Sequential Action Control for Tracking of Free Invariant Manifolds

Alex Ansari * **Kathrin Flaßkamp** * **Todd D. Murphey** *

* *Department of Mechanical Engineering, McCormick School of Engineering and Applied Science, Northwestern University, Evanston, IL 60208, USA, (email: alex2011@u.northwestern.edu, kathrin.flasskamp@northwestern.edu, t-murphey@northwestern.edu).*

Abstract:
This paper presents a hybrid control method that controls to unstable equilibria of nonlinear systems by taking advantage of systems' stable manifold of free dynamics. Resulting nonlinear controllers are closed-loop and can be computed in real-time. Thus, we present a computationally efficient approach to optimization-based switching control design using a manifold tracking objective. Our method is validated for the cart-pendulum and the pendubot inversion problems. Results show the proposed approach conserves control effort compared to tracking the desired equilibrium directly. Moreover, the method avoids parameter tuning and reduces sensitivity to initial conditions. Finally, when compared to existing energy based swing-up strategies, our approach does not rely on pre-derived, system-specific switching controllers. We use hybrid optimization to automate switching control synthesis on-line for nonlinear systems.

Keywords: switching control; stable manifolds; manifold tracking; free dynamics; nonlinear control systems; closed-loop control; real-time algorithms

1. INTRODUCTION

This paper presents a hybrid control technique that exploits the free, i.e. uncontrolled, dynamics of nonlinear (control-affine) systems to reach a desired (unstable) equilibrium state. The method generates a closed-loop switching-type control, by which challenging and underactuated control problems such as pendulum inversion can be solved in a numerically efficient way. As a preliminary step, we compute the *stable manifold* for the desired target state based on the free dynamics. This manifold consists of the set of states for which the free dynamics guide the system to the equilibrium (cf. Section 2). Using the recently developed *Sequential Action Control* (SAC, Ansari and Murphey (2015)), we generate a sequence of constrained, least-norm optimal actions that track the nearest points on the manifold on-line, in a receding horizon fashion. By tracking the manifold rather than the desired state, controllers can conserve control effort by leveraging free dynamics as much as possible in reaching the desired state.

The idea of exploiting free dynamics in control problems originates from astro-dynamics. For instance, Marsden and Ross (2006) make use of inherent structures of nonlinear mechanical systems such as invariant (un)stable manifolds to design complex space mission trajectories with minimal control effort. In Flaßkamp et al. (2012), this idea was extended to general mechanical systems by defining motion primitives along the manifolds and sequencing them with control maneuvers into a motion plan. The work (and application in Flaßkamp et al. (2014)) focused on the off-line synthesis of single open-loop plans to serve as initial seeds for optimal control. In contrast, this work aims to directly track stable manifolds, i.e. objects in state space instead of time-dependent trajectories, in closed-loop.

Since manifolds can often only be approximated numerically, we require controllers that are robust to noisy data and, in particular, to derivative information. Control synthesis should also be computationally efficient for real-time requirements. Finally, we seek controllers that take advantage of free dynamics to conserve effort as often as possible, even before the system is close to the stable manifold. For these reasons, our approach is based on a switching control strategy that provides nonzero input only when it is most efficient. This paper achieves the desired control strategy using a specialized SAC controller with a system's free dynamics as a nominal mode and defines an alternate control mode that optimizes a manifold tracking objective. As a benefit, SAC automates the process of determining the switching policy (it decides when to switch and the value of the optimal control mode) on-line for nonlinear systems. In benchmark swing-up examples for a cart-pendulum and a pendubot, SAC rapidly synthesizes constrained switching controls over receding horizons that track stable manifolds with low control effort on-line.

We consider nonlinear control-affine systems,

$$\dot{x}(t) = f(x(t), u(t)) = g(x(t)) + h(x(t))\, u(t), \qquad (1)$$

with state trajectories assumed to be (Lebesgue) square-integrable curves, $x(\cdot) \in \mathcal{L}^2(V, \mathbb{R}^n)$, $V \subset \mathbb{R}$, and piecewise-

⋆ This material is based upon work supported by the National Science Foundation under Grant CMMI 1200321. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

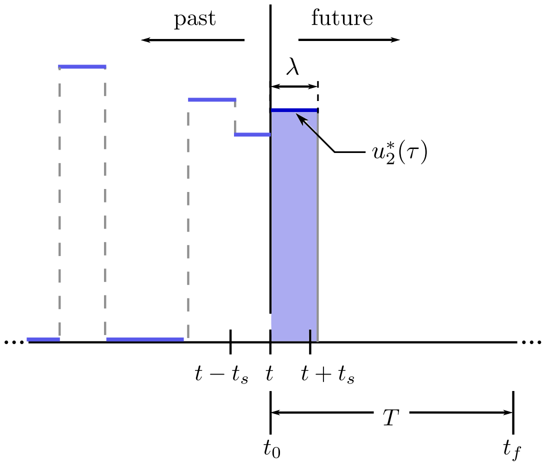

Fig. 1. SAC sequences finite horizon optimal switching control laws in receding horizon fashion. The process results in a piecewise-constant response to state.

constant controls $u(\cdot) : V \mapsto U$, $U \subset \mathbb{R}^m$. A *switched system* is defined by introducing two different modes. *Mode 1* is the default mode and corresponds to the system's free dynamics, i.e. $f_1 : \mathbb{R}^n \mapsto \mathbb{R}^n$ such that

$$f_1(x) = g(x) \ \ \forall x \in \mathbb{R}^n.$$

In *mode 2*, the system is controlled by a *constant control action* $u_2^* \in U$, i.e. the vector field in mode 2 is defined by $f_2 : \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}^n$, such that

$$f_2(x, u_2^*) = g(x) + h(x)\, u_2^* \ \ \forall x \in \mathbb{R}^n, u_2^* \in \mathbb{R}^m.$$

In an on-line process, we use SAC to compute a finite horizon, optimal switching sequence to apply from the current state assuming the mode sequence $\{1, 2, 1\}$. The algorithm determines the optimal value of mode 2 by selecting u_2^*. It chooses an optimal time, τ, to insert this mode, and selects a short duration, λ, which yields mode 2 switching times $\tau \pm \frac{\lambda}{2}$. In receding horizon format, SAC applies controls (based on the optimal sequence) for a brief sampling interval, updates the current state, and repeats the process to obtain the next optimal sequence. Figure 1 shows how the u_2^* from mode 2 are sequenced together into a piecewise-constant response.

Combining SAC with stable manifold tracking objectives resembles energy-based control methods (Fantoni et al. (2000); Xin and Yamasaki (2012); Åström and Furuta (2000); Spong (1995); Zhong and Röck (2001); Shiriaev et al. (2000); Chung and Hauser (1995)), which exploit dynamical structures (e.g. energy conservation and homoclinic orbits of closed-loop systems) for analytical control design. This is in contrast to our method, which is an optimization-based numerical technique that utilizes stable manifold structure of the free dynamics (cf. Section 3.2) and automates switching control synthesis.

Following this introduction, in Section 2 we formally introduce stable manifolds and discuss their numerical approximation. Section 3 provides an overview of SAC based on the switching structure and modes previously described (with f_1 equal to the free dynamics). We present our approach to manifold tracking based on the SAC algorithm. To validate our approach, Sections 4 and 5 present results for two underactuated examples systems. Section 4 formulates the manifold-tracking problem to invert a cart-pendulum system, while Section 5 solves a pendubot swing-up control problem and includes a comparison to prior results. Finally, Section 6 provides concluding remarks and future work.

2. (UN)STABLE MANIFOLDS IN FREE DYNAMICS

Stable and unstable manifolds belong to invariant objects, such as equilibria. They are subsets of the state space that are invariant w.r.t. the flow and form important organizing structures of the global dynamics (consider the separatrix in the phase portrait of a spherical pendulum, for instance). The *stable manifold* of an equilibrium, $\bar{x}$, consists of all points that approach $\bar{x}$ under the system's flow. Analogously, the *unstable manifold* contains all points that show the same behavior if time was reversed.

Formally, we start with a local definition of (un)stable manifolds. Denoting the system's flow by $\Phi(x, t)$, the *local stable manifold* for a neighborhood $U_{\bar{x}} \subset X$ of the state space X is given by (cf. e.g. Guckenheimer and Holmes (1983))

$$W^s_{\text{loc}}(\bar{x}) = \{x \in U_{\bar{x}} \mid \Phi(x, t) \to \bar{x} \text{ for } t \to \infty \\ \text{and } \Phi(x, t) \in U_{\bar{x}} \, \forall t \geq 0\}.$$

For the local unstable manifold, $W^u_{\text{loc}}(\bar{x})$, the definition holds in backward time, i.e. with $t \leq 0$ and $t \to -\infty$. The *stable manifold theorem* (cf. e.g. Guckenheimer and Holmes (1983)) ensures the existence and defines the dimension of the (un)stable manifolds under certain assumptions. For instance, if f is a smooth vector field and $\bar{x}$ a hyperbolic fixed point, the (un)stable manifold is a smooth manifold tangent to the (un)stable eigenspace of the linearization of f at $\bar{x}$ and of the same dimension. Therefore, (un)stable manifolds can be seen as generalizations of the stable and unstable eigenspaces of linear dynamical systems. The *global stable manifold* W^s is governed by the preimages of the flow on $W^s_{\text{loc}}(\bar{x})$, that is

$$W^s(\bar{x}) = \bigcup_{t \leq 0} \Phi(W^s_{\text{loc}}(\bar{x}), t)$$

and, respectively, the global unstable manifold W^u is obtained from images of $W^u_{\text{loc}}(\bar{x})$ under the flow.

An overview of different numerical approaches for the computation of (un)stable manifolds can be found in Krauskopf et al. (2005). In this paper we use the publically available software, GAIO (*Global Analysis of Invariant Objects*, Dellnitz and Junge (2002); Dellnitz et al. (2001)), for manifold approximation. GAIO, like several other numerical packages, iteratively grows the manifold object from a local neighborhood of the equilibrium.

3. SEQUENTIAL ACTION CONTROL

Sequential Action Control is a closed-loop receding horizon style method for nonlinear optimal control problems that has been developed in Ansari and Murphey (2015). Here, we present the method with free dynamics as a nominal mode and formulate manifold tracking objectives.

Consider system dynamics (1), a tracking cost functional to be minimized,

$$J = \int_{t_0}^{t_f} \ell(x(t))\, dt + m(x(t_f)), \tag{2}$$

and the free dynamics as mode 1 of the switched system. Then, SAC consecutively solves a hybrid optimization problem by computing

1. the *schedule* of optimal control values $u_2^*(\cdot) : V \mapsto U$ to which J is maximally sensitive,
2. the optimal time, τ, for *when* to apply $u_2^* \in U$, and
3. the duration, λ, to apply $u_2^* \in U$,

which together define the control input of mode 2 and the switching times from mode 1 to mode 2 and back at $\tau \pm \frac{\lambda}{2}$. Note that with this notation, the mode 2 control is defined based on the schedule of control values, $u_2^*(\cdot)$, and optimal time, τ, such that $u_2^* := u_2^*(\tau)$. We now discuss the three optimization steps of SAC in more detail.

3.1 Control synthesis using free dynamics

The sensitivity of (2) to an infinitesimal insertion of mode 2 at any time τ is provided by the *mode insertion gradient*, denoted by $\frac{dJ}{d\lambda^+}$ (for a background on the mode insertion gradient and its use in hybrid mode scheduling see Caldwell and Murphey (2013); Gonzalez et al. (2010); Egerstedt et al. (2006); Wardi and Egerstedt (2012)). With mode 1 as the free dynamics, the mode insertion gradient,

$$\begin{aligned}\frac{dJ}{d\lambda^+}(\tau, u_2^*) &= \rho^T (f_2(x, u_2^*) - f_1(x))\big|_{t=\tau} \\ &= \rho^T (g(x) + h(x)\, u_2^* - g(x))\big|_{t=\tau} \\ &= \rho(\tau)^T h(x(\tau))\, u_2^*,\end{aligned}$$

measures the effect of applying the control value u_2^* around a time τ as duration $\lambda \to 0^+$. The ρ term is the *adjoint variable*, which is defined by the linear differential equation

$$\begin{aligned}\dot{\rho} &= -D_x \ell(x)^T - D_x f_1(x)^T \rho \\ &= -D_x \ell(x)^T - D_x g(x)^T \rho, \\ \rho(t_f) &= D_x m(x(t_f))^T.\end{aligned}$$

The first optimization problem to find the schedule of optimal control values, is stated as

$$u_2^*(\cdot) := \arg\min_{u(\cdot)} \frac{1}{2} \int_{t_0}^{t_f} \left[\frac{dJ}{d\lambda^+}(t, u(t)) - \alpha_d \right]^2 + \|u(t)\|_R^2 \, dt, \tag{3}$$

with $\alpha_d \in \mathbb{R}^-$ as a design parameter defining the desired sensitivity and $R = R^T > 0$ weighting control effort. As shown in (Ansari and Murphey, 2015, Theorem 1), the solution to (3), can be written in closed-form as

$$u_2^*(t) = \left[\left(h(x)^T \rho \rho^T h(x) + R \right)^{-1} h(x)^T \rho \, \alpha_d \right]_t .$$

In each finite horizon switching time optimization, SAC assumes the mode sequence $\{1, 2, 1\}$. Rather than choosing τ as the current time, SAC provides the option to choose an optimal time, τ, to apply a control from $u_2^*(\cdot)$. This time (along with the duration, λ) specifies when the switch to mode 2 occurs. The SAC algorithm determines the optimal time to apply a control value as a trade-off between the efficiency of control (based on the value of $\frac{dJ}{d\lambda^+}(\cdot, \cdot)$ relative to a norm on control effort), and the time of waiting,

$$\tau := \arg\min_t \frac{dJ}{d\lambda^+}(t, u_2^*(t)) + \|u_2^*(t)\| + (t - t_0)^\beta, \; \beta \in \mathbb{R}^+.$$

Once τ is specified, the next control value, u_2^*, is known. As described in Ansari and Murphey (2015), the control value can be restricted to satisfy box constraints with minimal assumptions and without any additional computation.

Finally, in order to fully specify the times to switch from mode 1 to 2 and back again, SAC computes a duration $\lambda > 0$ to define the switching sequence. While formal switching time optimization can be used to determine an optimal duration, in practice, λ is very short as controls are only applied briefly before the next iteration of finite horizon calculations update the signal (following the receding horizon process). As such, SAC locally approximates the change in tracking cost around τ as

$$\Delta J \approx \frac{dJ}{d\lambda^+}(\tau, u_2^*) \cdot \lambda \approx \alpha_d \cdot \lambda \, .$$

The algorithm then uses a line search to reduce λ from an initial duration until a value is found that provides a change in cost within tolerance of this model (see Ansari and Murphey (2015)).

3.2 Stable manifold tracking objective

If the desired target point is an unstable equilibrium exhibiting a (nontrivial) stable manifold, a control strategy can benefit from this structure by steering the system to (a larger set of) points on this manifold instead of the desired target. However, a numerical optimization-based approach is not guaranteed to exploit this structure automatically, e.g. due to the chosen time horizon in a receding horizon implementation or because of local optima in the nonlinear optimization problem. To overcome the issue, we propose a control method that explicitly incorporates and tracks stable manifolds in a cost functional.

In general, (un)stable manifolds of nonlinear dynamical systems can only be approximated numerically. As the process is computationally costly, we perform this step off-line using GAIO (see Section 2). With a numerical approximation of the stable manifold $W^s(\bar{x})$, the task becomes designing $\ell(x)$ in (2) such that $\ell(x) = 0$ if and only if $x \in W^s(\bar{x})$. The same is true for the final cost $m(x(t_f))$, if present. Possible design choices for these cost functions are discussed in Section 4 and Section 5. Note that due to the numerical manifold approximation, $\ell(x)$ is not generally continuous.

Combining SAC with stable manifold tracking objectives resembles energy-based control methods (Fantoni et al. (2000); Xin and Yamasaki (2012); Åström and Furuta (2000); Spong (1995); Zhong and Röck (2001); Shiriaev et al. (2000); Chung and Hauser (1995)), which have been developed for various single and double pendulum alternatives. In these works, feedback controllers are analytically designed using partial feedback linearization and Lyapunov functions derived from system energy functions. For design and stability analysis, dynamical structures (e.g. homoclinic orbits) of the closed-loop system are exploited. In contrast, we present an optimization-based numerical approach which automates the exploitation of free dynamics and synthesizes switching control laws on-line. Our method is not restricted to energy-preserving systems and works robustly even with coarse approximations of the stable manifold. Also, by combining manifold and state tracking goals, our approach can avoid undesirable convergence to homoclinic orbits.

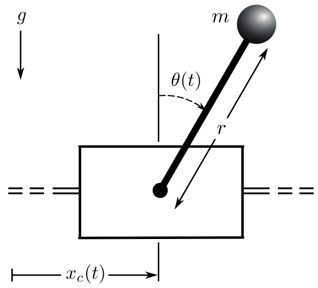

Fig. 2. Model of the acceleration controlled cart-pendulum.

4. ENERGY TRACKING FOR THE CART-PENDULUM

Demonstrating a scenario where stable manifold tracking reduces to energy tracking, this section includes swing-up results for a cart-pendulum. We take advantage of the low state dimension of this example for graphical analysis. Included control and energy phase portraits (Figs. 3 and 5) illustrate how the proposed switching structure yields SAC controllers that leverage free dynamics whether or not manifold tracking goals are included in costs. The example also shows how trajectories evolve through phase space and onto stable manifolds under these different objectives.

This section pertains to the frictionless, acceleration controlled cart-pendulum in Fig. 2, with length $r = 2$ m, mass $m = 1$ kg, and gravity $g = 9.81\ \frac{\mathrm{m}}{\mathrm{s}^2}$. The uncontrolled pendulum is a Hamiltonian system with energy

$$E(\theta, \dot{\theta}) = \frac{1}{2} m r^2 \dot{\theta}^2 + mgr(\cos\theta + 1)$$

such that the free dynamics are

$$\begin{pmatrix} \dot{\theta} \\ \ddot{\theta} \end{pmatrix} = f_1(\theta, \dot{\theta}) = \begin{pmatrix} \dot{\theta} \\ \frac{g}{r}\sin(\theta) \end{pmatrix}.$$

The dynamics of the acceleration controlled cart are defined as $\ddot{x}_c(t) = u(t)$, so that the controlled mode 2 is

$$f_2(\theta, \dot{\theta}, u_2^*) = \begin{pmatrix} \dot{\theta} \\ \frac{g}{r}\sin(\theta) + \frac{u_2^*}{r}\cos(\theta) \end{pmatrix}.$$

4.1 Stable manifold and cost formulation

While the pendulum's downward equilibrium, $(\bar{\theta}, \dot{\bar{\theta}}) = (\pi, 0)$, is stable, the upright equilibrium, $\bar{x} := (\bar{\theta}, \dot{\bar{\theta}}) = (0, 0)$, is not. The eigenvalue spectrum of the linearization at this point consists of one pair of real, stable and unstable eigenvalues. Thus, there are one-dimensional local (un)stable manifolds. For this low-dimensional system, the manifolds can be computed analytically by the energy conservation property, i.e. for $(\theta, \dot{\theta}) \in W^s_{\mathrm{loc}}(\bar{x}) \cup W^u_{\mathrm{loc}}(\bar{x})$ it holds $E(\theta, \dot{\theta}) = E(\bar{x}) = 2mgr$ and we define $\bar{E} := E(\bar{x})$. Locally around $\bar{x}$, the stable manifold is given by

$$W^s_{\mathrm{loc}}(\bar{x}) = \left\{ (\theta, \dot{\theta}) \;\middle|\; \dot{\theta} = -\mathrm{sign}(\theta)\sqrt{2\frac{g}{r}(1 - \cos\theta)} \right\}$$

and the unstable manifold by the same relation with opposite sign. Globally, the stable and unstable manifolds form a so called homoclinic orbit (cf. red curve in Fig. 3).

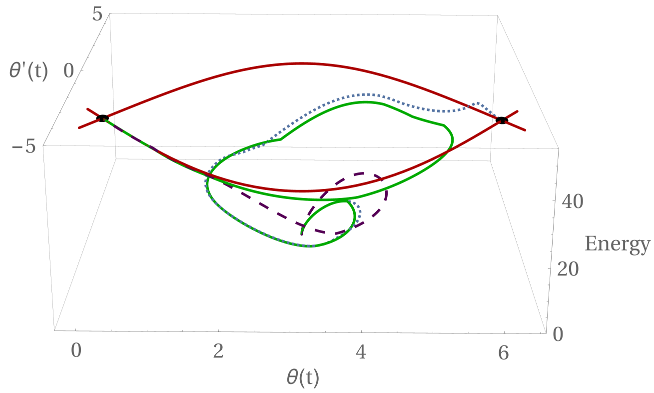

Fig. 3. Manifold tracking solution (green) to the inverted equilibrium, $\bar{x}$, of the cart pendulum system. The stable manifold (homoclinic orbit) is in red, and the inverted equilibrium is indicated by black spheres. For comparison, classical tracking solutions are given: only for longer time horizons (see purple, long dashed curve) comparable results can be obtained, otherwise solutions (cf. blue trajectory, short dashes) requires roughly twice the control effort (see Fig. 4).

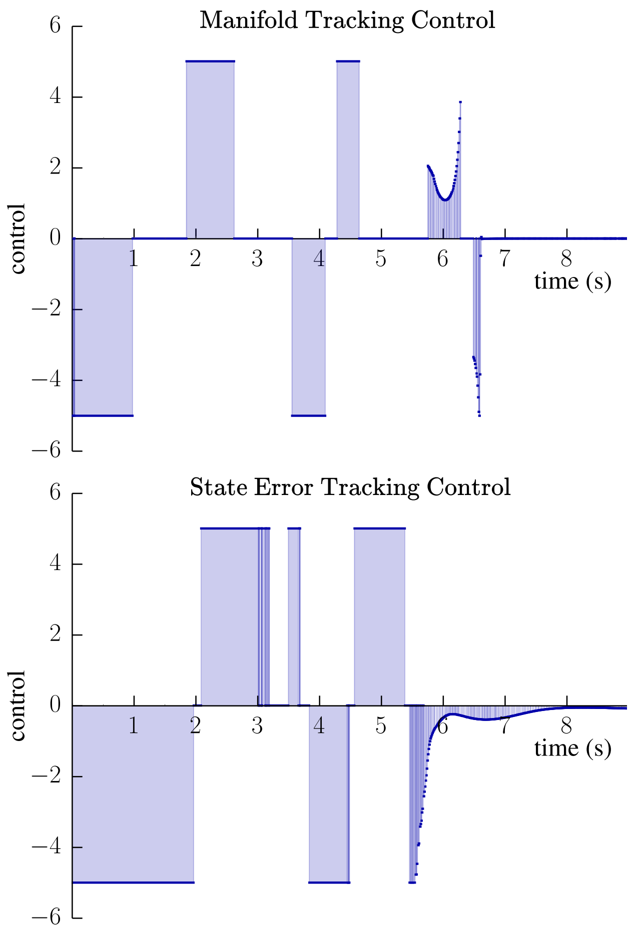

Fig. 4. The top plot corresponds to the green manifold tracking solution from Fig. 3. The bottom plot corresponds to the blue trajectory (short dashed curve in the same figure) that is based on the same parameters but does not use the manifold to invert.

Because of the system's periodicity, we can simplify the manifold tracking problem to tracking the energy of the homoclinic oribit, $\bar{E}$. As simulations revealed integral errors, $\ell(x)$, in (2) were unnecessary provided a terminal energy cost, manifold tracking results use

$$J = J_{\bar{E}} := \frac{1}{2}(E(\theta(t_f), \dot{\theta}(t_f)) - \bar{E})^2.$$

For comparison, Figs. 3 and 4 include trajectory results based on directly tracking the inverted equilibrium state, $\bar{x}$. These results are derived using a state tracking cost,

$$J_{\bar{x}} = \frac{1}{2}\int_{t_0}^{t_f} \|x(t) - \bar{x}\|_Q^2 \, dt + \frac{1}{2}\|x(t_f) - \bar{x}\|_{P_1}^2,$$

with weight matrix $Q = Diag(\{1000, 10\})$ and $P_1 = 0$. [1]

4.2 Discussion of numerical results

Figure 3 includes three different swing-up trajectories in phase space along with the energy $E(\theta, \dot{\theta})$ at each state. Starting from the downwards equilibrium, $(\pi, 0)$, with zero energy, SAC controllers steer each system upwards toward the equilibrium with $\bar{E} = 39.24$. The red curve indicates the stable manifold (homoclinic orbit) of states with energy $\bar{E}$, from which the free dynamics will lead the system to the inverted equilibrium.

The solid green curve in Fig. 3 and control results at the top of Fig. 4 correspond to the SAC trajectory resulting from manifold (energy) tracking cost, $J_{\bar{E}}$. The blue curve (short dashes) in Fig. 3 results when the same SAC controller uses the state error cost, $J_{\bar{x}}$. The controls for this trajectory are included at the bottom of Fig. 4. These controllers are derived with horizons of $T = 0.5$ s, constraints $u \in [-5, 5]\,\frac{\mathrm{m}}{\mathrm{s}^2}$, and $R = 1.0$. The desired rate of cost improvement is specified based on the current cost as $\alpha_d = \gamma J - \alpha_0$. Because closed-form SAC controls (3) are linear state feedback controllers around $\bar{x}$ (see Ansari and Murphey (2015)), one can linearize the dynamics around $\bar{x}$ and choose α_d to provide local stability based on eigenvalue analysis of the closed-loop (LTI) system. Following this approach, we specify $\alpha_0 = -10$ to guarantee stability as $J_{\bar{x}} \to 0$ and apply $\gamma = -5$ to scale α_d based on the current cost when the system is away from equilibrium.

Note that the green manifold tracking solution in Fig. 3 reaches the manifold (red curve) well before $\bar{x}$. Though it converges to $\bar{x}$ at $t \approx 9$ s, the control plot shows SAC ceases control at $t = 6.5$ s. At this point the system is on the manifold and so follows the free dynamics to the goal. In contrast, the state tracking solution does not use the manifold to reach the inverted state. Its control plot shows effort is required until convergence at $t \approx 8 - 9$ s. The manifold tracking solution also better utilizes the free dynamics throughout the trajectory (indicated by the intervals of zero control in Fig. 4). As such, the trajectory uses less effort to invert, with an $\mathcal{L}^2$ norm of 35 compared to 60 for the same controller using $J_{\bar{x}}$. While state tracking costs also yield results that use the manifold, this is only the case for certain parameters and typically longer horizons. For instance, the long dashed purple curve in Fig. 3 tracks $J = J_{\bar{x}}$ with $T = 1.2$ s. The controller reaches the manifold and switches to the free dynamics because the time horizon is sufficient to see it will reach $\bar{x}$ and it will be detrimental to switch to mode 2.

Another benefit we found is that tracking stable manifolds reduces sensitivity to control parameters. Sampling T in the range $[0.05, ..., 1.5]$ s and $\gamma \in [-1, ..., -100]$, we found solutions with only modest qualitative differences (even when we varied the control constraints / norm). In contrast, controlled trajectories derived for $J_{\bar{x}}$ vary dramatically as T changes. [2] Testing of initial conditions for pendulum angle $\theta \in [0, \ldots, 2\pi]$ in increments of 0.1 rad confirmed that manifold tracking is successful in all cases for a variety of horizons. For state error controllers, horizons near $T = 1.2$ s proved best, with convergence from all test conditions. Horizons near $T = 0.5$ s resulted in a failure rate of $\approx 58\%$ and became worse as horizon length further reduced. We emphasize that SAC's receding horizon style calculations are (approximately) linearly dependent on horizon length. Because $J_{\bar{E}}$ facilitates shorter horizons and uses only a terminal cost, manifold tracking allows higher bandwidth feedback and control. [3]

Finally, Fig. 5 shows the state-dependent SAC switching control computed over a grid (0.01 discretization) of the phase plane. [4] Controls track $J_{\bar{E}}$ with the parameters described previously but relaxed constraints, $u \in [-10, 10]\,\frac{\mathrm{m}}{\mathrm{s}^2}$. Streamlines indicate the flow resulting from closed-loop vector field. The controls have a defined switching structure that outlines the stable manifold. SAC applies no control in orange regions, conserving effort by allowing the system to drift along the free dynamics. Locally around the homoclinic orbit and in the neighborhood of the upright equilibrium at $(0, 0)$ and $(2\pi, 0)$, the color gradients indicate controls smoothly transition to zero. [5]

It is noteworthy that the phase plane plot, Fig. 5, shows some structural similarities to the energy-based feedback control laws proposed in Åström and Furuta (2000); Åström et al. (2005). However, in contrast to Åström et al. (2005), SAC automatically generates regions in the pendulum state space where energy is shaped by adding positive or negative damping. Moreover, although the control saturation of SAC generates large regions of "bang-bang-type" control, we do not see the undesirable large energy overshoots as obtained from the minimum-time swing up solution of Åström and Furuta (2000) (cf. Fig. 3).

5. SWING-UP OF THE PENDUBOT

The pendubot is a two-link manipulator, see Fig. 6, with only the first link actuated. The pendubot's states are its angles and velocities, $x = (\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2)$, and its control is the torque about the attachment point of the first link, $u = \tau_1$. This section uses the methods previously described to compute a switching control sequence that

[1] Numerical simulations revealed these matrices resulted in reliable tracking of the desired equilibrium.

[2] Shorter time horizons lead to more direct "pushing" toward the goal and longer horizons yield behaviors similar to energy tracking.

[3] As a benchmark, a typical 10 s trajectory with $T = 0.5$ s and 100 Hz feedback requires ≈ 150 ms to compute using $J_{\bar{E}}$ versus ≈ 250 ms using $J_{\bar{x}}$ on a laptop with an Intel i7 processor.

[4] These plots take seconds to compute and can be used as look-up tables to control low dimensional nonlinear systems to stable manifolds on-line.

[5] Future work will show SAC provides linear state feedback controllers in these regions.

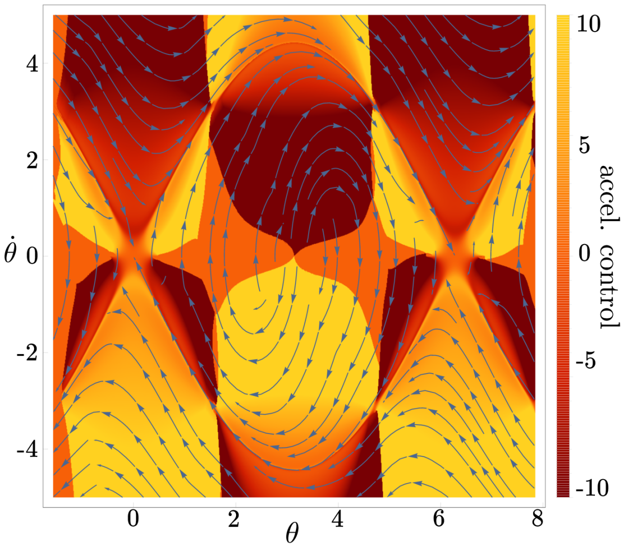

Fig. 5. (Constrained) SAC switching controls computed over a portion of the phase space. Streamlines indicate closed-loop flow. The controls have a defined switching structure that outlines the stable manifold (red curve in Fig. 3). SAC applies no control in orange regions around the stable equilibrium, $(\pi, 0)$, and drifts under free dynamics to conserve effort.

swings the pendubot from the down-down equilibrium, $x_0 = (\pi, 0, \pi, 0)$, to its unstable up-up equilibrium, $\bar{x} := (\bar{\theta}_1, \dot{\bar{\theta}}_1, \bar{\theta}_2, \dot{\bar{\theta}}_2) = (0, 0, 0, 0)$.

The pendubot's dynamics match those from simulations in Ansari and Murphey (2015); Albahkali et al. (2009) and physical experiments in Orlov et al. (2006), with

$$\begin{array}{ll} \mathrm{m}_1 = 1.0367 \text{ kg} & \mathrm{m}_2 = 0.5549 \text{ kg} \\ \mathrm{l}_1 = 0.1508 \text{ m} & \mathrm{l}_2 = 0.2667 \text{ m} \\ \mathrm{l}_{c1} = 0.1206 \text{ m} & \mathrm{l}_{c2} = 0.1135 \text{ m} \\ \mathrm{I}_1 = 0.0031 \text{ kg m}^2 & \mathrm{I}_2 = 0.0035 \text{ kg m}^2 \ . \end{array}$$

We design a SAC controller that performs swing-up control tasks by tracking the pre-computed stable manifold for the inverted equilibrium. Final stabilization is provided by the same LQR controller as in Ansari and Murphey (2015), with state feedback gains

$$K_{lqr} = (-0.23,\ -1.74,\ -28.99,\ -3.86\,).$$

Through numerical simulations, we roughly estimated the region of attraction for the LQR controller and defined the switch to LQR stabilization to occur once $|\theta_1|, |\theta_2| \leq 0.25$ rad and $|\dot{\theta}_1|, |\dot{\theta}_2| \leq 0.5 \frac{\text{rad}}{\text{s}}$. Future work will use formal Sums of Squares methods from Parrilo (2005) to optimize and better define the region of attraction. Results described in this section apply the same control constraints to the LQR controller as enforced for SAC.

5.1 Stable manifold approximation and cost formulation

The inverted equilibrium, $\bar{x}$, is a hyperbolic equilibrium of the pendubot's free dynamics that is structurally equivalent to a frictionless planar double pendulum. The equilibrium has 2-D stable and unstable manifolds (cf. Flaßkamp et al. (2014)), which are computed using GAIO as described in Section 2. Figure 7 shows the box approximation of the stable manifold.

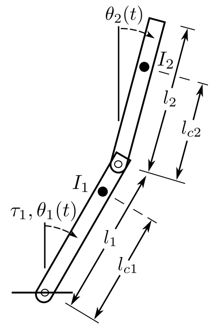

Fig. 6. Model of the pendubot

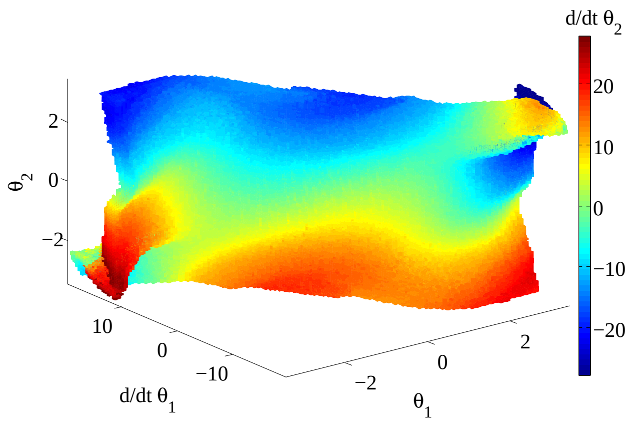

Fig. 7. Box approximation of the stable manifold of the pendubot's inverted equilibrium. The box coloring indicates the value of the fourth coordinate, $\dot{\theta}_2$.

Over the region depicted, states on the stable manifold can be characterized as a function of angular coordinates, $S : \mathbb{R}^2 \mapsto \mathbb{R}^4$, $S(\theta_1, \theta_2) := (\theta_1, S_1(\theta_1, \theta_2), \theta_2, S_2(\theta_1, \theta_2))$, where $S_i : \mathbb{R}^2 \mapsto \mathbb{R}$, $i = 1, 2$, map from angular coordinates to manifold velocities. The cost,

$$\begin{aligned} J(x, u) := J_S = \frac{1}{2} \int_{t_0}^{t_f} \| x(t) - S(\theta_1(t), \theta_2(t)) \|_Q^2 \, dt \\ + \frac{1}{2} \| x(t_f) - S(\theta_1(t_f), \theta_2(t_f)) \|_{P_1}^2 \, , \end{aligned}$$

tracks these stable manifold states. We apply $Q = Diag(\{0, 5, 0, 10\})$ with $P_1 = 0$. After testing 25 initial conditions with angles (θ_1, θ_2) sampled (linearally) in a 0.4 rad windows centered around their (down-down) equilibrium values, these weight matrices lead to the highest inversion rate for a variety of SAC parameters (T and γ).

In practice, $S(\theta_1, \theta_2)$ is obtained from GAIO's discrete manifold representation by sampling over a 64×64 grid of the (θ_1, θ_2) plane and storing the corresponding manifold velocities $(\dot{\theta}_1, \dot{\theta}_2)$ in two 64×64 matrices, S_1 and S_2. Through the same initial condition trails just described, we tested different techniques for approximating derivatives (the adjoint variable from SAC requires the derivative of the integrand and terminal cost of J_S w.r.t. x) and interpolating the coarsely sampled manifold representation. As an unexpected benefit of SAC calculations, tests showed no significant differences when using manifold derivatives, $\frac{dS}{dx}$, versus approximating the cost derivatives by zeroing these terms. This was true when computing derivatives by forwards, backwards, and central differences and for different parameters Q, P_1, T, and γ. Similar experiments showed bilinear interpolation of the manifold performed no better than rounding (using the value of the nearest sample point in the 64×64 grid).

The manifold derivatives likely prove of little use due to noise in GAIO data. Similar noise issues may limit the effectiveness of bilinear data interpolation. In either case, the fact that SAC calculations can be applied to coarsely sampled data and sampled cost functions with only approximated derivatives is ideal in that it reduces computation (no finite differences) and filtering requirements. The following subsection shows that, in spite of these issues, SAC can outperform alternatives and successfully inverts the pendubot with only manifold tracking goals.

5.2 Discussion of numerical results

For comparison with previous pendubot swing-up results from Ansari and Murphey (2015), which are based on a SAC controller performing state tracking with $J_{\bar{x}}$, $Q = Diag(\{100, 0.0001, 200, 0.0001\})$, and $P_1 = 0$, we present results based on the same SAC control parameters. As such we use $\gamma = -15$, $\alpha_0 = 0$, $T = 0.6$ s, $R = 0.1$, $u \in [-7, 7]$ Nm, and receding horizon style control computations occur at a 200 Hz feedback sampling rate.

Figure 8 shows the swing-up solution produced by manifold tracking cost, J_S, with the control parameters described. These results are similar to those achieved using $J_{\bar{x}}$ (in Ansari and Murphey (2015)). Without any state error goal, SAC successfully inverts the pendubot in roughly the same time of ≈ 4 s using the same peak torque (matching physical experiments in Orlov et al. (2006) and half that from simulations in Albahkali et al. (2009)). In both cases (tracking with J_S and $J_{\bar{x}}$), SAC controllers use free dynamics and apply control only when needed.

As for the cart-pendulum, typical pendubot swing-up control laws exhibit sections where no control is applied and the system is allowed to drift (see Fig. 8). Unlike for the cart-pendulum, after sampling initial conditions and a variety of parameter values, we found no reliable differences in control effort according to an $\mathcal{L}^2$ norm (in some cases state tracking outperforms manifold tracking and in other cases we see the opposite). Of more practical importance however,[6] results show manifold tracking requires much less peak control effort for swing-up tasks (better $\mathcal{L}^\infty$ norm). While, we were unable to find parameters to invert the pendubot using less peak torque with the state tracking goal, $J_{\bar{x}}$, parametric exploration revealed several combinations of parameters that invert the pendubot using less peak torque under the manifold tracking goal, J_S. Simply adding the terminal cost $P_1 = Diag(\{0, 15, 0, 10\})$ to J_S, yields inversion with $|u| < 4$ Nm, nearly *half that of the best case* state error tracking results.

Also, similar to the cart-pendulum case, tracking the stable manifold was more robust to both control parameters of the SAC algorithm and to initial conditions. Upon simulating different combinations of time horizon and cost, we found time horizons as low as $T = 0.1$ s would invert the pendulum using only a terminal cost, $P_1 = Diag(\{0, 15, 0, 10\})$ in J_S, with all the same parameters and constraints defined earlier. Again, we were unable to find values of Q, P_1 or γ that allow horizons significantly below $T = 0.6$ s and still invert the pendubot with a state error cost. As mentioned previously, the ability to use shorter horizons (and only a terminal cost) is an advantage to using the stable manifold that allows control calculation and feedback at higher rates.[7] Initial condition tests (described previously) confirm that well chosen parameters can invert the pendubot from all 25 sampled conditions for manifold tracking, while the (best case) parameters identified for tracking $J_{\bar{x}}$ fail for 3 of the 25 sampled conditions.

[6] Peak torque requirements drive motor selection.

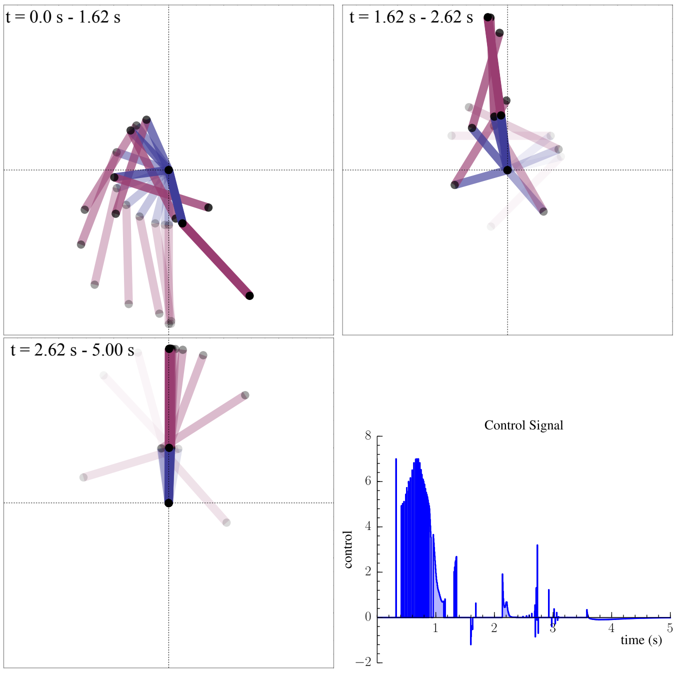

Fig. 8. Manifold-tracking swing-up solution.

6. CONCLUSION

This paper shows that nonlinear control to unstable equilibria can be efficiently computed using a hybrid SAC controller that tracks the stable manifold of free dynamics. Two benchmark underactuated swing-up control examples show the resulting nonlinear controller can be easily computed in real-time and in closed-loop. Using free dynamics (in stable manifold goals and the choice of SAC switching modes), our approach requires less control authority than direct equilibrium tracking in swing-up tasks. Stable manifold tracking provides a larger target set that reduces sensitivity to control parameters and initial conditions. In particular, results show manifold targets allows shorter horizons in SAC receding horizon calculations. Hence, our approach facilitates higher frequency feedback and control. As opposed to existing energy-based strategies, proposed controllers use hybrid optimization to automate synthesis and do not rely on pre-derived analytical strategies.

[7] At 200 Hz, SAC calculations typically range from < 1s to several seconds to compute a 20s swing-up trajectory (depending on parameters) on an Intel i7 laptop. Time horizon shares a (roughly) linear relationship with simulation timing.

To generalize the proposed approach, future work will focus on developing general yet computationally efficient metrics for tracking of "nearest" points on stable manifolds. Furthermore, we intend to evaluate data filtering and low-dimensional storage methods for an optimal representation of manifold data. Finally, in terms of stability, Sums of Squares methods offer a means to numerically define and optimize regions of attraction around time varying trajectories. Because SAC controllers are closed-form linear (time-varying) control laws around desired trajectories, such methods offer numerical means to guarantee stability to stable manifold trajectories.

REFERENCES

Albahkali, T., Mukherjee, R., and Das, T. (2009). Swing-up control of the pendubot: an impulse–momentum approach. *IEEE Transactions on Robotics*, 25(4), 975–982.

Ansari, A. and Murphey, T. (2015). Sequential Action Control: Closed-form optimal control for nonlinear systems. Submitted to IEEE Transactions on Robotics. `http://nxr.northwestern.edu/publications`.

Åström, K.J., Aracil, J., and Gordillo, F. (2005). A new family of smooth strategies for swinging up a pendulum. In P. Piztek (ed.), *Proceedings of the 16th IFAC World Congress, Prague, Czech Republic, July 3-8, 2005*. Elsevier.

Åström, K. and Furuta, K. (2000). Swinging up a pendulum by energy control. *Automatica*, 36(2), 287–295.

Caldwell, T.M. and Murphey, T.D. (2013). Projection-based optimal mode scheduling. In *IEEE Conference on Decision and Control*.

Chung, C.C. and Hauser, J. (1995). Nonlinear control of a swinging pendulum. *Automatica*, 31(6), 851–862.

Dellnitz, M., Froyland, G., and Junge, O. (2001). The algorithms behind GAIO – set oriented numerical methods for dynamical systems. In B. Fiedler (ed.), *Ergodic Theory, Analysis, and Efficient Simulation of Dynamical Systems*, 145–174. Springer.

Dellnitz, M. and Junge, O. (2002). Set oriented numerical methods for dynamical systems. In B. Fiedler (ed.), *Handbook of Dynamical Systems*, chapter 5, 221 – 264. Elsevier Science. doi:10.1016/S1874-575X(02)80026-1.

Egerstedt, M., Wardi, Y., and Axelsson, H. (2006). Transition-time optimization for switched-mode dynamical systems. *IEEE Transactions on Automatic Control*, 51(1), 110–115.

Fantoni, I., Lozano, R., and Spong, M.W. (2000). Energy based control of the pendubot. *IEEE Transactions on Automatic Control*, 45, 725–729.

Flaßkamp, K., Ober-Blöbaum, S., and Kobilarov, M. (2012). Solving optimal control problems by exploiting inherent dynamical systems structures. *Journal of Nonlinear Science*, 22(4), 599–629.

Flaßkamp, K., Timmermann, J., Ober-Blöbaum, S., and Trächtler, A. (2014). Control strategies on stable manifolds for energy-efficient swing-ups of double pendula. *International Journal of Control*, 87(9), 1886–1905.

Gonzalez, H., Vasudevan, R., Kamgarpour, M., Sastry, S.S., Bajcsy, R., and Tomlin, C.J. (2010). A descent algorithm for the optimal control of constrained nonlinear switched dynamical systems. In *ACM Conference on Hybrid Systems: Computation and Control*, 51–60.

Guckenheimer, J. and Holmes, P. (1983). *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, volume 42 of *Applied Mathematical Sciences*. Springer.

Krauskopf, B., Osinga, H.M., Doedel, E.J., Henderson, M.E., Guckenheimer, J., Vladimirsky, A., Dellnitz, M., and Junge, O. (2005). A survey of methods for computing (un)stable manifolds of vector fields. *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, 15(3), 763–791.

Marsden, J.E. and Ross, S.D. (2006). New methods in celestial mechanics and mission design. *Bulletin of the American Mathematical Society*, 43, 43–73.

Orlov, Y., Aguilar, L.T., Acho, L., and Ortiz, A. (2006). Swing up and balancing control of pendubot via model orbit stabilization: Algorithm synthesis and experimental verification. In *45th IEEE Conference on Decision and Control*, 6138–6143.

Parrilo, P.A. (2005). *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. Ph.D. thesis, California Institute of Technology.

Shiriaev, A., Pogromsky, A., Ludvigsen, H., and Egeland, O. (2000). On global properties of passivity-based control of an inverted pendulum. *International Journal of Robust and Nonlinear Control*, 10(4), 283–300.

Spong, M.W. (1995). The swing up control problem for the acrobot. In *IEEE Control Systems Magazine*, 15, 49–55.

Wardi, Y. and Egerstedt, M. (2012). Algorithm for optimal mode scheduling in switched systems. In *American Control Conference*, 4546–4551.

Xin, X. and Yamasaki, T. (2012). Energy-based swing-up control for a remotely driven acrobot: Theoretical and experimental results. *IEEE Trans. Contr. Sys. Techn.*, 20(4), 1048–1056.

Zhong, W. and Röck, H. (2001). Energy and passivity based control of the double inverted pendulum on a cart. In *Proceedings of the 2001 IEEE International conference on Control Applications*. Mexico City, Mexico.