

Hybrid Control for Tracking of Invariant Manifolds

Kathrin Flaßkamp^{a,1,*}, Alex Ansari^{a,2}, Todd D. Murphey^a

^a*Mechanical Engineering Department, Northwestern University, 2145 Sheridan Road,
Evanston, IL 60208*

Abstract

This paper presents a hybrid control method that controls to unstable equilibria of nonlinear systems by taking advantage of systems' free dynamics. The approach uses a stable manifold tracking objective in a computationally efficient, optimization-based switching control design. Resulting nonlinear controllers are closed-loop and can be computed in real-time. 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. The resulting feedback map for the cart-pendulum has a switching structure similar to existing energy based swing-up strategies. We use the Lyapunov function from these prior works to numerically verify local stability for our feedback map. However, unlike the 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: 93C30, 49M05, 49J15

*Corresponding author

Email address: kathrin.flaskamp@uni-bremen.de (Kathrin Flaßkamp)

¹Present address: Optimization and Optimal Control, Center for Industrial Mathematics (ZeTeM), Bibliothekstr. 1, 28359 Bremen, Germany

²Present address: Carnegie Mellon University, Robotics Institute, 5000 Forbes Ave, Pittsburgh, PA 15213

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. 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. Instead of minimizing the distance to the desired target point, this cost functional can be minimized by any trajectory that approaches an arbitrary point on the stable manifold.

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. Using the recently developed control method *Sequential Action Control* (SAC, [1]), 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, SAC controllers can conserve control effort by leveraging free dynamics as much as possible in reaching the desired state.

Contributions. 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. Furthermore, 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. We show how these requirements can be met by

our approach that is based on a switching control strategy providing nonzero input only when it is most efficient. The control strategies use 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. Thus, we
35 present a method to automate the process of determining the switching policy (SAC 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.
40 We compare the resulting feedback map for the cart-pendulum to those from established energy-tracking control methods and adapt their Lyapunov function to numerically show stability of our controller.

A preliminary version of this work has been presented in [2]. The current version extends the results; see in particular Sections 3, 4.3, and 4.4.

45 *Related Works.* The idea of exploiting free dynamics in control problems originates from astro-dynamics. For instance, [3] makes 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 [4], this idea was extended to general mechanical systems by defining motion primitives
50 along the manifolds and sequencing them with control maneuvers into a motion plan. The work (and application in [5]) 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.

55 SAC computes control modes in a receding horizon fashion, thus it shares characteristics with nonlinear model predictive control (NMPC) methods [6, 7, 8]. SAC action plans are applied in open loop, i.e. previously computed actions are applied while current calculations complete as is done in NMPC. However, while NMPC methods have to solve nonlinear optimal control problems by iterative optimization methods repeatedly, the design of SAC actions allows for
60

computationally more efficient solutions and thus, higher bandwidth applications.

The SAC approach is based on ideas from switched system optimization. The mode insertion gradient, i.e. the first-order sensitivity of a cost function to a switch in dynamics of infinitesimal length [9, 10, 11, 12] is used in a novel way to define infinitesimal optimal actions (see Section 2.3 for details).

Combining SAC with stable manifold tracking objectives resembles energy-based control methods ([13, 14, 15, 16, 17, 18, 19]), 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 and automates switching control synthesis. We will recite from these related works in more detail as we analyze the feedback control map for the cart-pendulum system (Section 4.3).

Outline. Following this introduction, we summarize all preliminary material in Section 2: The definition and numerical approximation of stable manifolds, a hybrid system definition for our control method, and an overview of SAC for the hybrid system optimization. Section 3 discusses invariance and stability of our proposed closed-loop manifold-tracking method, as well as general computational aspects. To validate our approach, 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. Preliminaries

This section provides background material related to the manifold tracking method developed in this paper. In Section 2.1, we recite the definition of (un)stable manifolds from dynamical systems' literature and summarize the computation method from [20, 21] that we apply in our examples. We then present a hybrid system definition that is tailored to SAC for manifold tracking

90 in Section 2.2. Section 2.3 summarizes the presentation of SAC from [1] with
a focus on uncontrolled dynamics and the previously derived hybrid system
definition. However, we emphasize the fact that SAC can be used with any
nominal, non-zero control, and we refer to [1] for details.

2.1. Computation of (Un)Stable Manifolds in Free Dynamics

95 Stable and unstable manifolds belong to invariant objects, such as equilibria,
of uncontrolled systems. They are subsets of the state space that are invariant
w.r.t. the flow and form important organizing structures of the global dynamics,
i.e. they separate regions of qualitatively different system behavior, for instance.
The *stable manifold* of an equilibrium, $\bar{x}$, consists of all points that approach $\bar{x}$
100 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. [22])

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

For the local unstable manifold, $W_{\text{loc}}^u(\bar{x})$, the definition holds in backward time,
i.e. with $t \leq 0$ and $t \rightarrow -\infty$. The *stable manifold theorem* (cf. e.g. [22]) en-
sures 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 hyper-
bolic 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 $W_{\text{loc}}^s(\bar{x})$, that is

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

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

The free planar pendulum (cf. also our results on the cart-pendulum in
105 Section 4) gives an example how stable and unstable manifolds organize the
dynamical behavior in state space. Here, the (un)stable manifolds form the so
called *separatrix* that separates the pendulum’s swinging motions with rollovers
from those without. Due to the pendulum’s 2π periodicity, the global stable
and unstable manifolds coincide and are also known as the *homoclinic orbit*. In
110 other words, all points on these manifolds have exactly the right combination
of angle and velocity to reach the equilibrium if left unforced.

An overview of different numerical approaches for the computation of (un)-
stable manifolds can be found in [23]. In this paper we use the publically
available software, GAIO (*Global Analysis of Invariant Objects*, [20, 21]), for
115 manifold approximation. GAIO, like several other numerical packages, itera-
tively grows the manifold object from a local neighborhood of the equilibrium.
Since the unstable manifold is the global attractor relative to prescribed set B
(see [20]), convergence of the approximation to the part of the unstable manifold
is guaranteed. From the definitions above, it follows that the stable manifold
120 is equal to the unstable manifold of the time-reversed system. Therefore, the
stable manifold can be computed by integrating backwards in time starting from
the neighborhood of the equilibrium.

We shortly summarize the idea of GAIO’s continuation method in the fol-
lowing, but we refer to [21] for a detailed description. A large predefined box B
125 in state space (see Fig. 1) and a coarseness for the subdivision of the box have
to be defined beforehand and an equilibrium point has to be specified. The al-
gorithm starts with determining the small initial box containing the target fixed
point. Iteratively, in the continuation step, test points in all found boxes are
mapped forward by a short time integration of the system’s dynamics. Then,
130 all boxes that are hit by one or more test points are marked as “active” and
used for the next continuation step. The algorithm terminates if the part of
the unstable manifold in the predefined large box B is fully covered, i.e. no new
boxes inside B are hit by mapped test points, or if a predefined maximal num-
ber of continuation steps has been performed. The algorithm takes advantage of

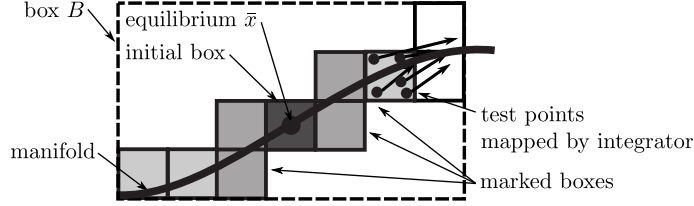


Figure 1: GAIO’s continuation method for the computation of unstable manifolds: Firstly, an initial box (dark gray) containing the equilibrium $\bar{x}$ is marked as “active”. Iteratively, test points in active boxes are mapped forward by an integrator for a short time horizon (continuation step). If mapped test points reach new boxes, these are set to “active” (gray shadings) and will be filled with new test points for the next continuation step. This procedure ends if the entire part of the unstable manifold in the predefined large box B is covered by boxes.

135 the unstable manifold being the attractor of the system: test points which are slightly off the manifold due to the box discretization will be pulled towards the manifold in the following integration step. The size of the small boxes has to be chosen as a trade-off between accuracy of the box covering and computational effort. The default integration method is a fourth-order Runge-Kutta scheme.

140 GAIO can be used for problems with state spaces of moderate dimensions (usually between two and ten), since the computational effort primarily depends on the dimension of the manifold that has to be approximated, and not on the dimension of the full state space.

2.2. Hybrid Control System Definition

145 Our control method aims to conserve control effort by leveraging free dynamics as much as possible, in particular along the stable manifold of the target state. Therefore, we design a hybrid switching control with a nominal mode of zero control and a second mode with a constant control value that is determined within the SAC procedure.

To begin with, we consider nonlinear control-affine systems,

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

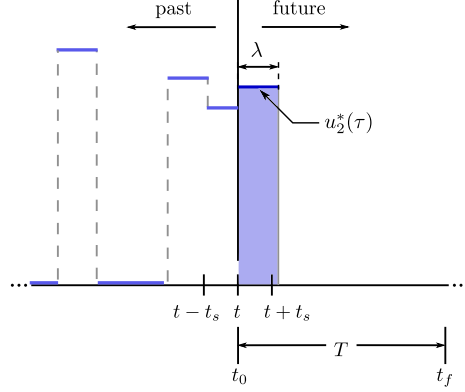


Figure 2: SAC sequences finite horizon optimal switching control laws in receding horizon fashion. The process results in a piecewise-constant response to state.

150 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-constant controls $u(\cdot) : V \mapsto U$, $U \subset \mathbb{R}^m$.

Based on this, we define a *switched system* 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

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

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

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

Switching mode sequences are constrained to be of the form $\{1, 2, 1\}$ and are computed by SAC in a receding horizon online optimization process.

2.3. Sequential Action Control

155 Sequential Action Control is a hybrid control method for nonlinear (control-affine) systems that has been developed in [1].

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

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

and the free dynamics as mode 1 of the switched system. The SAC 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 the optimal controls (based on the $\{1, 2, 1\}$ sequence) for a brief sampling interval, updates the current state, and repeats the process to obtain the next optimal sequence. Figure 2 shows how the u_2^* from mode 2 are sequenced together into a piecewise-constant response.

Thus, SAC consecutively solves hybrid optimization problems 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.

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 [9, 10, 11, 12]). With mode 1 as the free dynamics, the mode insertion gradient,

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

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

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

in which the partial derivatives are evaluated for the nominal trajectory, i.e. the free dynamics solution.

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

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

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 [1, 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,

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

175 Once τ is specified, the next control value, u_2^* , is known. As described in [1], 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 [1]).

3. Closed-Loop Tracking of Stable Manifolds

As previously discussed, our approach to closed-loop tracking of stable manifolds is comprised of the following steps:

1. Computing the stable manifold for the desired equilibrium based on the free dynamics,
2. formulating a manifold tracking cost functional,
3. computing the control signal by SAC on-line.

This section discusses invariance and stability of the resulting control, as well as general computational aspects. The method will be evaluated for benchmark problems in the remainder of this paper.

Theorem 3.1 (Stable Manifold Invariance with SAC). *Assume we use SAC with free dynamics as the default mode and with manifold tracking objectives. That is, we assume $\ell(x) = \|a(x)\|^2$ and $m(x) = \|b(x)\|^2$ in (2), where $a(x) = b(x) = 0$ if and only if $x \in W^s$, i.e. x lies on the stable manifold of the desired final point. Then, SAC returns zero as the optimal schedule of control actions for current states on the manifold, independent of the chosen time horizon.*

Proof: The stable manifold is invariant under the flow of free dynamics. Therefore, the predicted dynamics of mode 1 satisfy $x(t) \in W^s$ for all $t \in [0, t_f]$ assuming the initial state $x(0) \in W^s$. The adjoint's boundary condition is $\rho(t_f) = D_x m(x(t_f))^T = D_x \left(\frac{1}{2} \|b(x(t_f))\|^2 \right) = b(x(t_f)) \circ D_x b(x(t_f)) = 0$, because $b(x(t_f)) = 0$ for $x(t_f) \in W^s$. Similarly, we have that $D_x \ell(x(t))^T = a(x(t)) \circ D_x a(t) = 0$, so that the adjoint equations reduce to purely linear differential equations and the solution is $\rho(t) \equiv 0$ for all $t \in [0, t_f]$. Then, the

205 SAC control gives $u_2^* \equiv 0$ as well, i.e. SAC does not steer the system away from the manifold once the system is on it. $\blacksquare$

Note that the invariance of SAC control along the stable manifold is due to the choice of cost function. The standard SAC implementation that tracks single states does not necessarily have the same behavior. While SAC with manifold tracking is less sensitive of the chosen time horizon, the performance
 210 of SAC with single state tracking crucially depends on the chosen final time t_f . That is, only if t_f is sufficiently large, the state prediction can get close enough to the desired final point so that SAC recognizes that it would be detrimental to switch to the other control mode. This can be observed in the cart-pendulum
 215 example in Section 4 (see the discussion of Fig. 4, in particular).

Theorem 3.2 (Local Stability along Manifolds). *Let $x_d : [t_0, t_f] \rightarrow \mathbb{R}^n$ be a trajectory along the stable manifold and consider the cost function $J = \frac{1}{2} \int_{t_0}^{t_f} \|x(t) - x_d(t)\|_Q^2 dt + \frac{1}{2} \|x(t_f) - x_d(t_f)\|_{P_1}^2$. Then, SAC approximates a linear feedback regulator in the neighborhood of x_d .*

Proof: The manifold trajectory satisfies $\dot{x}_d = f(x_d, u_d)$ with $u_d \equiv 0$, i.e. x_d is a solution for the mode 1 dynamics. Thus, the linearization along this trajectory gives $\dot{x} \approx f(x_d, u_d) + D_x f(x_d, u_d)x_e + D_u f(x_d, u_d)u$ with $x_e(t) = x(t) - x_d(t)$. Define $A(t) := D_x f(x_d, u_d)$, $B(t) := D_u f(x_d, u_d)$, and $u_e(t) = u(t) - u_d(t) = u(t)$. Then, the error dynamics are

$$\dot{x}_e = \dot{x} - \dot{x}_d = A(t)x_e + B(t)u_e = A(t)x_e + B(t)u.$$

Assume that the original system is in the neighborhood of (x_d, u_d) so that the linearization provides a first-order approximation. Note that the cost is then equal to $\frac{1}{2} \int_{t_0}^{t_f} \|x_e(t)\|_Q^2 dt + \frac{1}{2} \|x_e(t_f)\|_{P_1}^2$, i.e. our aim is to steer the error system to zero. We apply SAC with zero nominal control. The final cost gives the boundary condition $\rho(t_f) = P_1 x_e(t_f)$, which, by continuity assumptions, can be extended to $\rho(t) = P(t)x_e(t)$ in some neighborhood $t \leq t_f$. A first order

approximation of the SAC action leads to

$$\begin{aligned} u_2^* &= (B^T(t)P x_e x_e^T P^T B + R)^{-1} (B^T P x_e x_e^T P^T B u_1 + B^T P \alpha_d x_e) \\ &\approx R^{-1} B^T P \alpha_d x_e. \end{aligned}$$

We find P from the adjoint equation $\dot{\rho} = -Qx_e - A^T(t)\rho$ and by differentiating the linear relationship (recall that the adjoint depends on the nominal state trajectory with zero control)

$$\begin{aligned} \dot{\rho} &= \dot{P}x_e + P\dot{x}_e = \dot{P}x_e + P(Ax_e + Bu) = \dot{P}x_e + PAx_e \\ \Rightarrow \quad (-Q - A^T P)x_e &= \dot{P}x_e + PAx_e \\ \Rightarrow \quad -\dot{P} &= Q + A^T P + PA. \end{aligned}$$

220 The linear closed-loop system is given by $\dot{x}_e = (A + BR^{-1}B^T P \alpha_d)x_e$. ■

The stability of the closed-loop system can be determined by analyzing the time-varying state-transition matrix. In particular, this analysis can be used for selecting the design parameters, e.g. α_d , in SAC. As discussed in [1], we
225 could switch to a *continuous feedback law* in the vicinity of x_d . However, unlike in single state tracking, this would still result in time-dependent feedback and therefore, LTV stability analysis has to be applied.

Computational Approach to Stable Manifold Tracking. In general, (un)stable manifolds of nonlinear dynamical systems can only be approximated numeri-
230 cally. As the process is computationally costly, we perform this step off-line using GAIO (see Section 2.1). 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.

In the special case of Hamiltonian systems with one-degree-of-freedom, the
235 energy function can be used to represent one-dimensional manifolds. Combining SAC with stable manifold tracking objectives then resembles energy-based control methods ([13, 14, 15, 16, 17, 18, 19, 24]), which have been developed for

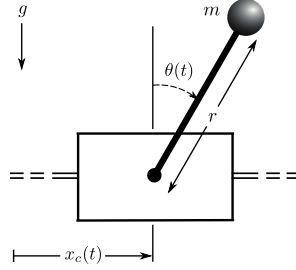


Figure 3: Model of the acceleration controlled cart-pendulum.

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, our approach is optimization-based and thereby, it automates the exploitation of free dynamics and the on-line synthesis of switching control laws. The 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.

We study this approach for the cart-pendulum example in Section 4. For higher-dimensional manifolds, it is not necessarily desirable to restrict to single manifold trajectories x_d as in Theorem 3.2. An alternative design choice based on a projection to the stable manifold velocities is presented in Section 5. Note that due to the numerical manifold approximation, $\ell(x)$ is not generally continuous.

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.

260 Included control and energy phase portraits (Figs. 4 and 6) 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. 3, with length $r = 2$ m, mass $m = 1$ kg, and gravity $g = 9.81 \frac{\text{m}}{\text{s}^2}$. The uncontrolled pendulum is a Hamiltonian system with energy

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

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 $(\theta, \dot{\theta}) \in W_{\text{loc}}^s(\bar{x}) \cup W_{\text{loc}}^u(\bar{x})$ 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

$$W_{\text{loc}}^s(\bar{x}) = \left\{ (\theta, \dot{\theta}) \mid \dot{\theta} = -\text{sign}(\theta) \sqrt{2\frac{g}{r}(1 - \cos \theta)} \right\}$$

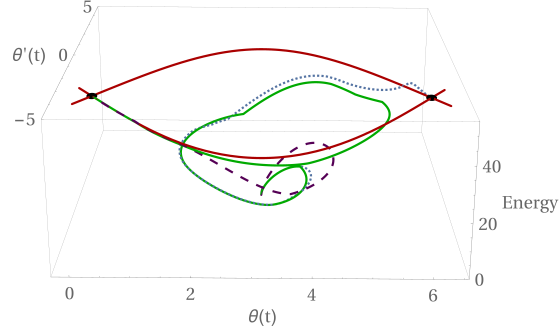


Figure 4: Manifold tracking solution (green) to the inverted equilibrium, $\bar{x}$, of the cart-pendulum system. The stable manifold (homoclinic orbit) is in red, and the inverted equilibrium is indicated by black spheres. For comparison, classical tracking solutions are given: only for longer time 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. 5).

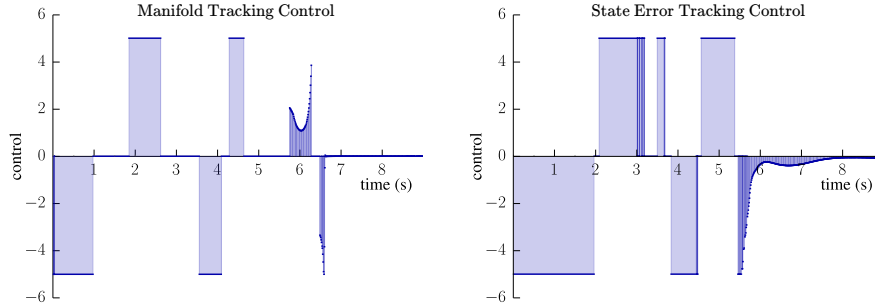


Figure 5: The left plot corresponds to the green manifold tracking solution from Fig. 4. The right plot corresponds to the blue trajectory (short dashed curve in the same figure) that is based on the same parameters but does not use the manifold to invert.

and the unstable manifold by the same relation with opposite sign. Globally, the
 265 stable and unstable manifolds form a so called homoclinic orbit (cf. red curve
 in Fig. 4).

For this system, we can simplify the manifold tracking problem to tracking
 the energy of the homoclinic orbit, $\bar{E}$. Simulations revealed that a running cost
 term in the objective function (2) was unnecessary provided a terminal energy
 cost. Thus, the manifold tracking results use

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

4.2. Comparison to State-Tracking Solutions

For comparison, Figs. 4 and 5 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 $Q = \text{Diag}(\{1000, 10\})$ and $P_1 = 0$. Numerical simulations
 verified that reliable tracking is possible with these matrices.

270 *Testing scenario.* Both SAC controllers are derived with horizons of $T = 0.5$ s,
 constraints $u \in [-5, 5] \frac{\text{m}}{\text{s}^2}$, 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 [1]), one can
 linearize the dynamics around $\bar{x}$ and choose α_d to provide local stability based
 275 on eigenvalue analysis of the closed-loop (LTI) system. Following this approach,
 we specify $\alpha_0 = -10$ to guarantee stability as $J_{\bar{x}} \rightarrow 0$ and apply $\gamma = -5$ to
 scale α_d based on the current cost when the system is away from equilibrium.

Swing-Up Results. Figure 4 includes three different swing-up trajectories in
 phase space along with the energy $E(\theta, \dot{\theta})$ at each state. Starting from the
 280 downwards equilibrium, $(\pi, 0)$, with zero energy, SAC controllers steer each sys-
 tem 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. 4 and control results on the left of Fig. 5 correspond to the SAC trajectory resulting from manifold (energy) tracking cost, $J_{\bar{E}}$. The blue curve (short dashes) in Fig. 4 results when the same SAC controller uses the state error cost, $J_{\bar{x}}$. The controls for this trajectory are included on the right of Fig. 5. Note that the green manifold tracking solution in Fig. 4 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 $t \approx 8$ s. The manifold tracking solution also better utilizes the free dynamics throughout the trajectory (indicated by the intervals of zero control in Fig. 5). 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 $J_{\bar{x}}$. State tracking costs yield results that use the manifold only for certain parameters and typically longer horizons. For instance, the long dashed purple curve in Fig. 4 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.

Enhanced Robustness and Bandwidth of Algorithm. Another benefit we found is that tracking stable manifolds reduces sensitivity to control parameters. Sampling T in the range $[0.05, \dots, 1.5]$ s and $\gamma \in [-1, \dots, -100]$, 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. Whereas shorter time horizons lead to more direct “pushing” toward the goal, longer horizons yield behaviors similar to energy tracking.

Testing of initial conditions for pendulum angle $\theta \in [0, \dots, 2\pi]$ 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 this became worse as horizon length further reduced.

315 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. 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
 320 ≈ 250 ms using $J_{\bar{x}}$ on a laptop with an Intel i7 processor.

4.3. Comparison to Energy-Based Control Laws

The left plot in Figure 6 shows the state-dependent SAC switching control computed over a grid (0.01 discretization) of the phase plane.³ Controls track $J_{\bar{E}}$ with the parameters described previously but relaxed constraints,
 325 $u \in [-10, 10] \frac{\text{m}}{\text{s}^2}$. Streamlines indicate the flow resulting from closed-loop vector field.

We compare this phase plane plot to the energy-based feedback control laws proposed in [15], illustrate in Fig. 7. These feedback laws have been analytically derived from the appropriate Lyapunov functions which, in turn, are based on
 330 the system’s energy function. The bang-bang-type control (upper right plot in Fig. 7) has been chosen so that it changes energy as fast as possible. However, these feedback laws have been designed to provide stability, but do not consider optimality. Opposed to that, our approach is based on optimization in order to develop efficient, admissible control solutions on-line and it does not require
 335 identification of Lyapunov functions.

As can be seen in Fig. 7, SAC as well as the energy-based control-laws all capture the system’s internal structure given by the stable manifold/homoclinic orbit, i.e. they have a defined switching structure that outlines the stable man-

³This plot takes seconds to compute and can be used as a look-up table to control low dimensional nonlinear systems to stable manifolds on-line.

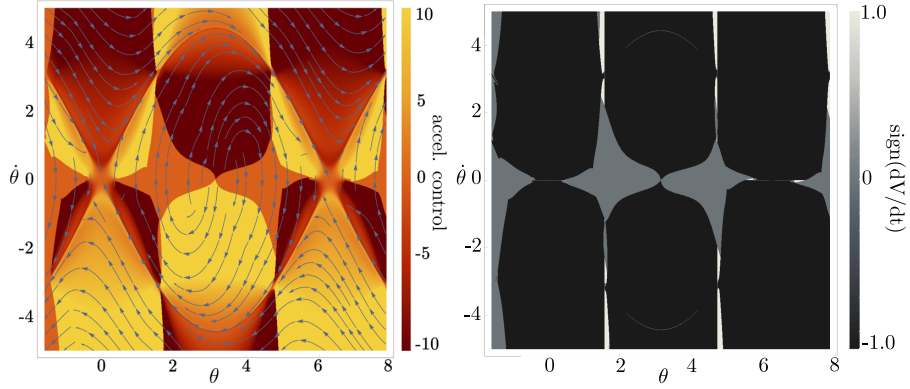


Figure 6: *Left:* SAC switching controls (constrained to $u_2^* \in [-10, 10]$) computed over a portion of the phase space. Streamlines indicate closed-loop flow. The controls have a defined switching structure that outlines the stable manifold (red curve in Fig. 4). SAC applies no control but drifts under the free dynamics in orange regions. *Right:* Sign structure of the Lyapunov function derivative showing stability over large regions of state space.

ifold. Except for the bang-bang control, all controls go to zero on the stable manifold.

The SAC approach 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. Although our choice of SAC control constraints generates large regions of “bang-bang-type” control apart from that (i.e. dark red and light yellow regions with $u = \pm u_{max}$), we do not see the undesirable energy overshoots as obtained from the minimum-time swing up solution of [15] (compare also the green trajectory depicted in Fig. 4).

4.4. Stability Analysis of SAC Feedback Map

We use the Lyapunov function proposed in [15] for the energy-based feedback laws (see Fig. 7) to analyze the stability of the cart-pendulum feedback. That is, we take the Lyapunov function $V = \frac{(E - \bar{E})^2}{2}$, which has the derivative

$$\frac{d}{dt}V = (E - \bar{E})mr\dot{\theta}\cos(\theta)u.$$

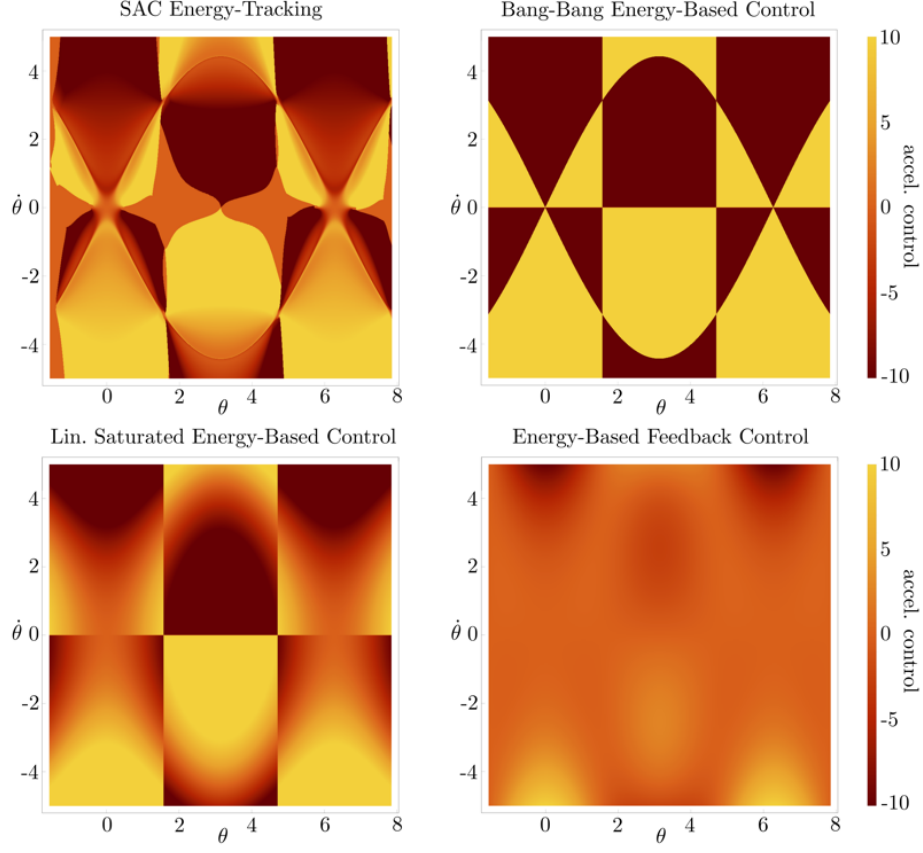


Figure 7: Comparison of feedback control laws for the cart-pendulum system. Top left: feedback from SAC with energy tracking. Top right: bang-bang feedback control defined by $u = u_{\max} \text{sign}(E - \bar{E}) \dot{\theta} \cos(\theta)$. Bottom left: saturated control law defined by $u = \max(-u_{\max}, \min(u_{\max}, k \cdot (E - \bar{E}) \text{sign}(\dot{\theta} \cos(\theta))))$ with $k = 0.5$. Bottom right: linear, energy-based feedback law $u = k \cdot (E - \bar{E}) \dot{\theta} \cos(\theta)$ with $k = 0.04$. The SAC control shares structural similarities with the analytically designed feedback laws and combines their advantages: bang-bang type control to change the energy as fast as possible, but also large regions of zero/low control as in the linear feedback map.

This expression is evaluated for the control from the SAC feedback map and its sign structure is depicted in Fig. 6 on the right.

Locally around the target state $(\theta, \dot{\theta}) = (0, 0)$ (and $(2\pi, 0)$, respectively), we see that $\frac{dV}{dt} \leq 0$ except for very few points that probably occur due to numerical inaccuracies in the computation of SAC controls. Thus, this proves local stability of SAC under the Lyapunov function chosen in [15]. As in [15], we cannot formally show asymptotic stability by means of this Lyapunov function, since $\frac{dV}{dt}$ is equal to zero for all $\dot{\theta} = 0$ and for $u = 0$. However, observing the vector field, we see that the cart-pendulum does not stay within the areas of zero control or on the $\dot{\theta} = 0$ axis. This corresponds to our computational tests, in which the pendulum stabilized to the inverted equilibrium for all globally chosen initial conditions. The closed-loop vector field obtained from our numerical results (Fig. 6) also indicate it is likely that we can prove global optimality using a better choice of Lyapunov function. We leave this for future work.

4.5. Concluding Discussion

Conclusions from studying the cart-pendulum example are twofold: Firstly, the comparisons to final state tracking and energy-tracking methods serves as a validation of the SAC manifold tracking approach. In particular, the SAC feedback map is sandwiched by the bang-bang and the linear feedback law from [15] (cf. Fig. 7 top right and bottom right plot). Thus the SAC map not only provides controls in a reasonable range, it also automatically determines state space regions where bang-bang control and linear control, respectively, are favorable. Moreover, these results show how SAC as an optimization-based control method can be used to combine classical types of energy-shaping feedback (linear feedback, bang-bang) in a beneficial way. Secondly, unlike classical energy-based control laws that are generated by a system specific analysis, SAC is an optimization-based numerical approach. The energy tracking cost used for the cart-pendulum can be replaced by a manifold tracking cost that uses numerical approximations of the stable manifold. Thus our approach also scales to higher dimensional systems as demonstrated for the pendubot in Section 5.

5. Swing-Up of the Pendubot

The pendubot is a two-link manipulator, see Fig. 8, 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, $u = \tau_1$.

385 This section uses the methods previously described to compute a switching control sequence that 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 [1, 25] and physical experiments in [26], with

$$\begin{aligned} m_1 &= 1.0367 \text{ kg} & m_2 &= 0.5549 \text{ kg} \\ l_1 &= 0.1508 \text{ m} & l_2 &= 0.2667 \text{ m} \\ 390 \quad l_{c1} &= 0.1206 \text{ m} & l_{c2} &= 0.1135 \text{ m} \\ I_1 &= 0.0031 \text{ kg m}^2 & I_2 &= 0.0035 \text{ kg m}^2 . \end{aligned}$$

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 [1], 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 [27] 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.

395

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, which are computed using GAIO as described in Section 2.1 (also see [5] for details). Figure 9 shows the box approximation of the stable manifold.

400

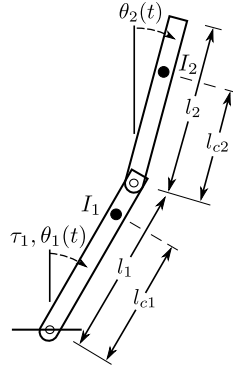


Figure 8: Model of the pendubot

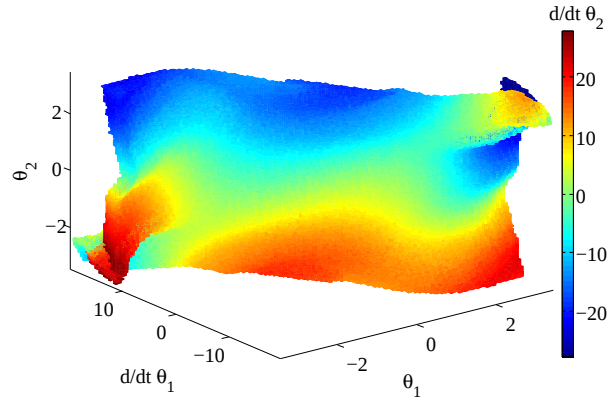


Figure 9: Box approximation of the stable manifold of the pendubot's inverted equilibrium. The box coloring indicates the value of the fourth coordinate, $\dot{\theta}_2$.

Over the region depicted, states on the stable manifold can be parametrized as a function of the two angular coordinates (θ_1, θ_2) , i.e.

$$S : \mathbb{R}^2 \mapsto \mathbb{R}^4, S(\theta_1, \theta_2) := (\theta_1, S_1(\theta_1, \theta_2), \theta_2, S_2(\theta_1, \theta_2)),$$

where $S_i : \mathbb{R}^2 \mapsto \mathbb{R}$, $i = 1, 2$, map from angular coordinates to manifold velocities. The cost,

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

tracks these stable manifold states. We apply $Q = \text{Diag}(\{0, 5, 0, 10\})$ and $P_1 = 0$. After testing 25 initial conditions with angles (θ_1, θ_2) sampled (linearly) 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. 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
425 that it reduces computation (no finite differences) and filtering requirements.
The following subsection shows that, in spite of these issues, SAC can out-
perform alternatives and successfully inverts the pendubot with only manifold
tracking goals.

5.2. Discussion of Numerical Results

430 For comparison with previous pendubot swing-up results from [1], which are
based on a SAC controller performing state tracking with $J_{\bar{x}}$, $Q = \text{Diag}(\{100,$
 $0.0001, 200, 0.0001\})$, 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
435 200 Hz feedback sampling rate.

Figure 10 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 [1]). Without any state error goal, SAC successfully inverts
the pendubot in roughly the same time of ≈ 4 s using the same peak torque
440 (matching physical experiments in [26] and half that from simulations in [25]).
In both cases (tracking with J_S and $J_{\bar{x}}$), SAC controllers use free dynamics and
apply control only when needed.

Lower peak torque swing-up solutions. As for the cart-pendulum, typical pen-
dubot swing-up control laws exhibit sections where no control is applied and
445 the system is allowed to drift (see Fig. 10). Unlike for the cart-pendulum, af-
ter 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 op-
posite). Results show manifold tracking requires much less peak control effort
450 for swing-up tasks (better $\mathcal{L}^\infty$ norm). This is of higher practical importance,
since peak torque requirements drive motor selection. 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 $P_1 = \text{Diag}(\{0, 15, 0, 10\})$ to J_S , yields inversion with $|u| < 4$ Nm, nearly *half that of the best case* state error tracking results.

Enhanced Robustness and Bandwidth of SAC. 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, $P_1 = \text{Diag}(\{0, 15, 0, 10\})$ in J_S , with all the same parameters and constraints defined earlier. Again, with a state error cost, 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. 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.⁴ The previously described initial condition tests 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.

⁴At 200 Hz, SAC calculations typically range from < 1 s 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.

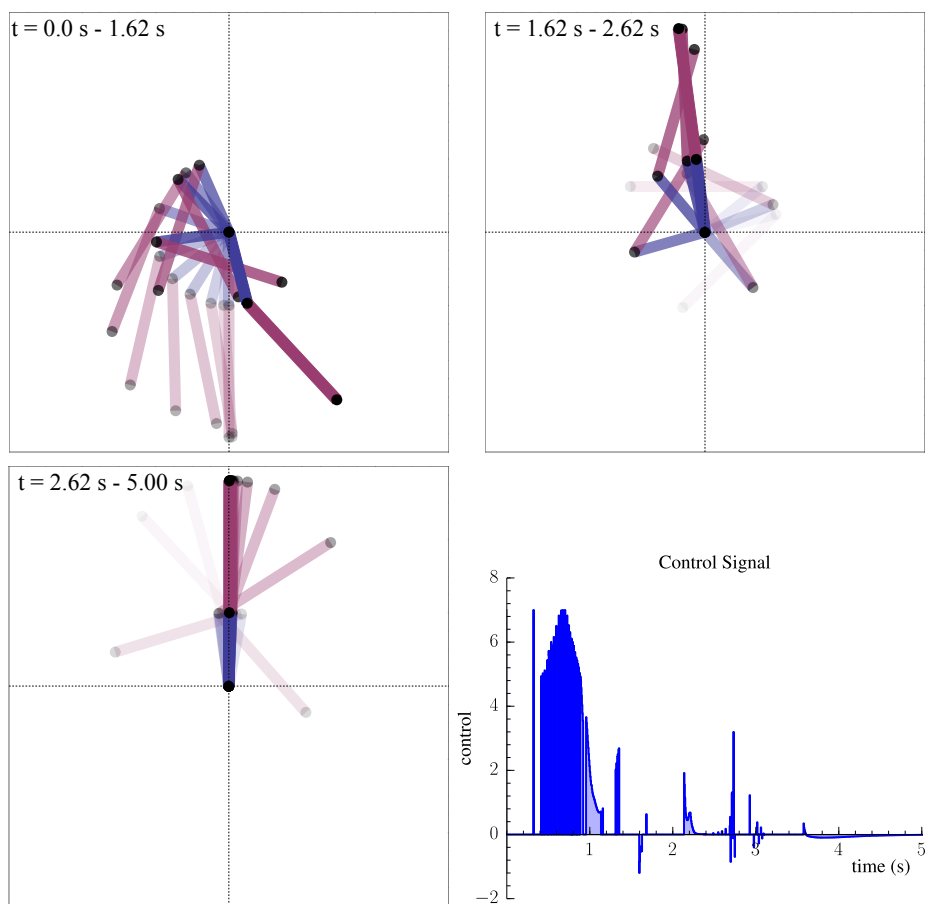


Figure 10: Manifold-tracking swing-up solution.

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 allow shorter horizons in SAC receding horizon calculations. Hence, our approach facilitates higher frequency feedback and control. For the cart-pendulum example, our feedback map automatically combines structures of existing energy-based control strategies and it is computationally shown to be stabilizing. As opposed to existing energy-based strategies, the proposed controllers use hybrid optimization to automate synthesis and do not rely on pre-derived analytical strategies.

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 the representation of manifold data. The feedback map for the cart-pendulum can be used to define automata for symbolic control, as it has already been presented in [28] and will be investigated further in future work.

Finally, in terms of stability, as mentioned in Section 5, 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.

500 *Acknowledgements*

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.

505 **References**

- [1] A. Ansari, T. Murphey, Sequential Action Control: Closed-form optimal control for nonlinear systems, submitted to IEEE Transactions on Robotics. <http://nxr.northwestern.edu/publications> (2015).
- [2] A. Ansari, K. Flaßkamp, T. D. Murphey, Sequential action control for tracking of free invariant manifolds, in: 5th IFAC Conference on Analysis and Design of Hybrid Systems, 2015, pp. 335–342.
- [3] J. E. Marsden, S. D. Ross, New methods in celestial mechanics and mission design, *Bulletin of the American Mathematical Society* 43 (2006) 43–73.
- [4] K. Flaßkamp, S. Ober-Blöbaum, M. Kobilarov, Solving optimal control problems by exploiting inherent dynamical systems structures, *Journal of Nonlinear Science* 22 (4) (2012) 599–629.
- [5] K. Flaßkamp, J. Timmermann, S. Ober-Blöbaum, A. Trächtler, Control strategies on stable manifolds for energy-efficient swing-ups of double pendula, *International Journal of Control* 87 (9) (2014) 1886–1905.
- [6] F. Allgöwer, R. Findeisen, Z. K. Nagy, Nonlinear model predictive control: From theory to application, *J. Chin. Inst. Chem. Engrs* 35 (3) (2004) 299–315.
- [7] F. Allgöwer, A. Zheng (Eds.), *Nonlinear Model Predictive Control*, Progress in Systems and Control Theory, Springer, 2000.
- [8] E. Camacho, C. Alba, *Model Predictive Control*, Advanced Textbooks in Control and Signal Processing, Springer London, 2013.

- [9] T. M. Caldwell, T. D. Murphey, Projection-based optimal mode scheduling, in: IEEE Conference on Decision and Control, 2013.
- [10] H. Gonzalez, R. Vasudevan, M. Kamgarpour, S. S. Sastry, R. Bajcsy, C. J. Tomlin, A descent algorithm for the optimal control of constrained nonlinear switched dynamical systems, in: ACM Conference on Hybrid Systems: Computation and Control, 2010, pp. 51–60.
- [11] M. Egerstedt, Y. Wardi, H. Axelsson, Transition-time optimization for switched-mode dynamical systems, IEEE Transactions on Automatic Control 51 (1) (2006) 110–115.
- [12] Y. Wardi, M. Egerstedt, Algorithm for optimal mode scheduling in switched systems, in: American Control Conference, 2012, pp. 4546–4551.
- [13] I. Fantoni, R. Lozano, M. W. Spong, Energy based control of the pendubot, IEEE Transactions on Automatic Control 45 (2000) 725–729.
- [14] X. Xin, T. Yamasaki, Energy-based swing-up control for a remotely driven acrobot: Theoretical and experimental results., IEEE Trans. Contr. Sys. Techn. 20 (4) (2012) 1048–1056.
- [15] K. Åström, K. Furuta, Swinging up a pendulum by energy control, Automatica 36 (2) (2000) 287–295.
- [16] M. W. Spong, The swing up control problem for the acrobot, in: IEEE Control Systems Magazine, 15, 1995, pp. 49–55.
- [17] W. Zhong, H. Röck, 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, 2001.
- [18] A. Shiriaev, A. Pogromsky, H. Ludvigsen, O. Egeland, On global properties of passivity-based control of an inverted pendulum, International Journal of Robust and Nonlinear Control 10 (4) (2000) 283–300.

- [19] C. C. Chung, J. Hauser, Nonlinear control of a swinging pendulum, *Automatica* 31 (6) (1995) 851–862.
- 555 [20] M. Dellnitz, O. Junge, Set oriented numerical methods for dynamical systems, in: B. Fiedler (Ed.), *Handbook of Dynamical Systems*, Elsevier Science, 2002, Ch. 5, pp. 221 – 264.
- [21] M. Dellnitz, G. Froyland, O. Junge, The algorithms behind GAIO – set oriented numerical methods for dynamical systems, in: B. Fiedler (Ed.),
560 *Ergodic Theory, Analysis, and Efficient Simulation of Dynamical Systems*, Springer, 2001, pp. 145–174.
- [22] J. Guckenheimer, P. Holmes, *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, Vol. 42 of *Applied Mathematical Sciences*, Springer, 1983.
- 565 [23] B. Krauskopf, H. M. Osinga, E. J. Doedel, M. E. Henderson, J. Guckenheimer, A. Vladimirovsky, M. Dellnitz, O. Junge, 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) (2005) 763–791.
- 570 [24] K. J. Åström, J. Aracil, F. Gordillo, 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, 2005.
- [25] T. Albahkali, R. Mukherjee, T. Das, Swing-up control of the pendubot: an impulse–momentum approach, *IEEE Transactions on Robotics* 25 (4)
575 (2009) 975–982.
- [26] Y. Orlov, L. T. Aguilar, L. Acho, A. Ortiz, Swing up and balancing control of pendubot via model orbit stabilization: Algorithm synthesis and experimental verification, in: *45th IEEE Conference on Decision and Control*, 2006, pp. 6138–6143.

- 580 [27] P. A. Parrilo, Structured semidefinite programs and semialgebraic geometry
methods in robustness and optimization, Ph.D. thesis, California Institute
of Technology (2005).
- [28] A. Mavrommati, T. D. Murphey, Automatic synthesis of control alphabet
policies, accepted to IEEE International Conference on Automation Science
585 and Engineering (CASE) 2016.