A Robust Hybrid Heavy Ball Algorithm for Optimization with High Performance

Dawn M. Hustig-Schultz and Ricardo G. Sanfelice

Abstract— **This paper proposes hybrid control algorithms for optimization of a convex objective function with fast convergence, reduced oscillations, and robustness. Developed using hybrid system tools, the algorithms feature a uniting control strategy, in which two standard heavy ball algorithms, one used globally and another used locally, with properly designed gravity and friction parameters, are employed. The proposed hybrid control strategy switches the parameters to converge quickly to the set of minimizers of the convex objective function without oscillations. A hybrid control algorithm implementing a switching strategy that measures the objective function and its gradient, and another algorithm that only measures its gradient, are designed. Key properties of the resulting closed-loop systems, including existence of solutions, asymptotic stability, and robustness, are analyzed. Numerical results validate the findings.**

I. Introduction

In this paper, the problem of finding the critical points of a scalar, continuously differentiable objective function L is considered. In particular, we are interested in algorithms to solve optimization problems of the form $\min_{\xi \in \mathbb{R}^n} L(\xi)$ by finding the minima of L. The *heavy ball method* is an accelerated gradient method capable of guaranteeing convergence to the set of minimizers of L [1] [2]. Unlike classical gradient descent, the heavy ball method adds a velocity term to the gradient so as to speed up convergence. The method consists of running a dynamical system that, under appropriate assumptions, has all of its solutions converging to the set of critical points of the function L. The dynamical system used in the heavy ball method is given by the second-order system

$$\ddot{\xi} + \lambda \dot{\xi} + \gamma \nabla L(\xi) = 0 \tag{1}$$

where λ and γ are positive tunable parameters; see [2], [1], [3]. This system resembles the dynamics of a particle sliding on a profile defined by L, with friction. In such a setting, the parameter λ represents the ratio between the viscous friction coefficient and the mass of the particle, and γ represents the gravity constant. This heavy ball algorithm can be applied, in its current form, to control a single autonomous agent, and it has potential to be extended to multiagent systems [4], [5], wireless sensor or communication networks [5], [6], and machine learning [5], [7], to name a few applications. The performance of the heavy ball method is highly dependent on λ and γ. Specifically, when λ is large, heavy ball converges very slowly, and when λ is small, heavy ball converges quickly, but with oscillations near the minimum [3]. This behavior suggests the need of a dynamic adaptation of the value of λ for fast convergence without oscillations to be possible.

Global convergence of the heavy ball method for a convex function L is demonstrated in [7]. In [3], convergence bounds for the continuous-time heavy ball method, both when L is convex and when L is a Morse function, are derived, but global asymptotic stability is not established. In [8], a general, continuous-time Lyapunov function for accelerated gradient methods is developed using Bregman Lagrangians. In [9], Lyapunov functions for the heavy-ball method are proposed to establish stability and convergence, including rate estimates. In [4] and [10], global asymptotic stability of the continuous-time heavy ball method, when L is convex and $\mathcal{C}^2$, is demonstrated.

Contrary to classical gradient descent, accelerated gradient methods suffer from error accumulation. The authors in [11], [12], and [13] suggest that the heavy ball method is sensitive to perturbations, due to its acceleration component. In [13], the effect of white noise on the discrete-time heavy-ball method is analyzed, and robustness to such noise is attained through the use of varying step-sizes. In [4], a perturbed continuous-time heavy ball system is analyzed and shown to be robust, but at the expense of the system measuring the Hessian of L. In [10], a continuous-time heavy ball with perturbations is also formulated and analyzed, where the system employs an observer to measure these perturbations.

The main contributions of this paper are as follows. We propose heavy ball control algorithms for optimization of a convex objective function L, with fast convergence and reduced oscillations. The algorithms utilize a uniting control strategy, developed using hybrid system tools [14], which switches between two standard heavy ball algorithms with different gravity and friction parameters. We design a hybrid control algorithm implementing a switching strategy that measures both L and its gradient, and then extend it to the case where it measures only the gradient of L. The algorithms require no measurements of the Hessian of L. We prove global asymptotic stability of the set of minimizers of L for the resulting closed-loop system, and establish robustness results. The arguments we present follow similarly

D. M. Hustig-Schultz and R. G. Sanfelice are with the Department of Electrical and Computer Engineering, University of California, 1156 High Street, Santa Cruz, CA 95064, USA. dhustigs@ucsc.edu, ricardo@ucsc.edu. This research has been partially supported by the National Science Foundation under CAREER Grant no. ECS-1450484, Grant no. ECS- 1710621, and Grant no. CNS-1544396, by the Air Force Office of Scientific Research under Grant no. FA9550-16-1-0015, by the Air Force Research Laboratory under Grant no. FA9453-16-1-0053, and by CITRIS and the Banatao Institute at the University of California.

for the case when $\xi \in \mathbb{R}^n$, but for simplicity we present our results for $n = 1$.

The rest of the paper is organized as follows. Section II contains a brief explanation of notation and the hybrid systems framework. Section III introduces the uniting algorithm and presents its nominal properties, and Section IV pertains to robustness. Due to space constraints, proofs of the results will be published elsewhere.

II. Preliminaries

A. Notation

We denote the real, positive real, and natural numbers $\mathbb{R}$, $\mathbb{R}_{>0}$, and $\mathbb{N}$, respectively. The set C^n represents the family of n-th continuously differentiable functions. For vectors v and w, $|v| = \sqrt{v^\top v}$ denotes the Euclidean vector norm of v, and $\langle v, w\rangle = v^\top w$ the inner product of v and w. The closure of a set S is denoted $\overline{S}$. The distance of a point x to a set S is defined by $|x|_S = \inf_{y\in S} |y - x|$. Given a set-valued mapping $M : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$, the domain of M is the set $\text{dom } M = \{x \in \mathbb{R}^m : M(x) \neq \emptyset\}$. A function $\beta : \mathbb{R}_{>0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a class-$\mathcal{KL}$ function, also written $\beta \in \mathcal{KL}$, if it is nondecreasing in its first argument, nonincreasing in its second argument, $\lim_{r\to 0^+} \beta(r, s) = 0$ for each $s \in \mathbb{R}_{\geq 0}$, and $\lim_{s\to\infty} \beta(r, s) = 0$ for each $r \in \mathbb{R}_{\geq 0}$.

B. Preliminaries on Hybrid Systems

In this paper, a hybrid system $\mathcal{H}$ has data (C, F, D, G) and is defined as [14, Definition 2.2]

$$\mathcal{H} = \begin{cases} \dot{x} \in F(x) & x \in C \\ x^+ \in G(x) & x \in D \end{cases} \tag{2}$$

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. The notation $\rightrightarrows$ indicates that F and G are set-valued maps. A solution ϕ is parameterized by $(t, j) \in \mathbb{R}_{\geq 0} \times \mathbb{N}$, where t is the amount of time that has passed and j is the number of jumps that have occurred. The domain of ϕ, namely, $\text{dom}\phi \subset \mathbb{R}_{\geq 0} \times \mathbb{N}$ is a hybrid time domain, which is a set such that for each $(T, J) \in \text{dom}\phi$, $\text{dom}\phi \cap ([0, T] \times \{0, 1, \ldots, J\}) = \cup_{j=0}^{J}([t_j, t_{j+1}], j)$ 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. The set $\mathcal{S}_\mathcal{H}$ contains all maximal solutions to $\mathcal{H}$. 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 [14, Assumption 6.5].

III. Hybrid Uniting Framework for the Heavy Ball Method

A. Motivation and Problem Statement

As mentioned in Section I, the performance of the heavy ball method for finding the minimizers of an objective function is highly dependent on the choice of γ and λ. In particular, for a fixed value of γ, the choice of the "friction parameter" λ significantly affects the asymptotic behavior of the solutions to (1). In fact, large values of λ give rise to slowly converging solutions, resembling solutions yielded by steepest descent, while smaller values give rise to fast solutions with oscillations getting wilder as λ decreases [3]. This is illustrated in the top two plots in Figure 1. Motivated by these properties, we propose a logic-based algorithm that determines which set of parameters (or algorithm) should be used far from the minimizer and which one should be used nearby it so as to guarantee fast convergence with reduced oscillations. The challenge is that the objective function and the minimizers are unknown, so it is not evident when to switch and how to avoid chattering. The third plot in Figure 1 shows that the improvement provided by the proposed algorithm is significant.

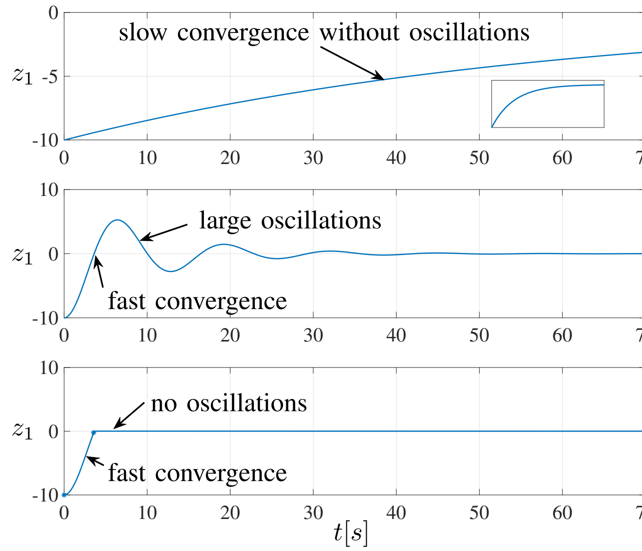

Fig. 1. Comparison of the performance of the heavy ball method, with large and small values of λ, and with proposed logic-based algorithm for $L(z_1) = \frac{1}{4}z_1^2$. Top: when λ is large, heavy ball converges very slowly (zoomed out plot included). Middle: when λ is small, heavy ball converges quickly, but with wild oscillations. Bottom: our proposed logic-based strategy yields fast convergence, with no oscillations.

The problem to solve is stated as follows:

Problem($\star$): Given a scalar, real-valued, continuously differentiable objective function L with a single isolated or a connected continuum of minimizers, design an optimization algorithm that guarantees fast convergence without oscillations, without knowing the function L or the location of its minimizers, and with robustness. $\square$

B. Modeling

We interpret (1) as a control system consisting of a plant and a control algorithm. The plant is given by the double integrator

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} z_2 \\ u \end{bmatrix} =: F_P(z, u) \qquad (z, u) \in \mathbb{R}^2 \times \mathbb{R} =: C_P \tag{3}$$

with output $y = h(z)$, which will adopt different forms in the upcoming sections. The control algorithm leading to (1) is given by $u = \kappa(h(z)) = -\lambda z_2 - \gamma \nabla L(z_1)$. However, to cope with the trade-off between damping oscillations and converging fast, we propose a logic-based algorithm that unites two control algorithms, one with small λ used far from the minimizer to quickly get close to the critical point, and one with large λ used near the minimizer to avoid oscillations. The proposed logic-based algorithm "unites" two individual controllers, or, equivalently, optimization algorithms, that are designed as static state-feedback laws of the form

$$\kappa_q(h(z)) := -\lambda_q z_2 - \gamma_q \nabla L(z_1) \tag{4}$$

for each $q \in Q := \{0, 1\}$ and $z \in \mathbb{R}^2$. The parameters $\lambda_q > 0$ and $\gamma_q > 0$ should be designed for each $q \in Q$, so as to achieve fast convergence without oscillations nearby the minimizer. The algorithm for $q = 1$ will be designed to achieve fast convergence and is referred to as global. The algorithm for $q = 0$ will be designed to achieve stable convergence near the minimizer and is referred to as local. Due to q jumping between 0 and 1 with hysteresis, we refer to the proposed logic-based algorithms as hybrid algorithms.

The design of the hybrid control algorithms is done using the function[1]

$$V_q(z) := \gamma_q \left(L(z_1) - L(\mathcal{A}_1)\right) + \frac{1}{2} z_2^2 \tag{5}$$

defined for each $q \in Q$ and each $z \in \mathbb{R}^2$, where

$$\mathcal{A}_1 = \{z_1 \in \mathbb{R} : \nabla L(z_1) = 0\}\,. \tag{6}$$

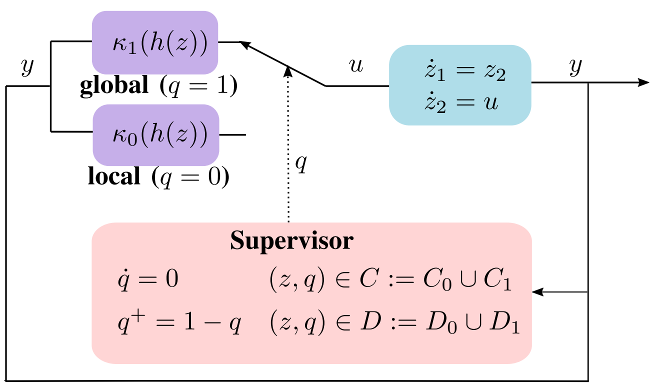

Fig. 2. Feedback diagram of the hybrid closed-loop system $\mathcal{H}$, in (7), uniting global and local optimization algorithms.

The switch between κ_0 and κ_1 is governed by a *supervisor*. The supervisor selects between these two autonomous optimizaton algorithms, based on the plant's output and on the optimization algorithm currently applied. In our simplest algorithm, which is introduced first, the idea is to define sublevel sets of V_q and use hysteresis to switch between the global heavy ball algorithm and the local one. More precisely, when the supervisor is using the global optimization algorithm and $V_1(z) \leq c_{1,0}$ with $c_{1,0}$ small, then z_1 is close to the minimum and a switch to the local algorithm is performed to converge without oscillations. When the supervisor is using the local algorithm and $V_0(z) \geq c_0$ with $c_0 > c_{1,0}$, then z_1 is too far from its minimum and the supervisor switches to the global algorithm to converge quickly to the neighborhood of the minimum. These switching events constitute the jumps in the hybrid closed-loop system, and the c_0- and $c_{1,0}$-sublevel sets need to be properly tuned to solve **Problem**($\star$). At times other than when these events occur, the hybrid algorithm executes the individual optimization algorithm associated with the current value of q, namely, it applies (4) to (3).

To capture the mechanism outlined above, we define the data of the hybrid closed-loop system $\mathcal{H}$ as follows[2]:

$$F(x) := \begin{bmatrix} z_2 \\ \kappa_q(h(z)) \\ 0 \end{bmatrix} \quad \forall x \in C := C_0 \cup C_1 \tag{7a}$$

$$G(x) := \begin{bmatrix} z \\ 1 - q \end{bmatrix} \quad \forall x \in D := D_0 \cup D_1 \tag{7b}$$

Figure 2 shows the feedback diagram of this hybrid closed-loop system $\mathcal{H}$. We denote the closed-loop systems resulting from using the individual heavy ball algorithms (κ_q) as $\mathcal{H}_q$, for each $q \in Q = \{0, 1\}$; i.e., $\mathcal{H}_0$ denotes the local heavy ball algorithm that uses λ_0 and γ_0, and $\mathcal{H}_1$ denotes the global heavy ball algorithm that uses λ_1 and γ_1.

The reader may wonder whether a (nonhybrid) discontinuous algorithm would solve **Problem**($\star$) robustly. Unfortunately, that is not the case since solutions without hysteresis switching may exhibit chatter at the switching surface induced by a discontinuous algorithm. The proposed hybrid systems approach solves the problem with robustness by virtue of hysteresis switching.

C. *Uniting Optimization Algorithm Using Measurements of L and ∇L*

In this section, we present a uniting optimization algorithm with switching rules derived from sublevel sets of V_0 and V_1. This algorithm measures L and ∇L at the current value of z_1. That is, the output of (3) is

$$y = \begin{bmatrix} z_2 \\ \nabla L(z_1) \\ L(z_1) \end{bmatrix} =: h(z). \tag{8}$$

However, the algorithm has no knowledge of the particular objective function L or of its minima.

Based on the outline provided in Section III-B, the supervisor selects κ_0 or κ_1 in (4) using sublevel sets of V_q in (5). When the system measures L and ∇L, these sets are defined as follows. Let the set $\mathcal{U}_0$ be defined by the c_0-sublevel set of V_0, namely,

$$\mathcal{U}_0 := \left\{z \in \mathbb{R}^2 : V_0(z) \leq c_0\right\}. \tag{9}$$

The parameter $c_0 > 0$, along with λ_0 and γ_0, are designed so that $\mathcal{U}_0$ is in the region where κ_0 is to be used. In this design, λ_0 is large to avoid oscillations when converging to the minimum. As we will show in the upcoming results, $\mathcal{U}_0$ is contained in the basin of attraction induced by κ_0 due to the global attractivity property guaranteed by κ_0. Then, roughly speaking, when $q = 0$ and $V_0(z) \geq c_0$, the hybrid closed-loop system will switch to the global algorithm defined by

[1] Since the value of L is the same for all $z_1 \in \mathcal{A}_1$, $L(\mathcal{A}_1)$ is a singleton.

[2] The computational complexity of the hybrid algorithm described in (7) is no more than the complexity of integrating ordinary differential equations, which using parallel processing is $O(\log N)$, given $O(N)$ processors, where N is the number of iterations in the numerical integration method [15].

κ_1. Otherwise, the local algorithm κ_0 is used. In this way, the set $\mathcal{U}_0$ marks the outer portion of the hysteresis implemented by the supervisor.

The set $\mathcal{T}_{1,0}$, which marks the inner portion of the hysteresis mechanism in the supervisor, is defined by a $c_{1,0}$-sublevel of V_1 with $c_{1,0} \in (0, c_0)$ chosen so that $\mathcal{T}_{1,0}$ is contained in the interior of $\mathcal{U}_0$

$$\mathcal{T}_{1,0} := \left\{ z \in \mathbb{R}^2 : V_1(z) \leq c_{1,0} \right\} \tag{10}$$

This choice of $\mathcal{T}_{1,0}$ is possible since $\mathcal{U}_0$ and the sublevel sets of V_1 are compact for small enough constants c_0 and $c_{1,0}$. Then, due to the global attractivity guaranteed by κ_0, once z is in $\mathcal{T}_{1,0}$, the boundary of $\mathcal{U}_0$ will never be reached. When $q = 1$ and $V_1(z) \leq c_{1,0}$, the supervisor will switch from the global algorithm κ_1 to the local algorithm κ_0. Using the sublevel sets constructed above to formulate the switching rules, the flow and jump sets C and D in (7) are identified as follows:

$$C_0 := \left\{ z \in \mathbb{R}^2 : V_0(z) \leq c_0 \right\} \times \{0\} \tag{11a}$$
$$C_1 := \left\{ z \in \mathbb{R}^2 : V_1(z) \geq c_{1,0} \right\} \times \{1\} \tag{11b}$$
$$D_0 := \left\{ z \in \mathbb{R}^2 : V_0(z) \geq c_0 \right\} \times \{0\} \tag{11c}$$
$$D_1 := \left\{ z \in \mathbb{R}^2 : V_1(z) \leq c_{1,0} \right\} \times \{1\} \tag{11d}$$

In some of the results to follow, we will impose the following assumptions on the objective function L.

Assumption 3.1: (Properties of L)

(CM1) L is $\mathcal{C}^1$;

(CM2) There exist class-$\mathcal{K}_\infty$ functions[3] α_1 and α_2 such that $\alpha_1\left(|z_1|_{\mathcal{A}_1}\right) \leq L(z_1) - L(\mathcal{A}_1) \leq \alpha_2\left(|z_1|_{\mathcal{A}_1}\right)$ for all $z_1 \in \mathbb{R}$;

(CM3) ∇L is locally Lipschitz.

(CM4) $\mathcal{A}_1$ is compact and connected.

Under item (CM1) of Assumption 3.1, the hybrid closed-loop system $\mathcal{H}$ in (7), with C and D defined via (11), is well-posed as it satisfies the hybrid basic conditions. Moreover, when Assumption 3.1 holds, every maximal solution to $\mathcal{H}$ is complete and bounded, as stated in the following lemma.

Lemma 3.2: (Existence of solutions to $\mathcal{H}$) Let Assumption 3.1 hold. Then, every maximal solution to the hybrid closed-loop system $\mathcal{H}$ in (7)*, with C and D defined via* (11)*, is bounded and complete.*

The following result establishes that the hybrid closed-loop system $\mathcal{H}$ has the set

$$\mathcal{A} := \left\{ z \in \mathbb{R}^2 : \nabla L(z_1) = z_2 = 0 \right\} \times \{0\} = \mathcal{A}_1 \times \{0\} \times \{0\} \tag{12}$$

globally asymptotically stable[4]. To establish it, we use Lyapunov stability theory and an invariance principle.

Theorem 3.3: (Global asymptotic stability of $\mathcal{A}$ for $\mathcal{H}$) Let L satisfy Assumption 3.1, $\lambda_q > 0$, $\gamma_q > 0$, and $c_{1,0} \in (0, c_0)$. Then, the set $\mathcal{A}$ is globally asymptotically stable for $\mathcal{H}$.

The main points of the proof of Theorem 3.3 are as follows. The individual optimization algorithms $\mathcal{H}_0$ and $\mathcal{H}_1$ satisfy $\dot{V}_q = \langle \nabla V_q(z), F_P(z, \kappa_q(h(z))) \rangle = -\lambda_q z_2^2 \leq 0$ for all $q \in Q$, $\lambda_q > 0$, and $\gamma_q > 0$, and the largest weakly invariant set contained in $\left\{ z \in \mathbb{R}^2 : \dot{V}_q(z) = 0 \right\} \cap \left\{ z \in \mathbb{R}^2 : V_q(z) = r_q \right\}$ is when $r_q = 0$, which is equal to $\mathcal{A}_1$. Therefore, since every maximal solution is bounded and complete by Lemma 3.2, the individual optimization algorithms $\mathcal{H}_0$ and $\mathcal{H}_1$ have $\mathcal{A}_1 \times \{0\}$ globally asymptotically stable. Global asymptotic stability of $\mathcal{A}$ for $\mathcal{H}$ follows from the construction of G and D, which guarantees no more than two jumps in a solution.

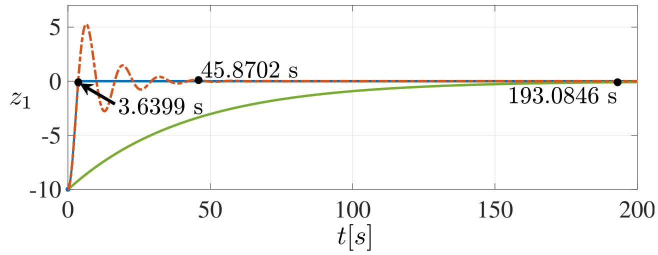

Fig. 3. A comparison of the evolution of z_1 over time for $\mathcal{H}_0$, $\mathcal{H}_1$, and $\mathcal{H}$, for a function L with a single minimum at $\mathcal{A}_1 = \{0\}$. The heavy ball algorithm $\mathcal{H}_1$ uses $\lambda_1 = \frac{1}{5}$ (shown in red) and settles to within a 1% margin of $\mathcal{A}_1$ in about 45.9 seconds. The heavy ball algorithm $\mathcal{H}_0$ uses $\lambda_0 = 10.5$ (shown in green) and settles to within a 1% margin of $\mathcal{A}_1$ in about 193.1 seconds. The hybrid closed-loop system $\mathcal{H}$ (shown in blue) settles to within a 1% margin of $\mathcal{A}_1$ in about 3.6 seconds.

Example 3.4: To show the effectiveness of the hybrid algorithm, we compared it in simulation with the individual optimization algorithms $\mathcal{H}_0$ and $\mathcal{H}_1$. None of the algorithms has knowledge of L, or of the location of its minima, but they have access to the values of L and ∇L at the current value of z_1. For simulation, we used the following parameter values: $\lambda_0 = 10.5$, $\lambda_1 = \frac{1}{5}$, and $\gamma_0 = \gamma_1 = \frac{1}{2}$. The sublevel set values are $c_0 = 12.5$ and $c_{1,0} = 6.3$. The objective function is simply $L(z_1) = \frac{1}{4} z_1^2$, which has a single minima at $\mathcal{A}_1 = \{0\}$. Initial conditions are $z_1(0,0) = -10$, $z_2(0,0) = 0$, and $q(0,0) = 0$. Figure 3 shows the z_1 solution component over time for each of the algorithms[5]. Black dots with times labeled in seconds denote the settling time to within a 1% margin of $\mathcal{A}_1$. Algorithm $\mathcal{H}_1$, using $\lambda_1 = \frac{1}{5}$ and shown in red, reaches the set $\mathcal{A}_1$ quickly, but oscillates wildly until it finally settles to within a 1% margin of $\mathcal{A}_1$ in about 45.9 seconds. Algorithm $\mathcal{H}_0$, using $\lambda_0 = 10.5$ and shown in green, slowly settles to within a 1% margin of $\mathcal{A}_1$ in about 193.1 seconds. The hybrid closed-loop system $\mathcal{H}$, shown in blue, settles to within a 1% margin of $\mathcal{A}_1$ in about 3.6 seconds, which is a 92.0% improvement over $\mathcal{H}_1$ and a 98.1% improvement over $\mathcal{H}_0$.

D. Uniting Optimization Algorithm Using Only Measurements of ∇L

In this section, we propose a switching rule for the hybrid algorithm that does not use measurements of L but rather measures ∇L only. With only such limited information, which emerges in many optimization problems, the switching rule proposed in Section III-C is not implementable. Namely,

[3] α_2 exists for free since $L(z_1) - L(\mathcal{A}_\infty)$ is defined on $\mathbb{R}$ and is positive definite with respect to $\mathcal{A}_1$.

[4] Global asymptotic stability means that all solutions that start close to the set of interest $\mathcal{A}$ stay close to, and converge to, $\mathcal{A}$.

[5] Code at gitHub.com/HybridSystemsLab/UnitingLevelSetsHeavyBall

the algorithm in this section only measures

$$y = \begin{bmatrix} z_2 \\ \nabla L(z_1) \end{bmatrix} =: h(z). \tag{13}$$

As in Section III-C, the algorithm does not have prior knowledge of L or of its minima, but relies on convexity and quadratic growth of L. In fact, in addition to Assumption 3.1, the following assumptions will be imposed in some of the results in this section.

Assumption 3.5: (Convexity and quadratic growth of L)

(C1) L is convex;

(C2) L has quadratic growth away from its minima $\mathcal{A}_1$; i.e., there exists $\alpha > 0$ such that[6] $L(z_1) - L(\mathcal{A}_1) \geq \alpha\, |z_1|^2_{\mathcal{A}_1}$ for all $z_1 \in \mathbb{R}$.

The following lemma relates the size of the gradient at a point to the distance from the point to the set of minimizers $\mathcal{A}_1$.

Lemma 3.6: (ε-suboptimality when L is convex, with quadratic growth) Let L satisfy Assumptions 3.1 and 3.5. Let α come from (C2). For some $\varepsilon > 0$, if $z_1 \in \mathbb{R}$ is such that $|\nabla L(z_1)| \leq \varepsilon\alpha$, then $|z_1|_{\mathcal{A}_1} \leq \varepsilon$.

The ε-suboptimality condition from Lemma 3.6 is typically used as a stopping condition for optimization, as it indicates that the argument of L is close enough to the set of minimizers. We will exploit Lemma 3.6 to determine when the state component z_1 of the hybrid closed-loop system is close enough to the global minimizer to switch to the local optimization algorithm.

To that end, let $\varepsilon_0 > 0$, $\alpha_0 > 0$, $\gamma_0 > 0$ from κ_0, and $c_0 > 0$ be such that

$$\tilde{c}_0 := \varepsilon_0 \alpha_0 > 0 \tag{14a}$$

$$d_0 := c_0 - \gamma_0 \left(\frac{\tilde{c}_0^2}{\alpha_0} \right) > 0. \tag{14b}$$

Then, the set $\mathcal{U}_0$ in (9) is replaced by $\widetilde{\mathcal{U}}_0 := \left\{ z \in \mathbb{R}^2 : |\nabla L(z_1)| \leq \tilde{c}_0, \frac{1}{2} z_2^2 \leq d_0 \right\}$. This set contains the region where the switch to $\mathcal{H}_0$ is made. Note that $\widetilde{\mathcal{U}}_0$ is contained in the basin of attraction induced by κ_0, due to the global attractivity property it guarantees.

To construct the set $\widetilde{\mathcal{T}}_{1,0}$, playing the role of $\mathcal{T}_{1,0}$ in Section III-C, let $\varepsilon_{1,0} \in (0, \varepsilon_0)$, $\alpha_{1,0} > 0$, and $c_{1,0} \in (0, c_0)$ such that

$$\tilde{c}_{1,0} := \varepsilon_{1,0} \alpha_{1,0} \in (0, \tilde{c}_0) \tag{15a}$$

$$d_{1,0} := c_{1,0} - \gamma_1 \left(\frac{\tilde{c}_{1,0}^2}{\alpha_{1,0}} \right) \in (0, d_0) \tag{15b}$$

where γ_1 is from κ_1. Then, the set $\mathcal{T}_{1,0}$ in (10) is replaced by $\widetilde{\mathcal{T}}_{1,0} := \left\{ z \in \mathbb{R}^2 : |\nabla L(z_1)| \leq \tilde{c}_{1,0}, \frac{1}{2} z_2^2 \leq d_{1,0} \right\}$. By construction, this set is contained in $\left\{ z \in \mathbb{R}^2 : |\nabla L(z_1)| < \tilde{c}_0, \frac{1}{2} z_2^2 < d_0 \right\}$, namely, the interior of $\widetilde{\mathcal{U}}_0$. Let

$$D_1 := \widetilde{\mathcal{T}}_{1,0} \times \{1\} \tag{16}$$

be the set where $q = 1$ switches to $q = 0$. That is, when $(z_1, q) \in D_1$, the supervisor switches from $\mathcal{H}_1$ to $\mathcal{H}_0$. By Lemma 3.6, $|\nabla L(z_1)| \leq \tilde{c}_{1,0}$ implies $|z_1|_{\mathcal{A}_1} \leq \frac{\tilde{c}_{1,0}}{\alpha_{1,0}}$. Therefore, when $(z_1, q) \in D_1$, then $|z_1|_{\mathcal{A}_1} \leq \frac{\tilde{c}_{1,0}}{\alpha_{1,0}}$. Since points $z \in D_1$ satisfy $\frac{1}{2} z_2^2 \leq d_{1,0}$ then each $z \in D_1$ satisfies $V_0(z) = \gamma_0 \left(L(z_1) - L(\mathcal{A}_1) \right) + \frac{1}{2} z_2^2 \leq \gamma_0 \alpha_2 \left(\frac{\tilde{c}_{1,0}}{\alpha_{1,0}} \right) + d_{1,0} =: r_0$, where $r_0 > 0$ and α_2 comes from (CM2). In order to ensure that solutions do not reach D_0 from D_1, choose $r_0' > r_0$, and define

$$D_0 := \left\{ z \in \mathbb{R}^2 : \gamma_0 \alpha_1 \left(|z_1|_{\mathcal{A}_1} \right) + \frac{1}{2} z_2^2 \geq r_0' \right\} \times \{0\} \tag{17}$$

Then, if $z \in D_0$, then $V_0(z) \geq r_0'$, which implies that z_1 is far from the minimum, the supervisor in the hybrid algorithm switches from $\mathcal{H}_0$ to $\mathcal{H}_1$. It follows that the sets in (16) and (17) define the jump sets of $\mathcal{H}$ in (7).

The flow set is defined via

$$C_0 := \overline{(\mathbb{R}^2 \times \{0\}) \setminus D_0}, \ C_1 := \overline{(\mathbb{R}^2 \times \{1\}) \setminus D_1} \tag{18}$$

The hybrid algorithm $\mathcal{H}$, with C and D defined via (18), is well-posed, as it satisfies the hybrid basic conditions [14, Assumption 6.5].

When the objective function L satisfies Assumptions 3.1 and 3.5, every maximal solution to the hybrid closed-loop system $\mathcal{H}$ is complete and bounded. Additionally, every solution, with the exception of those such that $|\nabla L(z_1(0,0))| = \tilde{c}_0$, $\frac{1}{2} z_2(0,0)^2 = d_0$, and $q(0,0) = 0$, are unique, as stated in the following lemma.

Lemma 3.7: (Existence of solutions to $\mathcal{H}$) Let L satisfy Assumption 3.1 and Assumption 3.5. Then, every maximal solution to $\mathcal{H}$, with C and D as defined via (17), (16), *and* (18), *is bounded and complete. Furthermore, if* $c_{1,0} \in (0, c_0)$, $\tilde{c}_{1,0} \in (0, \tilde{c}_0)$ *in* (14a) *and* (15a), *and* $d_{1,0} \in (0, d_0)$ *in* (14b) *and* (15b), *each solution x to $\mathcal{H}$ from $x(0,0) = \begin{bmatrix} z_1(0,0) \\ q(0,0) \end{bmatrix}$ is unique, except for those such that $\gamma_0 \alpha_1 \left(|z_1(0,0)|_{\mathcal{A}_1} \right) + \frac{1}{2} z_2(0,0)^2 = r_0'$ and $q(0,0) = 0$.*

The following result shows that the hybrid closed-loop system $\mathcal{H}$, with C and D as defined via (17), (16), and (18), has the set $\mathcal{A}$ in (12) globally asymptotically stable.

Theorem 3.8: (Global asymptotic stability of $\mathcal{H}$) Let L satisfy Assumptions 3.1 and 3.5, and let $c_{1,0} \in (0, c_0)$, $\tilde{c}_{1,0} \in (0, \tilde{c}_0)$ *in* (14a) *and* (15a), *and* $d_{1,0} \in (0, d_0)$ *in* (14b) *and* (15b). *Then, the set $\mathcal{A}$ is globally asymptotically stable for $\mathcal{H}$, when C and D are defined via* (17), (16), *and* (18).

Example 3.9: As in Example 3.4, we compare the individual optimization algorithms $\mathcal{H}_0$ and $\mathcal{H}_1$ with the hybrid algorithm $\mathcal{H}$. None of the algorithms has knowledge of L, or of the location of its minima, but they measure ∇L at the current value of z_1. For simulation, $\lambda_0 = 11.5$, $\lambda_1 = \frac{1}{5}$, $\gamma_0 = \frac{2}{3}$, and $\gamma_1 = \frac{1}{2}$. To calculate the sets $\widetilde{\mathcal{U}}_0$ and $\widetilde{\mathcal{T}}_{1,0}$, as well as r_0, we use $\alpha_0 = \alpha_{1,0} = 1$, $\varepsilon_0 = 4$, $\varepsilon_{1,0} = 0.7$, and sublevel set values $c_0 = 12.5$, $c_{1,0} = 6.39$. Additionally, $r_0' = r_0 + 1$. For class-$\mathcal{K}_\infty$ functions, we use $\alpha_1(u) = \frac{1}{6} u^2$

[6] Item (C2) can be relaxed to a ball $\mathcal{B}_\nu^L = \{ z_1 : L(z_1) - L(\mathcal{A}_1) < \nu \}$ for some $\nu > 0$.

and $\alpha_2(u) = \frac{1}{2}u^2$. The objective function is simply $L(z_1) = \frac{1}{4}z_1^2$, which has a single minimum at $\mathcal{A}_1 = \{0\}$. Initial conditions are $z_1(0,0) = -10$, $z_2(0,0) = 0$, and $q(0,0) = 1$. Figure 4 shows the z_1 solution component with time for each of the algorithms[7]. Black dots with times labeled in seconds denote when each heavy ball algorithm settles to within a 1% margin of $\mathcal{A}_1$. Algorithm $\mathcal{H}_1$, using $\lambda_1 = \frac{1}{5}$ and $\gamma_1 = \frac{1}{2}$, shown in red, settles to within 1% of $\mathcal{A}_1$ in about 45.9 seconds. Algorithm $\mathcal{H}_0$, using $\lambda_0 = 11.5$ and $\gamma_0 = \frac{2}{3}$, shown in green, slowly settles to within 1% of $\mathcal{A}_1$ in about 158.6 seconds. The hybrid closed-loop system $\mathcal{H}$, shown in blue, settles to within 1% of $\mathcal{A}_1$ in about 3.6 seconds, a 92.1% improvement over $\mathcal{H}_1$ and a 97.7% improvement over $\mathcal{H}_0$.

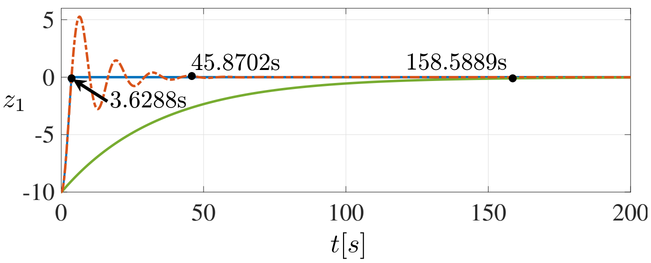

Fig. 4. A comparison of the evolution of z_1 over time for $\mathcal{H}_0$, $\mathcal{H}_1$, and $\mathcal{H}$ – which uses measurements of $\nabla L(z_1)$ – for $L(z_1) = \frac{1}{4}z_1^2$, with a single minimum at $\mathcal{A}_1 = \{0\}$. The global heavy ball algorithm $\mathcal{H}_1$ uses $\lambda_1 = \frac{1}{5}$ and $\gamma_1 = \frac{1}{2}$ (shown in red) and settles to within a 1% margin of $\mathcal{A}_1$ in about 45.9 seconds. The local heavy ball algorithm $\mathcal{H}_0$ uses $\lambda_0 = 11.5$ and $\gamma_0 = \frac{2}{3}$ (shown in green) and settles to within 1% of $\mathcal{A}_1$ in about 158.6 seconds. The hybrid closed-loop system $\mathcal{H}$ (shown in blue) settles to within 1% of $\mathcal{A}_1$ in about 3.6 seconds.

IV. Robustness of $\mathcal{H}$ to Small Perturbations

In Section III we constructed a hybrid closed-loop system $\mathcal{H}$, with flow and jump sets as defined in (7), (11), and (18), that have data satisfying [14, Assumption 6.5]. As a result, we can affirm that the global asymptotic stability of $\mathcal{A}$ on $\mathcal{H}$ that was proved in Theorems 3.3 and 3.8, is robust to small (general) perturbations. In fact, the algorithm defined in (7) is robust to any perturbation that fits in the outer perturbed model of the hybrid system; see the general model in [14, Chapter 8], which allows for sensing and actuation noise. The following result proves robustness for measurement noise.

Theorem 4.1: (Robustness of $\mathcal{KL}$ asymptotic stability) Suppose the hybrid system $\mathcal{H}$ in (7) *has the compact set $\mathcal{A}$ globally asymptotically stable*[8]*. Then, for every compact set $K \subset \mathbb{R}^2 \times Q$ and every $\epsilon > 0$, there exists δ_ρ such that for each perturbation $|\rho| \leq \delta_\rho$ to the measurements of ∇L, every solution to it from K satisfies $|x(t,j)|_{\mathcal{A}} \leq \beta(|x(0,0)|_{\mathcal{A}}, t+j) + \epsilon$ for all $(t,j) \in \mathrm{dom}\, x$.*

V. Conclusion and Future Work

We developed a hybrid algorithm for the heavy ball method, using hybrid system tools. the algorithm renders the set globally asymptotically stable, with robustness, fast convergence, and reduced oscillations. Two different sets of switching rules were derived: when L and ∇L are measured, and when only ∇L is measured. We showed how hybrid systems tools can be leveraged to ensure fast convergence, with reduced oscillations, on a convex scalar objective function. Many problems of interest, however, are represented by nonconvex functions, which have multiple local minima and maxima, and a common pitfall in such a case involves getting stuck at a local maximum. For future work, we will develop a hybrid system for optimization on scalar Morse functions, which have isolated extrema, nondegenerate critical points, and form a dense subset of all smooth C^2 functions.

[7]Code at gitHub.com/HybridSystemsLab/UnitingQuadraticGrowthHeavyBall

[8]Theorems 3.3 and 3.8 establish global asymptotic stability of $\mathcal{A}$.

References

[1] B. T. Polyak, "Introduction to optimization. translations series in mathematics and engineering," *Optimization Software*, 1987.

[2] ——, "Some methods of speeding up the convergence of iteration methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5, pp. 1–17, 1964.

[3] H. Attouch, X. Goudou, and P. Redont, "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*, vol. 2, no. 01, pp. 1–34, 2000.

[4] S. Michalowsky and C. Ebenbauer, "The multidimensional n-th order heavy ball method and its application to extremum seeking," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 2660–2666.

[5] E. Ghadimi, "Accelerating convergence of large-scale optimization algorithms," Ph.D. dissertation, KTH Royal Institute of Technology, 2015.

[6] J. Liu, A. Eryilmaz, N. B. Shroff, and E. S. Bentley, "Heavy-ball: A new approach to tame delay and convergence in wireless network optimization," in *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. IEEE, 2016, pp. 1–9.

[7] E. Ghadimi, H. R. Feyzmahdavian, and M. Johansson, "Global convergence of the heavy-ball method for convex optimization," in *14th IEEE European Control Conference*, 2015, pp. 310–315.

[8] A. C. Wilson, B. Recht, and M. I. Jordan, "A lyapunov analysis of momentum methods in optimization," *arXiv preprint arXiv:1611.02635*, 2016.

[9] B. Polyak and P. Shcherbakov, "Lyapunov functions: An optimization theory perspective," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 7456–7461, 2017.

[10] S. Michalowsky and C. Ebenbauer, "Extremum control of linear systems based on output feedback," in *55th IEEE Conference on Decision and Control*, 2016, pp. 2963–2968.

[11] L. Lessard, B. Recht, and A. Packard, "Analysis and design of optimization algorithms via integral quadratic constraints," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 57–95, 2016.

[12] O. Devolder, F. Glineur, and Y. Nesterov, "First-order methods of smooth convex optimization with inexact oracle," *Mathematical Programming*, vol. 146, no. 1-2, pp. 37–75, 2014.

[13] N. Flammarion and F. Bach, "From averaging to acceleration, there is only a step-size," in *Conference on Learning Theory*, 2015, pp. 658–695.

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

[15] P. Besarab, "Computational complexity of integrating ordinary differential equations," *Cybernetics and Systems Analysis*, vol. 32, no. 3, pp. 417–423, 1996.