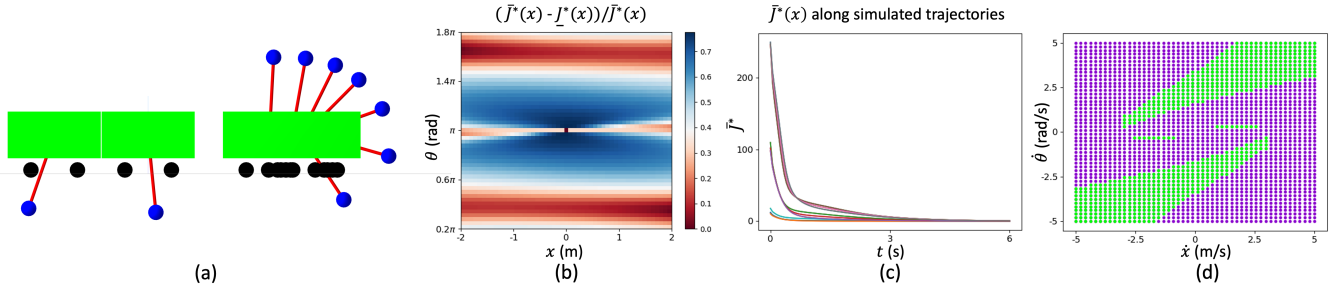
Fig. 4. Cart-pole: (a) snapshots of the full swing-up simulation using $\underline{\pi}(x)$, an animation can be found here; (b) (x, θ) slice of the value function relative approximation error; (c) the value function over-estimate $\bar{J}^*(x)$ always decreases along the trajectories starting within $\bar{J}^*(x)$'s ROA and simulated with $\bar{\pi}(x)$; (d) $(\dot{x}, \dot{\theta})$ slice of sampled initial states that can be stabilized by the SOS/LQR controller.
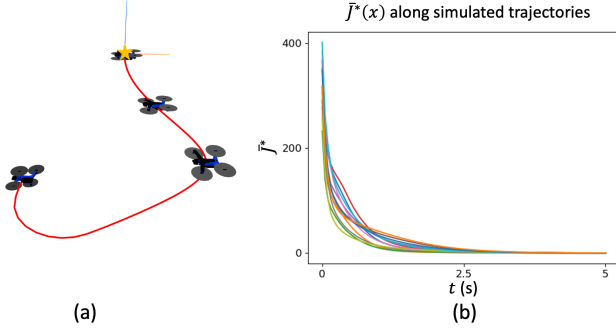

Fig. 5. Quadrotor: (a) snapshots of the simulation using $\bar{\pi}(x)$. (b) The quadratic value function over-approximation $\bar{J}^*(x)$ along the trajectories starting within $\bar{J}^*(x)$'s ROA and simulated with $\bar{\pi}(x)$.

librium $\theta = \pi$. We are interested in the region $[x, s, c, \dot{x}, \dot{\theta}] \in [\pm 2, \pm 1, \pm 1, \pm 5, \pm 5]$ as $\mathcal{X}$ and verify the HJB inequalities in the box region $[x, s, c, \dot{x}, \dot{\theta}] \in [\pm 3, \pm 1, \pm 1, \pm 6, \pm 6]$ as $\mathcal{X}^{\mathrm{h}}$. With input limits $100 \mathrm{~N} \cdot \mathrm{m}$ (cart mass 10 kg, pole mass 1 kg, pole length 0.5 m), a 6-degree value function under-estimate is required to swing up the cart-pole while a quadratic value function over-estimate with an LQR initial controller π_0 suffices to accomplish the task. The successful full swing-up of the cart-pole, accomplished with a nontrivial pumping, is visualized in Fig. 4a. Fig. 4b illustrates the relative approximation error between the value function over- and under- approximator. Fig. 4c shows that $\bar{J}^*(x)$ always decreases along the simulated trajectories from various initial states within the ROA, corroborating that the synthesized over-estimate is a Lyapunov function. In Fig. 4d, we sample 2500 initial states within the box region $[\dot{x}, \dot{\theta}] \in [\pm 5, \pm 5]$ with $[x, \theta] = [0, 0]$, and color them based on whether the SOS/LQR controller succeeds in stabilizing the initial state to the goal (we observe that our SOS controller can stabilize all the initial states in this 2-dimensional slice)

- Purple: SOS succeeds, LQR fails
- Green: both SOS and LQR succeed.

C. Quadrotor

We test our proposed method on a quadrotor model with 13 states (quadrotor orientation is parameterized by a unit quaternion) and 4 actuators [26]. Our goal is to steer the quadrotor to hover at the origin. Let $\mathbf{x} = [x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \omega_x, \omega_y, \omega_z]$ be the 12-dimensional state vector with the quadrotor's orientation parameterized by
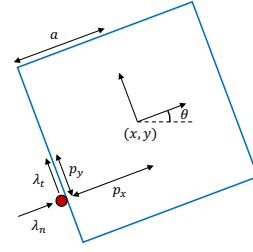

Fig. 6. The planar pusher system with one point pusher and a 4-face slider.

Euler angles, $\mathbf{x}^{\mathrm{h}} = [1, 1, 1, 0.5\pi, 0.2\pi, 0.5\pi, 1, 1, 1, 1, 1, 1]$. The HJB inequalities can be verified within the bounding box region $-\mathbf{x}^{\mathrm{h}} \leq \mathbf{x} \leq \mathbf{x}^{\mathrm{h}}$ as $\mathcal{X}^{\mathrm{h}}$ with input constraints $2.5\frac{mg}{4}$. As visualized in Fig. 5a, the controller obtained from the value function over-estimate with actuator saturation can stabilize the quadrotor with initial conditions as large as $[1, 1, 1, \pi, 0.4\pi, \pi, 8, 8, 8, 8, 8, 8]$.

Our method provides a tight quadratic under- and over-approximation on the true value function for the quadrotor system, as manifested by the small relative approximation error $\frac{\bar{J}^*(x) - \underline{J}^*(x)}{\bar{J}^*(x)}$ in Fig. 1b. We simulate the system starting from various initial conditions within the region of interest and observe that $\bar{J}^*(x)$ decreases along the simulated trajectories in Fig. 5b.

D. Planar pushing

The under-approximation method can be extended to handle certain hybrid dynamical systems. Here, we use the planar pushing system (illustrated in Fig. 6) as an example.

We adopt the quasi-static model [27] for robotic manipulation systems. We model this as a hybrid dynamical system [28] with 4 continuous modes (one mode for every slider face). The state is denoted as $x = [x_{\mathrm{pusher}}, x_{\mathrm{slider}}]$ with a mode index $x_{\mathrm{mode}} \in \{1, 2, 3, 4\}$, where $x_{\mathrm{pusher}}, x_{\mathrm{slider}}$ are the pusher/slider state with continuous values. The control is denoted as $u = [u_{\mathrm{mode}}, u_{\mathrm{pusher}}]$ where u_{mode} takes the discrete mode value, and u_{pusher} takes the continuous value whose meaning changes based on u_{mode} as we will see shortly.

At each time instant, the controller determines whether the pusher should stay on the current face or transition to a different face with u_{mode}. When the contact face remains unchanged (u_{mode} remains unchanged), the control action includes the pusher velocity $u_{\mathrm{pusher}} = v_{\mathrm{pusher}}$. The continuous

dynamics ([29], [30]) are denoted as

$$\dot{x}(t) = f_{x_{\text{mode}}}(x, u), \ x_{\text{mode}} = u_{\text{mode}}, \qquad (25)$$

when $\lim_{\tau \to t^-} u_{\text{mode}}(\tau) = \lim_{\tau \to t^+} u_{\text{mode}}(\tau)$. If the controller chooses to transition to a different mode (the pusher transitions to a different face) with $\lim_{\tau \to t^-} u_{\text{mode}}(\tau) \neq \lim_{\tau \to t^+} u_{\text{mode}}(\tau)$, the mode transition dynamics to post-transition state x^+ are

$$\underbrace{x^+_{\text{pusher}} = u_{\text{pusher}}, \ x^+_{\text{slider}} = x^-_{\text{slider}}}_{x^+ := \Delta(x,u)}, \ x^+_{\text{mode}} = u_{\text{mode}}, \qquad (26)$$

namely, the controller chooses the contact face x^+_{mode} and pusher position x^+_{pusher} after face transition. Due to the quasi-static assumption, the slider remains static during the transition. We assume the transition is instantaneous, with a mode transition cost $l_m(x, u) = |u_{\text{pusher}} - x_{\text{pusher}}|^2 > \varepsilon > 0$.

Since each face transition cost $l_m(x, u)$ is strictly larger than an absolute constant, there will only be a finite number of face transitions in any trajectory with finite cumulative cost. The controller chooses to transition to a mode k different from the current mode when the new state in mode k has a smaller value of the optimal cost-to-go (or its under-estimate $\underline{J}^*$) than that of the current mode under the best continuous action. Such mode transitions are modeled as instantaneous events where the control input causes a discontinuity in the state. We denote the time of mode switch as $t_s, s = 1, \ldots, N$. The optimal control problem is to minimize the cost function

$$\int_0^\infty l(x, u) dt + \sum_{s=1}^N l_m(x(t_s^-), u(t_s)). \qquad (27)$$

It is shown in [31] that $\underline{J}$ is a global under-estimate to the optimal cost-to-go for such hybrid systems if it satisfies (8) for all f_k and:

$$\forall x(t_s^-), \min_u \left[l_m(x(t_s^-), u) + \underline{J}(\Delta(x(t_s^-), u)) - \underline{J}(x(t_s^-)) \right] \geq 0. \quad (28)$$

In addition, friction can be incorporated into our framework with the complementarity formulation [24]

$$v_t \lambda_t \leq 0 \qquad (29a)$$
$$\mu^2 \lambda_n^2 - \lambda_t^2 \geq 0 \qquad (29b)$$
$$(\mu^2 \lambda_n^2 - \lambda_t^2) v_t = 0, \qquad (29c)$$

where v_t is the tangential velocity of the pusher, λ_n, λ_t are the normal and tangential contact forces of the pusher on the slider and μ is the coefficient of friction. In the quasi-static setting, the velocity v_t is part of the control u, and the normal and tangential forces λ_n, λ_t are neither part of the state x nor control u. By treating λ_n and λ_t as new indeterminates, we can define the set $\mathcal{D} = \{(u, \lambda_n, \lambda_t) \mid \text{satisfying (29)}\}$.

Similar to Sec. III-B, we can enforce the HJB inequality (28) on the compact region $\mathcal{X}^{\text{h}}$. We can extend the under-approximation program (15) to the hybrid system case with

friction via the program

$$\underline{J}^* := \underset{\underline{J}(x) \succeq 0}{\text{argmax}} \int_{\mathcal{X}} \underline{J}(x) \, dx$$
$$\text{s.t. for all } k:$$
$$l(x, u) + \frac{\partial \underline{J}}{\partial x} f_k(x, u) \geq 0 \text{ for } x \in \mathcal{X}^{\text{h}}, u \in \mathcal{U}, (u, \lambda_n, \lambda_t) \in \mathcal{D}$$
$$l_m(x, u) + \underline{J}(\Delta(x, u)) - \underline{J}(x) \geq 0 \text{ for } x, \Delta(x, u) \in \mathcal{X}^{\text{h}}, u \in \mathcal{U}. \quad (30)$$

The complementarity constraints on the input preclude an analytic form of the control input as in (5). Instead, a control input $\underline{\pi}$ at a particular state x can be synthesized by solving the nonlinear program for the continuous dynamics:

$$\min_{u \in \mathcal{U}, \lambda_n, \lambda_t} \left[l(x, u) + \frac{\partial \underline{J}^*}{\partial x} f_k(x, u) \right] \qquad (31)$$
$$\text{s.t. } (u, \lambda_n, \lambda_t) \in \mathcal{D}.$$

Similarly, the optimal action of the mode transition dynamics at a state x can be found with the mixed-integer program:

$$\min_{u \in \mathcal{U}} \left[l_m(x, u) + \underline{J}^*(\Delta(x, u)) - \underline{J}^*(x) \right]$$
$$\text{s.t. } \Delta(x, u) \in \mathcal{X}_k \text{ iff } b_k = 1. \ b_k \in \{0, 1\}, \sum_k b_k = 1, \quad (32)$$

where b_k is the binary variable indicating if $\Delta(x, u)$ is in mode k and $\mathcal{X}_k \subset \mathcal{X}^{\text{h}}$ is the compact set of mode k. During online execution, we calculate the optimal continuous control $\underline{\pi}(x)$ and mode switch $\Delta(x, u)$, and choose the control that incurs the smaller cost-to-go.

We synthesize a quadratic value function under-estimate and a controller with actuator saturation for the planar pushing task. The compact region

$$\mathcal{X} = \{x | (p_x + a)(p_x - a)(p_y + a)(p_y - a) = 0,$$
$$-a \leq p_x \leq a, -a \leq p_y \leq a\},$$

requires the pusher to stay on the slider's surface, where p_x, p_y are the slider's position and a is half of the slider's side length. In Fig. 7, we present two trajectories of the circle pusher pushing the blue slider from $[x, y, \theta] = [-0.28, 0.28, 0]$ to the origin. For Fig. 7a, the pusher is restricted to be in contact with the slider's left surface while it is allowed to transition to other faces in Fig. 7b. This is a nontrivial task even for humans, and the mode switches include the finger sticking, sliding up, and sliding down on the slider's left surface (indicated by red, orange and purple circles). Fig. 7b shows a trajectory that involves the finger changing to a different face of the slider in order to push the slider towards the origin. The finger starts from the left surface and pushes the slider to the position right above the origin. Then the finger pusher decides to transition to the top surface (green circle indicates the finger's face change) and push the slider downward.

E. Computation

The computation time of the SOS programs depends both on the system dimension n_x and the degree of the value function approximation. Table II outlines the function degrees and computation time to synthesize the value function
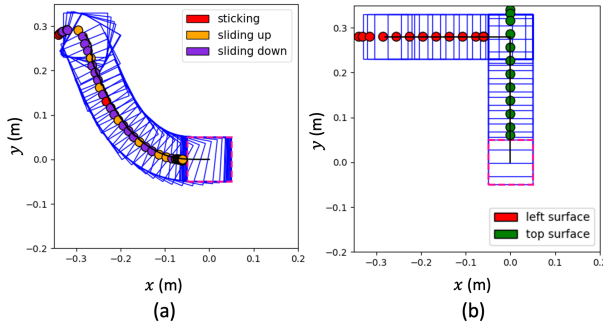
Fig. 7. Planar pushing: (a) snapshots of the simulation using $\underline{J}^*(x)$ starting from $[-0.28, 0.28, 0]$. The black curve is the slider's center of mass trajectory, the circle is the finger pusher and the pink dashed square is the desired slider pose. Different colors show that the finger pusher switches among 3 modes: red, orange and purple denote the finger sticking, sliding up and down along the slider's left face. (b) snapshots of the simulation involving face change using $\underline{J}^*(x)$. After finishing pushing the slider to the right, the finger pusher decides to transition to the top face of the slider (green circle denotes the face change) and then push it towards the origin.

approximations with stabilizing controllers for each task. In general, $\underline{J}^*(x)$ takes less time to synthesize than $\bar{J}^*(x)$ of the same degree because the fixed control law $\pi(x)$ raises the final over-estimate SOS program to higher degrees. However, because the under-estimate is not guaranteed to be a Lyapunov function, a low-degree $\underline{J}^*(x)$ might not be able to stabilize a large region in the state space. For instance, the cart-pole needs a 6th-degree $\underline{J}^*(x)$ to swing up the system and stabilize the state space, resulting in 9-minute computation compared to 6.4 seconds for a quadratic $\bar{J}^*(x)$.

VI. CONCLUSIONS

Building upon prior works on SOS hierarchy, we have summarized a unified approach to synthesizing regional under- and over- approximations to the value function via sums-of-squares programming, and synthesize dynamic controllers for various underactuated robotic systems. We performed local analysis on these value function approximations by computing the region of guaranteed controller performance and an inner approximation of the region of attraction using additional SOS programs. Moreover, we demonstrated how these SOS optimization programs can be adapted for rigid-body systems with input limits and hybrid systems with contacts. In future work, we plan to extend the algorithm to more complex systems and deal with disturbances, model uncertainties, and sensor noise.

ACKNOWLEDGMENT

REFERENCES

[1] J. B. Lasserre, D. Henrion, C. Prieur, and E. Trélat, "Nonlinear optimal control via occupation measures and lmi-relaxations," *SIAM journal on control and optimization*, vol. 47, no. 4, 2008.

[2] Y. Jiang and Z.-P. Jiang, "Global adaptive dynamic programming for continuous-time nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 60, no. 11, 2015.

[3] R. Bellman, "Dynamic programming," *Science*, vol. 153, 1966.

[4] D. Bertsekas, *Dynamic programming and optimal control: Volume I*. Athena scientific, 2012.

[5] R. E. Bellman and S. E. Dreyfus, *Applied dynamic programming*. Princeton university press, 2015, vol. 2050.

[6] D. P. De Farias and B. Van Roy, "The linear programming approach to approximate dynamic programming," *Operations research*, 2003.

[7] R. Tedrake, "Underactuated robotics: Learning, planning, and control for efficient and agile machines," *Course notes for MIT 6.832*, 2009.

[8] T. H. Summers, K. Kunz, N. Kariotoglou, M. Kamgarpour, S. Summers, and J. Lygeros, "Approximate dynamic programming via sum of squares programming," in *European Control Conference*, 2013.

[9] Y. Wang, B. O'Donoghue, and S. Boyd, "Approximate dynamic programming via iterated bellman inequalities," *International Journal of Robust and Nonlinear Control*, vol. 25, no. 10, 2015.

[10] M. G. Crandall, L. C. Evans, and P.-L. Lions, "Some properties of viscosity solutions of hamilton-jacobi equations," *Transactions of the American Mathematical Society*, vol. 282, no. 2, 1984.

[11] M. G. Crandall and P.-L. Lions, "Viscosity solutions of hamilton-jacobi equations," *Transactions of the American mathematical society*, vol. 277, no. 1, 1983.

[12] P. A. Parrilo, "Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization," Ph.D. dissertation, California Institute of Technology, 2000.

[13] K. Schmüdgen, "The moment problem on compact semi-algebraic sets," in *The Moment Problem*. Springer, 2017.

[14] S. Prajna, A. Papachristodoulou, and F. Wu, "Nonlinear control synthesis by sum of squares optimization: A lyapunov-based approach," in *5th Asian control conference*, vol. 1, 2004.

[15] S. Prajna, P. A. Parrilo, and A. Rantzer, "Nonlinear control synthesis by convex optimization," *IEEE Transactions on Automatic Control*, vol. 49, no. 2, 2004.

[16] R. Tedrake, I. R. Manchester, M. Tobenkin, and J. W. Roberts, "Lqr-trees: Feedback motion planning via sums-of-squares verification," *The International Journal of Robotics Research*, vol. 29, no. 8, 2010.

[17] A. Majumdar, "Funnel libraries for real-time robust feedback motion planning," Ph.D. dissertation, MIT, 2016.

[18] A. Clark, "Verification and synthesis of control barrier functions," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 6105–6112.

[19] H. Wang, K. Margellos, and A. Papachristodoulou, "Safety verification and controller synthesis for systems with input constraints," *arXiv preprint arXiv:2204.09386*, 2022.

[20] W. B. Powell, *Approximate Dynamic Programming: Solving the curses of dimensionality*. John Wiley & Sons, 2007, vol. 703.

[21] A. Rantzer and M. Johansson, "Piecewise linear quadratic optimal control," *IEEE transactions on automatic control*, vol. 45, no. 4, 2000.

[22] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, 1999.

[23] E. D. Sontag, "A lyapunov-like characterization of asymptotic controllability," *SIAM journal on control and optimization*, vol. 21, 1983.

[24] M. Posa, M. Tobenkin, and R. Tedrake, "Lyapunov analysis of rigid body systems with impacts and friction via sums-of-squares," in *International conference on Hybrid systems: computation and control*, 2013.

[25] S. Shen and R. Tedrake, "Sampling quotient-ring sum-of-squares programs for scalable verification of nonlinear systems," in *59th Conference on Decision and Control*, 2020.

[26] E. Fresk and G. Nikolakopoulos, "Full quaternion based attitude control for a quadrotor," in *European control conference*, 2013.

[27] J. Trinkle, S. Berard, and J.-S. Pang, "A time-stepping scheme for quasistatic multibody systems," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2005.

[28] M. S. Branicky, V. S. Borkar, and S. K. Mitter, "A unified framework for hybrid control," in *33rd Conference on Decision and Control*, 1994.

[29] F. R. Hogan, E. R. Grau, and A. Rodriguez, "Reactive planar manipulation with convex hybrid mpc," in *IEEE International Conference on Robotics and Automation*, 2018.

[30] M. Bauza, F. R. Hogan, and A. Rodriguez, "A data-efficient approach to precise and controlled pushing," in *Conference on Robot Learning*, 2018.

[31] S. Hedlund and A. Rantzer, "Optimal control of hybrid systems," in *38th Conference on Decision and Control*, 1999.

VII. APPENDIX

A. Optimal controller with input constraints

Since $R = \text{diag}(r_1, \cdots, r_{n_u})$ is diagonal, the quadratic program (QP) (6) can be decomposed into n_u small scalar QPs and solved separately

for $m = 1, \cdots, n_u$:

$$u_m^* = \underset{u_{\min} \leq u_m \leq u_{\max}}{\arg\min} \, r_m u_m^2 + \left[\frac{\partial J}{\partial x} f_2(x) \right]_m u_m, \quad (33)$$

where the subscript m denotes the m-th element of a vector. The optimal solution to this single-variable quadratic cost with interval bounds occurs either at the boundary of the interval, or when the gradient of the quadratic cost is zero, whichever takes smaller quadratic cost. Hence in the scalar case, the m-th element of the constrained optimal controller is

$$u_m^* = \text{clamp}\left(-\frac{1}{2} r_m^{-1} \left[f_2(x) \frac{\partial J}{\partial x} \right]_m, u_{\min}, u_{\max} \right). \quad (34)$$

B. SOS program

For a rigid-body system with s and c (defined in Sec. III-D Example 1) in the state variable, assume that $\mathcal{X}^h$ can be written as a basic semialgebraic set $\mathcal{X}^h = \{x | g_i(x) \leq 0, \forall i = 1, \ldots, n\}$ with $g_i(x)$ polynomials in x and the control input set $\mathcal{U} = \{u | h_j(u) \leq 0, \forall j = 1, \ldots, m\}$ with $h_j(u)$ polynomials in u. The value function under-approximation SOS program is

$$\max_{\underline{J}(x) \succeq 0, \tau(x,u), \bar{\tau}_i(x,u), \tilde{\tau}_j(x,u)} \int_{\mathcal{X}} \underline{J}(x) \, dx$$

$$\text{s.t. } l(x,u) + \frac{\partial \underline{J}}{\partial x} f(x,u) + \tau(x,u)(s^2 + c^2 - 1)$$

$$+ \sum_{i=1}^{n} \bar{\tau}_i(x,u) g_i(x) + \sum_{j=1}^{m} \tilde{\tau}_j(x,u) h_j(u) \text{ is SOS}$$

$$\bar{\tau}_i(x,u), \tilde{\tau}_j(x,u) \text{ are SOS}$$

$$\tau(x,u) \text{ is free polynomial.} \quad (35)$$

C. Proof of Proposition 1

Proof: Starting from $x(0) \in \mathcal{B}_{\pi^*}^h$, integrate the inequality (9) along the trajectory $(x, \pi^*(x))$

$$0 \leq \int_0^\infty [l(x, \pi^*(x)) + \underline{\dot{J}}^*(x)] \, dt$$

$$= \int_0^\infty l(x, \pi^*(x)) \, dt + \underline{J}^*(x(\infty)) - \underline{J}^*(x(0))$$

$$\underline{J}^*(x(0)) \leq \int_0^\infty l(x, \pi^*(x)) = J^*(x(0)),$$

where the optimal controller π^* leads to $x(\infty) = x_d$ and $\underline{J}^*(x(\infty)) = 0$. ∎

D. Proof of Proposition 3

Proof: We begin by noting that $\hat{\mathcal{B}}_{\bar{\pi}}^{\bar{\rho}}$ cannot intersect the boundary of $\mathcal{X}$. Indeed if $x_0 \in \hat{\mathcal{B}}_{\bar{\pi}}^{\bar{\rho}} \cap \partial\mathcal{X}$, then we must have that $\bar{J}^*(x_0) < \bar{\rho}$ which implies that $\bar{\rho}$ is not the minimum value on the boundary of $\mathcal{X}$. Now, seeking a contradiction, suppose $\hat{\mathcal{B}}_{\bar{\pi}}^{\bar{\rho}}$ is disconnected and thus the disjoint union of a non-empty open set $\mathcal{A}$ and an open set $\mathcal{Z}$ containing x_d. Neither of these sets intersect the boundary and therefore $\mathcal{A}, \mathcal{Z} \subset \mathcal{X}$ and so both are bounded. Without loss of generality, we have that the closure $\mathbf{cl}(\mathcal{A})$ is a compact set that does not contain x_d.

Recall that the stage cost $l(x, u) > 0, \, \forall u \in \mathcal{U}, \forall x \in \mathbb{R}^{n_x} \backslash \{x_d\}$. Moreover, $\mathcal{A}$ is an open sublevel set of $\bar{J}^*(x)$ and since $\mathcal{A} \subset \mathcal{X}, \, \dot{\bar{J}}^*(x) \leq -l(x, \pi_0(x)) < 0, \, \forall x \in \mathcal{A}$. Therefore, $\mathcal{A}$ is invariant.

By the extreme value theorem, the minimum of $\bar{J}^*(x)$ over the compact set $\mathbf{cl}(\mathcal{A})$ is attained at some point x_1 in $\mathbf{cl}(\mathcal{A})$, with $\bar{J}^*(x_1)$ strictly less than $\bar{\rho}$. Since all points $x \in \partial\mathcal{A}$ achieve $\bar{J}^*(x) = \bar{\rho}$, we have that $x_1 \in \mathcal{A}$.

However, $\dot{\bar{J}}^*(x_1) < 0$ and so there exists $\hat{x}_1$ such that $\bar{J}^*(\hat{x}_1) < \bar{J}^*(x_1)$. Since $\mathcal{A}$ is invariant, $\hat{x}_1 \in \mathcal{A}$ which contradicts the minimality of x_1. ∎