A Hybrid Algorithm for Practical Nonconvex Optimization

Dawn M. Hustig-Schultz [*] **Ricardo G. Sanfelice** [*]

[*] *Department of Electrical and Computer Engineering, University of California, 1156 High Street, Santa Cruz, CA 95064, USA. (e-mail: dhustigs, ricardo@ucsc.edu).*

Abstract: This paper proposes a hybrid algorithm for optimization, to ensure convergence to a local minimimzer of a nonconvex Morse objective function L with a single, scalar argument. Developed using hybrid system tools, and based on the heavy ball method, the algorithm features switching strategies to detect whether the state is near a critical point and enable escape from local maximizer, using measurements of the gradient of L. Key properties of the resulting closed-loop system, including existence of solutions and practical global attractivity, are revealed. Numerical results validate the findings.

Keywords: Hybrid systems, Optimization: theory and algorithms, Stability

1. INTRODUCTION

In this paper, we consider the problem of finding a local minimizer of a scalar, continuously differentiable objective function L with a single, scalar argument, which is not necessarily convex, and may have multiple local minimizers. In particular, we are interested in an algorithm capable of solving optimization problems of the form

$$\min_{\xi \in \mathbb{R}} L(\xi), \tag{1}$$

with a guarantee of *global* attractivity of the set of minimizers. By *global*, we mean "from any initial condition (or guess)." This is different from the typical use of the term global in the optimization literature, which corresponds to the guarantee that an optimization algorithm converges to the global minimizer rather than to a local minimizer. In fact, the objective functions considered in this paper may have multiple isolated critical points, which are known to impose challenges to optimization algorithms.

For the type of nonconvex optimization problem in which we are interested, and approaching the problem from a control theory viewpoint, it is infeasible to design an algorithm of the form

$$\dot{\xi} = f(\xi, \nabla L(\xi)), \tag{2}$$

that solves the problem with attractivity and robustness when small measurement noise exists in measurements of the gradient. This infeasibility suggests the need of an algorithm that is robust to measurement noise. Such an algorithm would detect when the state ξ is close to a local maximum, and then implement a strategy that moves the state away from that maximum. Instead of an algorithm of the form $\dot{\xi} = f(\xi, \nabla L(\xi))$, we propose an algorithm conveniently modeled and designed using hybrid system tools, based on the heavy ball method, for convergence to a local minimum of a nonconvex Morse objective function L. The *heavy ball* method is an accelerated gradient method capable of guaranteeing global convergence to the set of minimizers of L when L is convex Polyak (1964), Polyak (1987). Unlike classical gradient descent, the heavy ball method adds an inertial (or "velocity") term to the gradient to speed up convergence.

To the best of our knowledge, we propose the first algorithm based on the heavy ball method for which the set of minimizers of a nonconvex objective function L, with a single, scalar argument, is practically globally attractive, and for which we observe robustness to small noise in simulation. In contrast, the previous literature establishes only the convergence rate for the heavy ball method. In particular, the heavy ball method was first analyzed in a nonconvex setting in Zavriev and Kostyuk (1993). In Attouch et al. (2000), the convergence bounds for the heavy ball method, when L is a Morse function, are derived.

There has been a surge of interest in utilizing hybrid systems tools for gradient-based optimization. In Strizic et al. (2017), the authors propose a hybrid gradient descent algorithm using an adjustable diffeomorphism to ensure global asymptotic stability to the minimum of a compact manifold that is a circle. This algorithm is then extended to manifolds with an equal number of maxima and minima, and then propose a model-free version of the algorithm. The authors in Baradaran et al. (2018) present a class of hybrid stochastic gradient descent algorithms to solve nonconvex optimization problems on smooth manifolds. They prove uniform global asymptotic stability in probability and then extend the algorithm to a partially multiagent setting. In Kolarijani et al. (2018) and Kolarijani et al. (2019), the authors present two hybrid algorithms based on Nesterov' s accelerated gradient descent: one with a state-dependent, time-invariant damping input and another with an input that controls the magnitude of the gradient term. The algorithms require the objective function to satisfy the Polyak-Lojasiewicz inequality, which includes a subclass of nonconvex functions in which all stationary points are global minimizers. Although the authors in Kolarijani et al. (2019) prove an exponential convergence rate for these two algorithms, they do not analyze the global asymptotic attractivity property of the set of minimizers.

The main contributions of this paper are as follows. We develop an optimization algorithm, based on the heavy

[1] Research partially supported by NSF Grants no. ECS-1710621, CNS-1544396, and CNS-2039054, by AFOSR Grants no. FA9550-19-1-0053, FA9550-19-1-0169, and FA9550-20-1-0238, and by CITRIS and the Banatao Institute at the University of California.

ball method, for convergence to a local minimum of a nonconvex Morse objective function L with a single, scalar argument. We emphasize that our proposed algorithm is not designed to find all the local minimizers, but rather to converge to an element in the set of local minimizers. The algorithm employs a switching strategy, developed using hybrid system tools Goebel et al. (2012), to detect whether the state ξ is near a critical point and ensure escape from local maxima, depicted in Figure 1. Such a switching strategy employs measurements of the gradient of L – which in practice are typically approximated from measurements of L – and hysteresis to determine whether the state ξ needs to be pushed away from a nearby critical point, or whether the state ξ is far enough away from a critical point to resume use of the heavy ball method. The algorithm does not need to distinguish between local maximizers and local minimizers, and therefore does not need information about the Hessian. We prove practical global attractivity of the set of minimizers of L for the closed-loop system and, preliminarily, we observe that the algorithm is robust to arbitrarily small noise in measurements of the gradient, as illustrated in Figure 1.

The rest of the paper is organized as follows. Section 2 contains a brief explanation of notation and the hybrid systems framework employed. Section 3 outlines challenges to nonconvex optimization. Section 4 presents the problem statement and assumptions. Section 5 introduces the algorithm and presents its nominal properties. Due to space constraints, detailed proofs of results will be published elsewhere.

2. PRELIMINARIES

2.1 Notation

We denote the real, positive real, and natural numbers as $\mathbb{R}$, $\mathbb{R}_{>0}$, and $\mathbb{N}$, respectively. An n times continuously differentiable function is notated as C^n. By $\mathbb{B}$ we denote the open unit ball in $\mathbb{R}^n$ centered at the origin. For vectors v and w, $|v| = \sqrt{v^\top v}$ defines the Euclidean vector norm of v, and $\langle v, w\rangle = v^\top w$ defines the inner product of v and w. The closure of a set S is denoted as $\overline{S}$. The distance from a point x to a nonempty set S is defined by $|x|_S = \inf_{y\in S} |y - x|$. Given a set-valued mapping, denoted as $M : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$, the domain of M is the set dom $M = \{x \in \mathbb{R}^n : M(x) \neq \emptyset\}$. A continuous function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a class-$\mathcal{K}$ function if it is strictly increasing and it is such that $\alpha(0) = 0$.

2.2 Preliminaries on Hybrid Systems

In this paper, a hybrid system $\mathcal{H}$ has data (C, F, D, G) and is defined as Goebel et al. (2012)

$$\mathcal{H} = \begin{cases} \dot{x} \ \in F(x) & x \in C \\ x^+ \in G(x) & x \in D \end{cases} \tag{3}$$

where $x \in \mathbb{R}^n$ is the system state, $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is the flow map, $C \subset \mathbb{R}^n$ is the flow set, $G : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is the jump map, and $D \subset \mathbb{R}^n$ is the jump set. A solution ϕ is parameterized by $(t, j) \in \mathbb{R}_{\geq 0} \times \mathbb{N}$, where t is the amount of continuous time that has passed and j is the number of jumps that have occurred. The domain of ϕ, namely, dom$\phi \subset \mathbb{R}_{\geq 0} \times \mathbb{N}$, is a hybrid time domain, which is a set such that for each $(T, J) \in$ domϕ, dom$\phi \cap ([0, T] \times \{0, 1, \ldots, J\}) = \cup_{j=0}^{J}([t_j, t_{j+1}], j)$ for a finite sequence of times $0 = t_0 \leq t_1 \leq t_2 \leq \ldots \leq t_{J+1}$. A solution ϕ to $\mathcal{H}$ is called maximal if it cannot be extended further. A solution is called complete if its domain is unbounded. In the upcoming results, we will assume that our proposed hybrid closed-loop algorithm meets the hybrid basic conditions, as defined in (Goebel et al., 2012, Assumption 6.5).

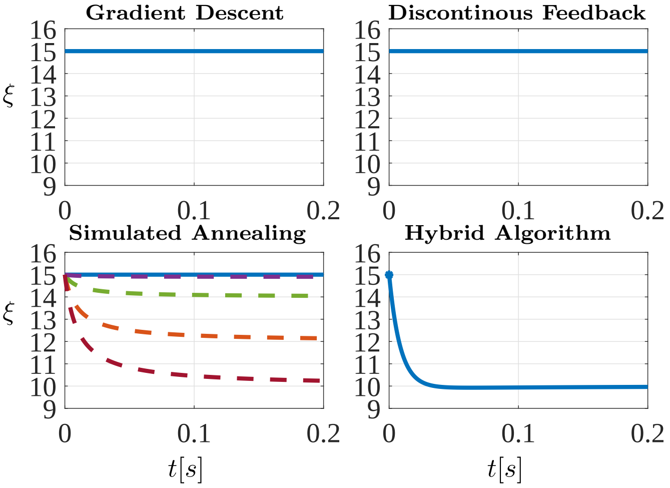

Fig. 1. Comparing performance of the proposed hybrid algorithm to other optimization methods, with small noise in measurements of the gradient, when the system starts near a local maximum, at $\xi_0 \approx 15$. For classic gradient descent (top left), a gradient-based optimization algorithm with discontinuous map f (top right), and simulated annealing, via Langevin diffusion (bottom left, solid line) the state ξ gets pushed to the local maximum at $\xi = 15$, and stays there. All trajectories in the bottom left plot have the noise signal $m := \left(-\frac{\tau(\log(\tau))^2}{c_{SA}}\right)\left(\nabla L(\xi) + \mu\text{sign}(\nabla L(\xi))(10^{-12})\right)$, where $\tau > 0$ is time, $c_{SA} > 0$ is large, and μ is a normally distributed random number. The signal m is a standard part of the algorithm. The trajectory with the solid line also has an added constant of $c_m = 5 \times 10^{-13}$, such that $m + c_m$, while the other three trajectories represented by dashed lines have added constants c_m equal to 3×10^{-13}, 10^{-13}, and 10^{-14}, respectively. The last trajectory, represented by the dashed line converging to the minimizer, has no constant added to m. The proposed hybrid algorithm (bottom right), with noise of the form $\left(-\frac{\tau(\log(\tau))^2}{c_{SA}}\right)\left(\nabla L(\xi) + \mu\text{sign}(\nabla L(\xi))(10^{-12})\right)$ added to the gradient of L, where μ is a normally distributed random number, is still able to escape the local maximum at $\xi = 15$ and converge to a local minimum at $\xi = 10$.

2.3 Preliminaries on Morse Functions

We will restrict the objective function L to the class of Morse functions Audin and Damian (2014).

Definition 2.1. (Morse function) The function $L : \mathbb{R}^n \to \mathbb{R}$ is a Morse function if none of its critical points is degenerate.

For functions $L : \mathbb{R}^n \to \mathbb{R}$, a critical point is degenerate if its Hessian is singular. The Morse Lemma describes the behavior near a critical point of a Morse function (Audin and Damian, 2014, Theorem 1.3.1). The Morse Lemma shows how a real-valued function $L : \mathbb{R}^n \to \mathbb{R}$ behaves on a manifold near a nondegenerate critical point, facilitating classification of an area around that critical point according to the index of L. For instance, the indices of minima, saddle points, and maxima are 0, 1, and 2, respectively. An immediate corollary of the Morse Lemma (Audin and Damian, 2014, Corollary 1.3.2) is as follows.

Corollary 2.2. The nondegenerate critical points of a Morse function are isolated.

The critical points of a Morse function are isolated, which means that critical points are single points, i.e., a Morse function cannot have a continuum of critical points. Note that although Definition 2.1 and the Morse Lemma refer more generally to manifolds, we will restrict our analysis to Morse functions on the one-dimensional manifold $\mathbb{R}$, namely, we consider Morse functions with a single, scalar argument. For $\mathcal{C}^2$ functions with a single argument in $\mathbb{R}$, a saddle point is a stationary point that is also an inflection point. For such inflection points, the determinant of the Hessian is always singular (Larson et al., 2007, Theorem 4.8), and therefore degenerate. Therefore, saddle points never occur in $\mathcal{C}^2$ for Morse functions on the one-dimensional manifold $\mathbb{R}$. See Section 7 for more details on possible extensions to higher dimensions, where saddle points can occur, i.e., for $L : \mathbb{R}^n \to \mathbb{R}$ where $n > 1$.

3. CHALLENGES IN NONCONVEX OPTIMIZATION

As mentioned in Section 1, it is infeasible to design an algorithm of the form (2) that solves nonconvex optimization problems of the form (1) with attractivity and robustness. To illustrate this point, consider the function L given by $L(\xi) = \frac{\xi^2(\xi-10)^2(\xi-20)^2(\xi-30)^2}{10{,}000}$ for each $\xi \in \mathbb{R}$, for which each $\xi \in \{0, 10, 20, 30\}$ is a local minimizer and each $\xi \in \{5(3-\sqrt{5}), 15, 5(3+\sqrt{5})\}$ is a local maximizer. Classic gradient descent, which corresponds to $f(\xi, \nabla L(\xi)) = -\nabla L(\xi)$, does not render the set of minimizers of this function globally attractive, since when the state ξ starts at a local maximizer, we have that ∇L is zero and the algorithm remains stuck at such a local maximizer. Moreover, when the state ξ starts close to the local maximizer and there is small noise added to the measurements of the gradient, then the algorithm cannot always push ξ away from the maximizer, even when the noise signal is arbitrarily small. This can be seen in the top left plot of Figure 1, where arbitrarily small noise in the gradient keeps the state close to the local maximizer of L at $\xi = 15$.[2]

Algorithms of the form (2) with a static, discontinuous map f, for which the nominal system has the set of minimizers of L globally asymptotically stable, are not robust to arbitrarily small measurement noise. Such a system is not well-posed[3], due to discontinuities in the map f, at local maximizers. In fact, when the state ξ starts close to one of the points of discontinuity, and when small noise is added to the measurements of the gradient, there will exist a solution that remains nearby such a point, even when the noise is arbitrarily small. The limit of such a solution as the noise goes to zero is a solution to the differential inclusion $\dot{\xi} \in F(\xi, \nabla L(\xi))$, where F is the *Krasovskii regularization* of $\xi \mapsto f(\xi, \nabla L(\xi))$. Such a solution, when the right-hand side is bounded, is also a Hermes solution (Goebel et al., 2012, Theorem 4.3), and represents an equilibrium point of $\dot{\xi} \in F(\xi, \nabla L(\xi))$, from which the state ξ cannot converge to a local minimizer. Therefore, the Krasovskii regularization does not have the set of minimizers of L globally attractive. According to Goebel et al. (2012), the attractivity of the original system $\dot{\xi} = f(\xi, \nabla L(\xi))$ with f discontinuous is not robust. This behavior can be seen in the top right plot of Figure 1, where arbitrarily small noise induces an equilibrium point at the maximizer located at $\xi = 15$, at which $f(\xi, \nabla L(\xi))$ is discontinuous.

Simulated annealing Chiang et al. (1987), via Langevin diffusion, is a popular alternative used to find the global minimizer of a nonconvex function. Langevin diffusion, which corresponds to $\dot{\xi} = -\nabla L(\xi) + c(t)m(t)$ combines classic gradient descent with a noise signal m, such as Brownian motion, for which the magnitude is controlled by the "temperature" function c. Although such a noise signal is used to help the state find the global minimum, it can also be detrimental to performance. It can be shown that when the state ξ starts close to a local maximizer the algorithm cannot always push ξ away from the maximizer, due to this noise signal, no matter how large the initial annealing temperature is. This is even the case when the noise is arbitrarily small. This behavior is shown by the solid line in the bottom left plot of Figure 1, where noise keeps the state ξ close to the local maximizer at $\xi \approx 15$. The dashed lines show the effect of other small noise, which causes the state ξ to drift away from the local maximum, and eventually converge to a local minimum. Essentially, as the size of the noise increases, even if still small, the more likely simulated annealing is to be stuck at a local maximizer.

The issues depicted in the top left, top right, and bottom left of Figure 1 show that nonconvex optimization problems cannot be efficiently solved with existing line search algorithms or stochastic algorithms. On the contrary, Figure 1 demonstrates the need of an algorithm, modeled and designed using hybrid system tools, that in simulation demonstrates robustness to measurement noise. Its performance is shown in the bottom right of Figure 1, starting at $\xi \approx 15$ with zero velocity, and converging despite the presence of noise in measurements of the gradient, as is present for the other algorithms in Figure 1.

4. PROBLEM STATEMENT AND ASSUMPTIONS

4.1 Problem Statement

The problem addressed in this paper is as follows.

Problem 1. Given a continuously differentiable Morse objective function $L : \mathbb{R} \to \mathbb{R}$, which may have multiple isolated minimizers and maximizers, design an optimization algorithm that guarantees practical convergence to a local minimizer from all initial conditions – including local maximizers – using measurements of ∇L.

We emphasize that, to solve Problem 1, the algorithm has no knowledge of the particular objective function L or of its critical points.

4.2 Assumptions and Definitions

The set of all local minimizers of L is denoted as

$$\mathcal{A}_{1_{\min}} = \left\{ z_1 \in \mathbb{R} : \nabla L(z_1) = 0, \nabla^2 L(z_1) > 0 \right\}. \quad (4)$$

Conversely, the set of all local maximizers of L is denoted as

$$\mathcal{A}_{1_{\max}} = \left\{ z_1 \in \mathbb{R} : \nabla L(z_1) = 0, \nabla^2 L(z_1) < 0 \right\}. \quad (5)$$

[2] Code at github.com/HybridSystemsLab/RobustnessHeavyBall

[3] For a purely continuous-time algorithm, well-posed means that solutions depend "continuously" with respect to initial conditions.

Then, the set of all critical points of $L : \mathbb{R} \to \mathbb{R}$ is given as

$$\mathcal{A}_1 = \mathcal{A}_{1_{\min}} \cup \mathcal{A}_{1_{\max}}. \tag{6}$$

The following assumptions are required by some of the forthcoming results.

Assumption 4.1. (Properties of the objective function L)

(M1) L is a Morse function;
(M2) L is $\mathcal{C}^2$;
(M3) There exists $d_0 > 0$ such that each $z^* = (z_1^*, 0) \in \mathcal{A}_1 \times \{0\}$ satisfies $(z^* + d_0\mathbb{B}) \cap ((\mathcal{A}_1 \times \{0\}) \setminus \{z^*\}) = \emptyset$;
(M4) L is radially unbounded;
(M5) There exists $\alpha \in \mathcal{K}$ such that for each $\varepsilon > 0$ sufficiently small, there exists $\delta \in (0, \alpha(\varepsilon))$ such that if $|\nabla L(z_1)| \leq \varepsilon$ then $|z_1|_{\mathcal{A}_1} \leq \delta$.

Remark 4.2. The finite separation $d_0 > 0$ between critical points from (M3) ensures that critical points do not accumulate, which is required for our algorithm to solve Problem 1. A similar finite separation assumption can be found in Jin et al. (2017). Additionally, we do not expect that a solution to Problem 1 exists without (M3). Item (M4) ensures radial unboundedness of the Lyapunov function used in the attractivity analysis of the proposed algorithm. Item (M5) of Assumption 4.1 means that z_1 is suboptimal Boyd and Vandenberghe (2004). Item (M5) is used to ensure that the algorithm can detect when the state z is near a critical point, using only measurements of ∇L.

5. A HYBRID ALGORITHM FOR NONCONVEX OPTIMIZATION

In this section, we present a logic-based algorithm for Morse functions that uses the heavy ball algorithm when the state z is far from a critical point and that uses linear feedback when the state z is near a critical point, to push z away from such a critical point.

Our proposed algorithm has a state $z := (z_1, z_2) \in \mathbb{R}^2$, where z_1 represents the argument of L and z_2 represents the "velocity" variable. The state z remains unchanged at jumps, but updates during flows according to

$$\dot{z}_1 = z_2, \ \dot{z}_2 = u \tag{7}$$

where u takes different forms depending on whether the state z is close to or far from a critical point. Our algorithm uses a logic variable, $q \in Q := \{0, 1\}$, to indicate when to push the state z_1 away from a critical point. The logic value $q = 0$ leads to the algorithm using the heavy ball method to converge to the neighborhood of a critical point, and $q = 1$ leads to the algorithm using linear feedback to push z_1 away from a critical point. In addition, our algorithm has a state ℓ to determine the magnitude and direction to push the state z_1 when close to a critical point. To trigger jumps, hysteresis parameters $0 < \varepsilon_1 < \varepsilon_2$ and $0 < \rho_1 < \rho_2$ are used. These parameters are small enough to ensure convergence to a neighborhood of a local minimum without overshooting to a neighboring maximum. The algorithm uses a parameter $\nu > 0$, to tune the speed of convergence.

A high-level description of the proposed algorithm is as follows. When the state z is near a critical point with small velocity, as determined by $|\nabla L(z_1)| \leq \varepsilon_1$ and $|z_2| \leq \rho_1$, the algorithm resets the logic variable q to 1 and assigns u to ℓ. Then, z moves away from the critical point according to $u = \ell$, where $\ell := \nu \text{sign}(z_2)$. The feedback $\nu\text{sign}(z_2)$ causes the state z_2 to change linearly and z_1 to change quadratically, thus eventually pushing the state z away from a critical point. When the state z is far away from the critical point and the velocity is larger, as determined by $|\nabla L(z_1)| \geq \varepsilon_2$ and $|z_2| \geq \rho_2$, the algorithm resets the logic variable q to 0 and assigns u to

$$\kappa(h(z)) := -\lambda z_2 - \gamma \nabla L(z_1), \tag{8}$$

which is defined for all $z \in \mathbb{R}^2$, where $\lambda > 0$ represents friction, $\gamma > 0$ represents gravity, and h is given by

$$h(z) := \begin{bmatrix} z_2 \\ \nabla L(z_1) \end{bmatrix}. \tag{9}$$

The function h characterizes the measurements used by the algorithm. With the proposed logic, the state z converges a nearby local minimizer, contained in D_0, with zero velocity via $u = \kappa(h(z))$. From such a point, although the state z is pushed to D_1, the state z will again converge to nearby the same local minimizer as before, via $u = \kappa(h(z))$, and this process repeats for all time. The positive parameters $(\varepsilon_1, \varepsilon_2, \rho_1, \rho_2)$ need to be properly tuned to keep z in a small neighborhood of a local minimizer. The complete algorithm is summarized in Algorithm 1.

Algorithm 1 Hybrid Algorithm

```
Set $q(0,0)$ to 0, and set $z(0,0)$ and $\ell(0,0)$ as initial conditions with arbitrary values.
while true do
    if $|\nabla L(z_1)| \leq \varepsilon_1$ and $|z_2| \leq \rho_1$ and $q = 0$ then
        Update $q$ to 1;
        Update $\ell$ to $\nu\text{sign}(z_2)$ and assign $u$ to $\ell$;
    else if $|\nabla L(z_1)| \geq \varepsilon_2$ and $|z_2| \geq \rho_2$ and $q = 1$ then
        Update $q$ to 0;
        Assign $u$ to $\kappa(h(z))$, defined via (8).
    else
        Allow flows of (7) with $u = \kappa(h(z))$ if $q = 0$ and with $u = \ell$ if $q = 1$.
    end if
end while
```

The rest of this section is organized as follows. Section 5.1 introduces the hybrid system model for the proposed algorithm. Finally, Section 5.2 contains the main results, which reveal the nominal properties of the proposed algorithm.

5.1 Hybrid System Model of the Proposed Algorithm

The proposed algorithm is modeled as a hybrid system $\mathcal{H}$ with parameter $\nu > 0$, state $x := (z, q, \ell) \in \mathbb{R}^2 \times Q \times \{-\nu, \nu\}$, and data (C, F, D, G) defined as follows:

$$F(x) := \begin{bmatrix} z_2 \\ \widetilde{\kappa}(x) \\ 0 \\ 0 \end{bmatrix} \quad \forall x \in C := \overline{(\mathbb{R}^2 \times Q \times \{-\nu, \nu\}) \setminus D} \tag{10a}$$

$$G(x) := \begin{bmatrix} z_1 \\ z_2 \\ 1 - q \\ \nu\text{sign}(z_2) \end{bmatrix} \quad \forall x \in D := D_0 \cup D_1 \tag{10b}$$

where $\nu > 0$ is properly tuned, $\text{sign}(z_2)$ is defined as the set-valued map

$$\text{sign}(z_2) = \begin{cases} 1 & \text{if } z_2 > 0 \\ \{-1, 1\} & \text{if } z_2 = 0 \\ -1 & \text{if } z_2 < 0 \end{cases} \tag{11}$$

and $\widetilde{\kappa}$ is defined as

$$\widetilde{\kappa}(x) = \begin{cases} \kappa(h(z)) & \text{if } q = 0 \\ \ell & \text{if } q = 1 \end{cases} \tag{12}$$

where $\kappa(h(z))$ is defined via (8). The sets D_0, and D_1 are defined below. As was outlined above and in Algorithm

1, the algorithm jumps when the state z is near a critical point with small velocity, as determined by $|\nabla L(z_1)| \leq \varepsilon_1$ and $|z_2| \leq \rho_1$, when $q = 0$. The algorithm also jumps when the state z is far from a critical point with larger velocity, as determined by $|\nabla L(z_1)| \geq \varepsilon_2$ and $|z_2| \geq \rho_2$, when $q = 1$, and when $\ell \in \{-\nu, \nu\}$. To this end, the sets D_0 and D_1 are defined as

$$D_0 := \left\{z \in \mathbb{R}^2 : |\nabla L(z_1)| \leq \varepsilon_1, |z_2| \leq \rho_1\right\} \times \{0\} \times \{-\nu, \nu\} \tag{13a}$$

$$D_1 := \left\{z \in \mathbb{R}^2 : |\nabla L(z_1)| \geq \varepsilon_2, |z_2| \geq \rho_2\right\} \times \{1\} \times \{-\nu, \nu\} \tag{13b}$$

where $\varepsilon_2 > \varepsilon_1 > 0$ and $\rho_2 > \rho_1 > 0$ are the inner and outer hysteresis bounds, used to determine whether the system is near a critical point – and needs to be pushed away from such a point using the feedback ℓ – or far enough away from a critical point to use the feedback $\kappa(h(z))$.

Remark 5.1. Our approach to tuning ε_2 and ε_1 uses the minimum separation $d_0 > 0$ between critical points, from item (M3) of Assumption 4.1. If for a given z_1, $|\nabla L(z_1)| \geq \varepsilon_2$, then the following relation can be derived: $0 < \varepsilon_1 < \varepsilon_2 < \min\left\{\nabla L(z_1) : z_1 \in \left\{z_1' \in \mathbb{R} : \nabla^2 L(z_1') = 0\right\}\right\}$. Since $\nabla^2 L(z_1') = 0$ occurs midway between critical points[4] , then such a tuning ensures that when the state z is near a local maximizer, it converges to the nearest local minimizer without overshooting to the next local maximizer. Such a tuning also ensures that if the state z is near a local minimizer, it stays near that same local minimizer. The function L, however, is not always known, and the hybrid closed-loop system in (10) assumes no knowledge of L. In practice, choosing $0 < \varepsilon_1 < \varepsilon_2$ small enough is sufficient.

5.2 Main Result

In this section, we show that the hybrid closed-loop system $\mathcal{H}$ with data (C, F, D, G) defined in (10) has the set

$$\mathcal{A} := \mathcal{A}_{1_{\min}} \times \{0\} \times Q \times \{-\nu, \nu\} \tag{14}$$

practically globally attractive in the positive parameters $(\varepsilon_1, \varepsilon_2, \rho_1, \rho_2)$, with basin of attraction that has a z_1 component equal to $\mathbb{R}$. Practical global attractivity of $\mathcal{A}$ means that, for each $\zeta > 0$ and for every solution x to $\mathcal{H}$, there exists $(t', j') \in \operatorname{dom} x$ such that $|x(t, j)|_{\mathcal{A}} \leq \zeta$ for all $(t, j) \in \operatorname{dom} x$ such that it is satisfying $t + j \geq t' + j'$.

Under item (M2) of Assumption 4.1, the hybrid closed-loop system $\mathcal{H}$, described in (10), is well-posed, as it meets the hybrid basic conditions.

When Assumption 4.1 holds, every maximal solution to the hybrid closed-loop system $\mathcal{H}$ is complete and bounded, as stated in the following lemma.

Lemma 5.2. (Existence of solutions for $\mathcal{H}$) Let L satisfy items (M2), (M3), and (M4) of Assumption 4.1. Then, every maximal solution to the closed-loop system $\mathcal{H}$ (10) *is bounded and complete.*

The following result shows that the hybrid closed-loop system $\mathcal{H}$ has the set $\mathcal{A}$ in (14) practically globally attractive.

Theorem 5.3. (Practical global attractivity of $\mathcal{A}$) Let L satisfy Assumption 4.1. Consider the hybrid closed-loop system $\mathcal{H}$ with data (C, F, D, G) defined in (10) *and the positive parameters $(\varepsilon_1, \varepsilon_2, \rho_1, \rho_2)$. Then, the set $\mathcal{A}$ in* (14) *is practically globally attractive for $\mathcal{H}$ in the sufficiently small parameters $(\varepsilon_1, \varepsilon_2, \rho_1, \rho_2)$; that is, for each $\zeta > 0$ sufficiently small, there exist parameters $(\varepsilon_1, \varepsilon_2, \rho_1, \rho_2)$ with $\varepsilon_1 \in (0, \varepsilon_2)$ and $\rho_1 \in (0, \rho_2)$ such that, for every solution x to $\mathcal{H}$, there exists $(t', j') \in \operatorname{dom} x$ such that*

$$|x(t, j)|_{\mathcal{A}} \leq \zeta \quad \forall (t, j) \in \operatorname{dom} x : t + j \geq t' + j' \tag{15}$$

Remark 5.4. The size of $\zeta > 0$ needs to be sufficiently small to keep the state z_1 in a small neighborhood of a local minimizer. To give some insight into its size, letting $\delta \in (0, \max\{\frac{d_0}{2}, \varepsilon\})$, where $d_0 > 0$ and where $\varepsilon > 0$ is sufficiently small as in (M5), then we need z such that $\max\{|z_1|_{\mathcal{A}_1}, |z_2|\} \leq \delta$. Moreover, since $q \in \{0, 1\}$ and $\ell \in \{-\nu, \nu\}$ always holds, then we need $\zeta \leq \sqrt{2\delta^2 + 1^2 + \nu^2}$ for practical global attractivity. Furthermore, we observe in simulation that solutions z to $\mathcal{H}$ converge to a neighborhood of $\mathcal{A}$ in the presence of small noise in measurements of the gradient.

6. NUMERICAL EXAMPLE

This example compares multiple solutions to demonstrate the effectiveness of the hybrid algorithm $\mathcal{H}$, both when escaping from local maxima, and when converging from initial points that are not maxima. The algorithm has no knowledge of L, or the location of its critical points, but it uses measurements of ∇L at the current value of z_1. The values of the heavy ball parameters are $\lambda = 145$, and $\gamma = \frac{3}{4}$, and the hybrid algorithm parameter values are $\varepsilon_1 = 0.05$, $\varepsilon_2 = 0.06$, $\rho_1 = 0.05$, $\rho_2 = 0.06$, and $\nu = 1$. The objective function is $L(z_1) = \frac{z_1^2(z_1-10)^2(z_1-20)^2(z_1-30)^2}{10{,}000}$, which has local minima at $\mathcal{A}_{1_{\min}} = \{0, 10, 20, 30\}$ and local maxima at $\mathcal{A}_{1_{\max}} = \{5(3 - \sqrt{5}), 15, 5(3 + \sqrt{5})\}$.

Initial conditions for the simulations are $z_1(0, 0) = \{-1, 5(3 - \sqrt{5}), 6, 15, 24.5, 5(3 + \sqrt{5}), 31\}$, $z_2(0, 0) = 0$, and $q(0, 0) = 0$. Note that the function L and parameter values are the same as those used in Figure 1, with the exception of $\nu = 10^7$, $\varepsilon_2 = 10$, and $\rho_2 = 10$ in Figure 1. The reason ν, ε_2, and ρ_2 are set differently in Figure 2 is that this example includes no noise in measurements of the gradient. Such noise can cause jump times to be different, and so such parameters needed to be tuned accordingly in Figure 1. Recall that Figure 1 shows that the state z_1 converges with our algorithm under arbitrarily small noise in the gradient measurements, when starting close to a local maximum at $z_1 = 15$, whereas for simulated annealing the state z_1 remains stuck at this same local maximum. Although noise in the gradient measurements is not present in Figure 2, it would be easy to see that simulated annealing – which still contains a noise signal – would get stuck when starting at the local maxima at $\mathcal{A}_{1_{\max}} = \{5(3 - \sqrt{5}), 15, 5(3 + \sqrt{5})\}$. In contrast, Figure 2 shows that the hybrid algorithm $\mathcal{H}$ converges to a local minimum from such initial conditions.

Figure 2 shows the evolution of z_1 and z_2 over time for multiple solutions with different initial conditions. Black dots with times labeled in seconds denote when each simulation converges to within 0.01 of $\mathcal{A}_1$ [5] . Conversely, the solutions which start in a small neighborhood of local maxima begin with a jump, followed by a switch to $u = \ell$, then jump again, switching to the heavy ball algorithm, before such solutions converge to a neighborhood of a local minimum. The solutions which do not start at critical

[4] Note that such a point is not itself a critical point, as it is not a stationary point, since L is a Morse function.

[5] Code at github.com/HybridSystemsLab/PGASHeavyBall

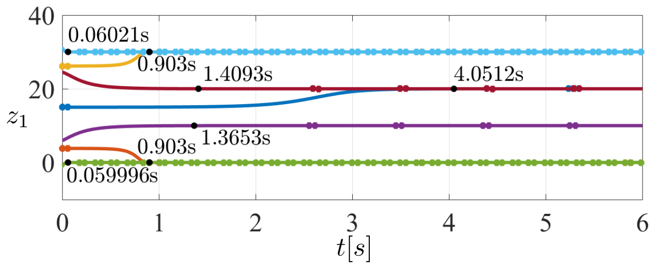

Fig. 2. The evolution of z_1 over time for the hybrid system $\mathcal{H}$, for the objective function $L(z_1) = \frac{z_1^2(z_1-10)^2(z_1-20)^2(z_1-30)^2}{10{,}000}$, with $\mathcal{A}_{1_{\min}} = \{0, 10, 20, 30\}$, $\mathcal{A}_{1_{\max}} = \{5(3-\sqrt{5}), 15, 5(3+\sqrt{5})\}$, and $\varepsilon_1 = 0.05$, $\varepsilon_2 = 0.06$, $\rho_1 = 0.05$, $\rho_2 = 0.06$, $\nu = 1$, $\lambda = 145$, and $\gamma = \frac{3}{4}$. This plot shows different solutions, starting from different initial conditions. Solutions start at local maxima at $z_1(0,0) = 15$, $z_1(0,0) = 5(3-\sqrt{5})$, and $z_1(0,0) = 5(3+\sqrt{5})$, as well as at the points $z_1(0,0) = 6$, $z_1(0,0) = 24.5$, $z_1(0,0) = -1$, and $z_1(0,0) = 31$, which are neither maxima nor minima. All solutions start with $z_2(0,0) = 0$ and $q(0,0) = 0$. Times when each solution converges to within 0.01 of $\mathcal{A}_{1_{\min}}$ are marked with black dots and labeled in seconds. Jumps are labeled with asterisks.

points start with the heavy ball algorithm. Although there are jumps near the local minimum to which such solutions converge, these solutions also do not leave the neighborhood of a local minimum, determined by the values chosen for $(\varepsilon_1, \varepsilon_2, \rho_1, \rho_2)$. Solutions that do not start near critical points converge more quickly than the other solutions in this example, in about 0.06 to about 0.903 seconds. Solutions which start at local maxima converge more slowly than the other solutions in this example, in about 1.4 to 4.05 seconds, as these solutions take more time to build inertia than those which do not start at local maxima.

The particular values chosen for $(\varepsilon_1, \varepsilon_2, \rho_1, \rho_2)$ keeps solutions within a neighborhood of size 0.01 around $\mathcal{A}_{1_{\min}}$. We conjecture that different tunings of $(\varepsilon_1, \varepsilon_2, \rho_1, \rho_2)$ would change the size of such a neighborhood, with larger $(\varepsilon_1, \varepsilon_2, \rho_1, \rho_2)$ yielding a larger neighborhood of $\mathcal{A}_{1_{\min}}$, and smaller $(\varepsilon_1, \varepsilon_2, \rho_1, \rho_2)$ resulting in a smaller neighborhood of $\mathcal{A}_{1_{\min}}$.

7. CONCLUSION AND FUTURE WORK

We developed a hybrid optimization algorithm to detect whether the state z is near a critical point, to ensure convergence to a neighborhood of a local minimizer of a nonconvex Morse objective function L, even when the state $z \in \mathbb{R}^2$ starts at a local maximizer. Designed using hybrid system tools, this algorithm utilizes a switching strategy that uses measurements of the gradient of L. Therefore, the algorithm we present renders the set $\mathcal{A}$ practically globally attractive.

In this paper, we address Morse functions $L : \mathbb{R} \to \mathbb{R}$. Extensions to $L : \mathbb{R}^n \to \mathbb{R}$, however, require techniques to keep the state from becoming stuck in a saddle point. Instead of relying on noise or knowledge of $\nabla^2 L$, our algorithm could be extended to $L : \mathbb{R}^n \to \mathbb{R}$ in the following manner. A restarting scheme, similar to the one proposed in O'Donoghue and Candes (2015), could be employed for escaping saddle points. Such a restarting scheme, which indicates when the "velocity" term z_2 is taking the state z_1 in a bad direction, could also be used to detect a saddle point. Then the algorithm could reset z_2 to move the state z in the correct direction.

REFERENCES

Attouch, H., Goudou, X., and Redont, P. (2000). The heavy ball with friction method, I. the continuous dynamical system: global exploration of the local minima of a real-valued function by asymptotic analysis of a dissipative dynamical system. *Communications in Contemporary Mathematics*, 2(01), 1–34.

Audin, M. and Damian, M. (2014). *Morse Theory and Floer Homology*. Springer.

Baradaran, M., Poveda, J.I., and Teel, A.R. (2018). Stochastic hybrid inclusions applied to global almost sure optimization on manifolds. In *2018 IEEE Conference on Decision and Control (CDC)*, 6538–6543. IEEE.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

Chiang, T.S., Hwang, C.R., and Sheu, S.J. (1987). Diffusion for global optimization in $\mathbb{R}^n$. *SIAM Journal on Control and Optimization*, 25(3), 737–753.

Goebel, R., Sanfelice, R.G., and Teel, A.R. (2012). *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton University Press, New Jersey.

Jin, C., Ge, R., Netrapalli, P., Kakade, S.M., and Jordan, M.I. (2017). How to escape saddle points efficiently. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 1724–1732. JMLR. org.

Kolarijani, A.S., Esfahani, P.M., and Keviczky, T. (2018). Fast gradient-based methods with exponential rate: A hybrid control framework. In *International Conference on Machine Learning*, 2728–2736.

Kolarijani, A.S., Esfahani, P.M., and Keviczky, T. (2019). Continuous-time accelerated methods via a hybrid control lens. *IEEE Transactions on Automatic Control*.

Larson, R., Hostetler, R., and Edwards, B.H. (2007). *Calculus: Early transcendental functions*. Houghton Mifflin, 4th edition.

O'Donoghue, B. and Candes, E. (2015). Adaptive restart for accelerated gradient schemes. *Foundations of computational mathematics*, 15(3), 715–732.

Polyak, B.T. (1964). Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5), 1–17.

Polyak, B.T. (1987). Introduction to optimization. translations series in mathematics and engineering. *Optimization Software*.

Strizic, T., Poveda, J.I., and Teel, A.R. (2017). Hybrid gradient descent for robust global optimization on the circle. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, 2985–2990. IEEE.

Zavriev, S. and Kostyuk, F. (1993). Heavy-ball method in nonconvex optimization problems. *Computational Mathematics and Modeling*, 4(4), 336–341.