

Accelerated Convex Optimization via Hamiltonian Dynamics with Deterministic Integration Time

Xiuyuan Wang*

Department of Computer Science, Yale University

XIUYUAN.WANG@YALE.EDU

Vishwak Srinivasan*

Department of EECS, Massachusetts Institute of Technology

VISHWAKS@MIT.EDU

Qiang Fu

Department of Computer Science, Yale University

QIANG.FU@YALE.EDU

Siddharth Mitra

Department of Computer Science, Yale University

SIDDHARTH.MITRA@YALE.EDU

Ashia Wilson

Department of EECS, Massachusetts Institute of Technology

ASHIA@MIT.EDU

Andre Wibisono

Department of Computer Science, Yale University

ANDRE.WIBISONO@YALE.EDU

Editors: Steve Hanneke and Tor Lattimore

Abstract

We develop Hamiltonian dynamics-based algorithms for smooth convex optimization that achieve accelerated rates of convergence. By exploiting contraction of averaged Hamiltonian flow trajectories rather than requiring contraction at trajectory endpoints, we show that Hamiltonian dynamics-based optimization methods admit deterministic and accelerated convergence guarantees, extending prior work that is limited to quadratic objectives or holds only in expectation. We analyze an idealized continuous-time algorithm and derive practical discrete-time implementations with optimal first-order complexity, thereby establishing Hamiltonian dynamics as a useful algorithmic primitive for deterministic accelerated convex optimization.

Keywords: Convex optimization, first-order optimization, Hamiltonian dynamics, acceleration

1. Introduction

Algorithms to optimize a function are ubiquitous in science and engineering, and they have attracted increased attention in recent years especially in machine learning, computer science, and statistics owing to the need to solve large-scale high-dimensional computational problems. Recent algorithmic designs and analyses have drawn insights from continuous-time dynamical systems, both to obtain a better understanding of existing optimization algorithms and their convergence analyses, as well as to design new optimization algorithms with rigorous theoretical guarantees.

In this work, we study the optimization problem

$$\min_{x \in \mathbb{R}^d} f(x)$$

where $f: \mathbb{R}^d \rightarrow \mathbb{R}$ is a differentiable convex function, and we assume access to the gradient oracle $\nabla f: \mathbb{R}^d \rightarrow \mathbb{R}^d$. This is classically referred to as *first-order convex optimization*. We propose a

*. Both authors contributed equally.

novel accelerated optimization algorithm based on *Hamiltonian dynamics*, which has been highly successful in the closely related algorithmic task of sampling.

The simplest gradient-based method for minimizing f is the gradient descent algorithm. Gradient descent can be viewed as the forward-Euler discretization of the *gradient flow* dynamics, which is a simple continuous-time greedy dynamics that converges to a minimizer of f when f is convex.

$$\underbrace{\dot{X}_t = -\nabla f(X_t)}_{\text{Gradient flow of } f} \xrightarrow{\text{forward Euler}} \underbrace{x_{k+1} = x_k - \eta_k \nabla f(x_k)}_{\text{Gradient descent of } f} .$$

The choice of step size η_k determines the rate of convergence of gradient descent to a minimizer of f , which is often chosen based on properties of f , or via a backtracking line search scheme. When f is known to be L -smooth (see [A_{L-sm}](#)), a recommended choice for η_k is $1/L$. In the table below, we present the scalings of time T for the continuous-time dynamics, and the number of the iterations K for the discrete-time algorithm, to return a point \hat{x} such that $f(\hat{x}) - f^* \leq \varepsilon$ in the settings when f is either convex or strongly convex, in addition to being L -smooth.

	Gradient Flow	Gradient Descent
f is convex	$1/\varepsilon$	L/ε
f is α -strongly convex	$(1/\alpha) \log(1/\varepsilon)$	$(L/\alpha) \log(1/\varepsilon)$

A natural question is if the complexity achieved by gradient flow and gradient descent can be improved. The seminal work of [Nesterov \(1983\)](#) develops a discrete-time accelerated gradient descent algorithm whose complexity are quantitatively better than gradient descent for smooth convex optimization. Notably, this method is *optimal* in the sense that it achieves the best possible complexity for this problem in discrete time ([Nesterov, 2018](#), §2.1). However, both the method and its analysis was rather delicate, and there have been many alternative constructions and analyses of accelerated algorithms ([Drori and Teboulle, 2014](#); [Bubeck et al., 2015](#); [Allen-Zhu and Orecchia, 2017](#); [Drusvyatskiy et al., 2018](#); [Wilson et al., 2019](#)).

A dynamical perspective of [Nesterov's](#) method was presented in [Su et al. \(2016\)](#), where they demonstrate accelerated gradient descent is a discretization of a second-order continuous-time dynamics:

$$\underbrace{\ddot{X}_t + \beta_t \dot{X}_t + \nabla f(X_t) = 0}_{\text{Accelerated Gradient Flow of } f} \xrightarrow{\text{discretize}} \text{Accelerated Gradient Descent of } f .$$

[Su et al. \(2016\)](#) shows the damping factor is $\beta_t = 3/t$ for convex f , and shows accelerated gradient flow has a matching convergence rate as accelerated gradient descent. For α -strongly convex f , [Wilson et al. \(2021\)](#) show that accelerated gradient flow with $\beta_t = 2\sqrt{\alpha}$ converges faster than the gradient flow, and that a discretization of these dynamics is equivalent to an accelerated method for strongly convex function ([Nesterov, 2018](#), §2.2.2). See table below for a summary of the rates.

	Accelerated Gradient Flow	Accelerated Gradient Descent
f is convex	$1/\sqrt{\varepsilon}$	$\sqrt{L/\varepsilon}$
f is α -strongly convex	$(1/\sqrt{\alpha}) \log(1/\varepsilon)$	$\sqrt{L/\alpha} \log(1/\varepsilon)$

While the accelerated gradient method continues to be studied past these developments, in recent years, there have been new approaches to improve the complexity of gradient descent via a more principled choice of stepsizes $\{\eta_k\}_{k \geq 0}$; see [Grimmer \(2024\)](#) for an overview of recent work. Among these approaches, the best known rates for minimizing L -smooth convex and α -strongly convex f scale as L/k^θ and $(1 - (\alpha/L)^{1/\theta})^k$, respectively, where $\theta = \log_2(1 + \sqrt{2}) \approx 1.2716$ ([Altschuler and Parrilo, 2025a,b](#)). These approaches are noteworthy in how they deviate from classical gradient descent analyses which advocated for small stepsizes to enforce a descent method.

In this vein, our work deviates from classical dynamics and algorithm design for methods that achieve accelerated rates of convergence for smooth (strongly) convex optimization by adopting ideas from sampling while retaining a dynamical perspective. We specifically take inspiration from the *Hamiltonian Monte Carlo* (HMC) ([Duane et al., 1987](#); [Neal, 2011](#)), which is an algorithmic framework for sampling based on the Hamiltonian flow (HF):

$$\dot{X}_t = Y_t, \quad \dot{Y}_t = -\nabla f(X_t). \quad (\text{HF})$$

HMC-based samplers are widely used in practice; a popular example is the No-U-Turn Sampler (NUTS) ([Hoffman and Gelman, 2014](#)), which underlies modern probabilistic programming languages such as Stan ([Carpenter et al., 2017](#)), and is regarded as a leading approach for high-dimensional Bayesian inference. In recent years, the perspective of sampling as performing optimization in the space of measures ([Wibisono, 2018](#)) have led to better analyses of sampling algorithms (e.g., [Dalalyan \(2017a,b\)](#); [Durmus et al. \(2019\)](#)) and the development of sampling algorithms that translate ideas from optimization (e.g., [Salim et al. \(2019\)](#); [Zhang et al. \(2020\)](#); [Ma et al. \(2021\)](#); [Srinivasan et al. \(2025\)](#); [Chen et al. \(2025\)](#)). In contrast, our work takes the opposite perspective, where we try to translate ideas from sampling to develop better optimization algorithms.

The Hamiltonian dynamics-based algorithm we study for optimization augments the position space \mathbb{R}^d we want to optimize over with a velocity space \mathbb{R}^d , to get the *phase space* \mathbb{R}^{2d} over which the Hamiltonian dynamics is defined (see [Section 2.1](#) for a more detailed overview). The most basic Hamiltonian dynamics-based optimization algorithms – which we call **HFopt** – performs the following iteratively: from current iterate x_{in} , do

$$\left. \begin{array}{l} \text{Step (i)} \quad \text{Lift to phase space as } (X_0, Y_0) = (x_{\text{in}}, \mathbf{0}) \in \mathbb{R}^{2d}. \\ \text{Step (ii)} \quad \text{Simulate (HF) for integration time } T > 0 \text{ to reach } (X_T, Y_T). \\ \text{Step (iii)} \quad \text{Output } x_{\text{out}} := X_T \text{ as the next iterate.} \end{array} \right\} \quad (\text{HFopt})$$

This scheme results in a non-increasing sequence of function values, which is due to the conservation property of the Hamiltonian dynamics. To the best of our knowledge, this was first proposed in [Teel et al. \(2019\)](#), and they show a non-accelerated rate of convergence. [Wang \(2025\)](#) studies the specific setting of minimizing strongly convex quadratics, and show that a careful choice of integration times can result in an accelerated rate of convergence. Subsequent work by [Fu and Wibisono \(2025\)](#) advance this frontier by showing that **HFopt** with *randomized* integration times achieve accelerated convergence rates for general smooth convex and strongly convex objective functions, albeit the guarantees being in expectation over the randomness of the integration time. See [Appendix A](#) for additional discussion on related works.

1.1. Our contributions

A key insight we develop in this work is in how time averages of the solutions of **HF** behave when f is convex; this is formalized as [Lemma 2](#) and [Corollary 1](#). Essentially, we discover that an average

Algorithm 1 Hamiltonian Flow for optimization with Averaging (HFA)

Input : Initialization: $X_0 \in \mathbb{R}^d$; number of iterations: $K \in \mathbb{N}$; parameter: $\lambda \geq 0$; integration time sequence: $\{T_1, \dots, T_K\}$.

for $k = 1$ **to** K **do**

Solve **HF** from $(X_0^{(k)}, Y_0^{(k)}) = (X_{k-1}, \mathbf{0})$ for time T_k to obtain trajectory $((X_t^{(k)}, Y_t^{(k)}))_{t \in [0, T_k]}$.

Compute $X^{\text{avg}}(X_{k-1}; T_k) = \frac{2}{T_k^2} \int_0^{T_k} (T_k - t) X_t^{(k)} dt$.

Set $X_k = \frac{1}{\lambda + 1} X^{\text{avg}}(X_{k-1}; T_k) + \frac{\lambda}{\lambda + 1} X_{T_k}^{(k)}$.

end

return X_K

of the positions $(X_t)_{t \in [0, T]}$ along **HF** over a sufficiently long integration time T is more conducive to the goal of optimizing f rather than simply the endpoint X_T used in **HFopt**; we provide an example to illustrate this in [Appendix E.1.1](#). Specifically, when f satisfies the α -quadratic growth condition (which is implied by α -strong convexity), we find a good choice of the integration time scales as $1/\sqrt{\alpha}$. Notably, this scaling coincides with the expected integration time considered by [Fu and Wibisono \(2025\)](#) in their randomized version of **HFopt** in the strongly convex setting. This can also be contrasted to short integration times for which non-accelerated guarantees for **HFopt** have been shown, see ([Fu and Wibisono, 2025](#), Theorem 1). This insight forms the basis of our proposed idealized algorithm ‘‘Hamiltonian Flow for optimization with Averaging’’ (**HFA**), shown as [Algorithm 1](#). (Here ‘‘idealized’’ means we assume we can simulate the continuous-time Hamiltonian flow **HF** exactly.) For minimizing convex and strongly convex f , we show in [Section 3](#) that **HFA** achieves accelerated convergence rates, matching those of the accelerated gradient flow; see [Theorem 2](#) for the convex case, and [Theorem 1](#) for the strongly convex case.

While the Hamiltonian flow **HF** provide valuable algorithmic guidance, these dynamics admit exact solutions only in special cases. Therefore, to implement the **HFA** as a concrete discrete-time algorithm, we discretize **HF** via a numerical integrator, and replace the continuous-time average with a discrete average of the iterates. In particular, we discretize the Hamiltonian flow using the *extragradient integrator* (which approximates the implicit integrator, see [Section 2.1](#) for a review). Our discrete-time version of **HFA** is presented in [Section 4](#). To the best of our knowledge, our algorithm differs from previous strategies for designing accelerated gradient methods based on averaging ([d’Aspremont et al., 2021](#)). We show that the iteration complexity of this algorithm to find an ε -approximate solution has the same scaling as the iteration complexity of Nesterov’s accelerated gradient method for smooth convex and strongly convex optimization presented in [Table 1](#), see [Theorem 4](#) for the convex case and [Theorem 3](#) for the strongly convex case. Furthermore, we emphasize the rates of our Hamiltonian-based algorithms are *deterministic*, in contrast those in the previous work of [Fu and Wibisono \(2025\)](#) which only hold in expectation.

2. Preliminaries

Notation. Throughout the paper, we consider minimizing a differentiable function $f: \mathbb{R}^d \rightarrow \mathbb{R}$. The minimum of f is assumed to exist uniquely, which we denote with x^* . For vectors v, w , the Euclidean inner product is $\langle v, w \rangle$, and $\|v\| = \sqrt{\langle v, v \rangle}$ denotes the Euclidean norm of v . For a

time-dependent quantity g_t , we use the shorthand $\dot{g}_t = \frac{d}{dt}g_t$ when the derivative exists. For a real number a , the ceiling $\lceil a \rceil$ denotes the smallest integer that is greater than or equal to a . We say that \hat{x} is an ε -accurate minimizer of f if $f(\hat{x}) - f(x^*) \leq \varepsilon$.

Assumptions. We begin by introducing structural properties of f .

A_{cvx} (**Convexity:**) for any $x, x' \in \mathbb{R}^d$, $f(x') \geq f(x) + \langle \nabla f(x), x' - x \rangle$.

A_{L-sm} (**L-Smoothness:**) for any $x, x' \in \mathbb{R}^d$, $f(x') \leq f(x) + \langle \nabla f(x), x' - x \rangle + \frac{L}{2} \|x' - x\|^2$.

A _{α -qg} (**α -Quadratic Growth:**) for any $x \in \mathbb{R}^d$, $f(x) \geq f(x^*) + \frac{\alpha}{2} \|x - x^*\|^2$.

A _{α -sc} (**α -Strong Convexity:**) for any $x, x' \in \mathbb{R}^d$, $f(x') \geq f(x) + \langle \nabla f(x), x' - x \rangle + \frac{\alpha}{2} \|x - x'\|^2$.

For the key property of the Hamiltonian dynamics that we derive in [Section 3](#), it is sufficient for [A_{cvx}](#) and [A_{L-sm}](#) to hold. Note that [A _{\$\alpha\$ -sc}](#) implies [A _{\$\alpha\$ -qg}](#) and [A_{cvx}](#), and our algorithmic results only require [A_{cvx}](#) and [A _{\$\alpha\$ -qg}](#). The assumptions [A_{cvx}](#) and [A _{\$\alpha\$ -qg}](#) imply for any $x \in \mathbb{R}^d$, $\langle \nabla f(x), x - x^* \rangle \geq \frac{\alpha}{2} \|x - x^*\|^2$; this is the *restricted strong convexity* condition ([Zhang and Yin, 2013](#), Definition 3), which is weaker than strong convexity, and implies f satisfies the Polyak-Łojasiewicz inequality, see [Karimi et al. \(2016, Appendix A\)](#). When f satisfies [A_{L-sm}](#) and [A _{\$\alpha\$ -qg}](#), we use $\kappa := L/\alpha$ to denote the *condition number* of f .

2.1. Hamiltonian Dynamics

We review the Hamiltonian dynamics, which is the key ingredient of the algorithms in this paper. Recall we seek to minimize an objective function $f: \mathbb{R}^d \rightarrow \mathbb{R}$. We define the *energy function* or the *Hamiltonian* $H: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ by

$$H(x, y) = f(x) + \frac{1}{2} \|y\|^2.$$

In classical mechanics, x denotes position and y denotes momentum (or velocity), and the Hamiltonian can be interpreted as the sum of the potential energy $f(x)$ and the kinetic energy $\frac{1}{2} \|y\|^2$ of a unit-mass particle. The Hamiltonian dynamics or Hamiltonian flow is a system of differential equations that describes the evolution of the position and velocity $(X_t, Y_t) \in \mathbb{R}^{2d}$ over time $t \geq 0$ via

$$\dot{X}_t = \nabla_y H(X_t, Y_t) = Y_t, \quad \dot{Y}_t = -\nabla_x H(X_t, Y_t) = -\nabla f(X_t). \quad (\text{HF})$$

A key property of the Hamiltonian dynamics is that the energy function of the system is conserved for all time $t \geq 0$; this can be shown by direct computation (see [Appendix B](#) for a review).

Lemma 1 *For any $t \geq 0$ and initial conditions (X_0, Y_0) , $f(X_t) + \frac{1}{2} \|Y_t\|^2 = f(X_0) + \frac{1}{2} \|Y_0\|^2$.*

The Hamiltonian flow map $\text{HF}: \mathbb{R}^{2d} \times [0, \infty) \rightarrow \mathbb{R}^{2d}$ maps a tuple (x, y) and time t to the solution of [HF](#) at time t from initial condition $(X_0, Y_0) = (x, y)$ i.e.,

$$\text{HF}((x, y); t) = (X_t, Y_t).$$

As discussed in [Section 1](#), the Hamiltonian flow map cannot be computed explicitly for a general potential f . A setting in which this map is available is when f is quadratic, which is leveraged by [Wang \(2025\)](#) to develop accelerated algorithms for minimizing strongly convex quadratic functions.

Numerical integrators. We discuss two numerical integrators to approximate $\text{HF}((x, y); t)$: the implicit integrator and the extragradient integrator, both of which depend on a step size $\eta > 0$. Below, $n \in \mathbb{N}_0$ denotes the discrete-time index.

Definition 1 *The implicit integrator of Hamiltonian flow (HF) from initial values (x_0, y_0) with step size $\eta > 0$ generates the sequence $\{(x_n, y_n)\}_{n \geq 0}$ where (x_n, y_n) satisfies the recursion*

$$\begin{aligned} x_{n+1} &= x_n + \eta y_{n+1}; \\ y_{n+1} &= y_n - \eta \nabla f(x_{n+1}). \end{aligned} \tag{HF^{imp}}$$

The implicit integrator is named as such since given the current iterate (x_n, y_n) , the next iterate (x_{n+1}, y_{n+1}) is obtained by solving an implicit update. We choose the implicit integrator as a starting point due to: (a) its dissipative nature for convex Hamiltonian, which is sufficient for a descent scheme; and (b) its numerical stability, which does not require smoothness of the Hamiltonian. The implicit integrator can be implemented, for example, using a proximal oracle for f in addition to a gradient oracle as stated below (see [Appendix B.2](#)).

$$\begin{aligned} x_{n+1} &= \text{Prox}_{\eta^2 f}(x_n + \eta y_n); \\ y_{n+1} &= y_n - \eta \nabla f(x_{n+1}). \end{aligned}$$

However, a proximal oracle is stronger than having access to gradients of f . To circumvent this, [Fu and Wibisono \(2025\)](#) propose the *extragradient integrator*, which approximates a step of the implicit integrator. This specifically operates by replacing the call to the proximal oracle for f by a gradient descent step, defined below.

Definition 2 *The extragradient integrator of Hamiltonian flow (HF) from initial values (x_0, y_0) with step size $\eta > 0$ generates the sequence $\{(x_n, y_n)\}_{n \geq 0}$ where (x_n, y_n) satisfies the recursion*

$$\begin{aligned} x_{n+1/2} &= x_n + \eta y_n; \\ x_{n+1} &= x_{n+1/2} - \eta^2 \nabla f(x_{n+1/2}); \\ y_{n+1} &= y_n - \eta \nabla f(x_{n+1}). \end{aligned} \tag{HF^{extg}}$$

Since we are interested in the first-order methods for minimizing f , we focus on the extragradient in the main text, and discuss further details of the integrators in [Appendix B.2](#). For either integrator, the output after n discrete iterations with step size η can be viewed as approximating the solution of HF at continuous time $t = n \times \eta$.

3. Exact Hamiltonian Flow for minimizing convex f

3.1. Properties of the average position along the Hamiltonian flow trajectory

The principle of energy conservation (see [Lemma 1](#)) is a key property of the Hamiltonian flow, and does not place any assumptions on the potential f beyond differentiability. As a consequence of energy conservation, when $Y_0 = \mathbf{0}$, we have $f(X_t) + \frac{1}{2}\|Y_t\|^2 = f(X_0)$, and hence $f(X_t) \leq f(X_0)$ for every $t \geq 0$. Therefore, outputting the endpoint of Hamiltonian flow is always a descent method. In particular, for short integration times of order $1/\sqrt{L}$ when f is L -smooth, endpoint guarantees can be obtained; see ([Fu and Wibisono, 2025](#), Lemma 6). However, such short-time endpoint

guarantees do not yield acceleration: in [Lemma 13](#), we show that with a fixed short integration time $T = c/\sqrt{L}$, the total integration time required to find an ε -close point with [\(HFopt\)](#) scales as $\Omega((\sqrt{L}/\alpha) \log(1/\varepsilon))$ in the worst case. Our key observation is that when f is convex, an *averaged* iterate over a suitably long trajectory can yield a stronger descent guarantee than that implied by the endpoint of Hamiltonian flow run from $Y_0 = \mathbf{0}$. We formalize this key property of the Hamiltonian flow trajectory in the following lemma, which to the best of our knowledge has not been derived previously.

Lemma 2 *Assume f satisfies [A_{cvx}](#). For any $t > 0$, initial position $X_0 \in \mathbb{R}^d$, and any reference position $z \in \mathbb{R}^d$, the Hamiltonian flow trajectory $(\text{HF}((X_0, \mathbf{0}); \tau))_{\tau \in [0, t]}$ satisfies*

$$f(X_t) - f(z) \leq \frac{2}{3}(f(X_0) - f(z)) - \frac{1}{6} \frac{d^2}{dt^2} \|X_t - z\|^2. \quad (1)$$

Proof Consider $h(t) = \langle X_t - z, Y_t \rangle = \frac{1}{2} \frac{d}{dt} \|X_t - z\|^2$. We have

$$\begin{aligned} \frac{1}{2} \frac{d^2}{dt^2} \|X_t - z\|^2 &= \dot{h}(t) \stackrel{(a)}{=} \|Y_t\|^2 - \langle X_t - z, \nabla f(X_t) \rangle \\ &\stackrel{(b)}{=} 2(f(X_0) - f(X_t)) - \langle X_t - z, \nabla f(X_t) \rangle \\ &\stackrel{(c)}{\leq} 2(f(X_0) - f(X_t)) - (f(X_t) - f(z)) \\ &= -3(f(X_t) - f(z)) + 2(f(X_0) - f(z)). \end{aligned}$$

Step (a) uses the Hamiltonian flow $\dot{X}_t = Y_t, \dot{Y}_t = -\nabla f(X_t)$ and the chain rule; step (b) uses the fact that the Hamiltonian is conserved for any $t \geq 0$; and step (c) uses the convexity of f . \blacksquare

For any $T > 0$, given the Hamiltonian flow trajectory $(X_t, Y_t) = \text{HF}((X_0, \mathbf{0}); t)$ for $0 \leq t \leq T$, we define the *weighted average*:

$$X^{\text{avg}}(X_0; T) = \frac{2}{T^2} \int_0^T (T-t) X_t dt. \quad (2)$$

An important corollary of [Lemma 2](#) is stated below, which we prove in [Appendix E.2.1](#).

Corollary 1 *Consider the setting of [Lemma 2](#). The weighted average $X^{\text{avg}}(X_0; T)$ satisfies*

$$f(X^{\text{avg}}(X_0; T)) - f(z) + \frac{1}{3T^2} \|X_T - z\|^2 \leq \frac{2}{3}(f(X_0) - f(z)) + \frac{1}{3T^2} \|X_0 - z\|^2. \quad (3)$$

[Corollary 1](#) implies that optimality gap of the time-average of the solution to the Hamiltonian flow can be controlled with respect to the optimality gap at the initialization, with a *universal* contraction factor $\frac{2}{3}$. This key property allows us to leverage the Hamiltonian flow as a subroutine to design *deterministic* algorithms for convex optimization with accelerated rates. Importantly, when $z \leftarrow x^*$ and $T \geq \sqrt{\frac{2\|X_0 - x^*\|^2}{f(X_0) - f(x^*)}}$, from the bound above we obtain

$$f(X^{\text{avg}}(X_0; T)) - f(x^*) \leq \frac{5}{6}(f(X_0) - f(x^*)).$$

Remark 1 A corollary of [Lemma 2](#) that we show en route proving [Corollary 1](#) is

$$\frac{2}{T^2} \int_0^T (T-t)(f(X_t) - f(z))dt + \frac{1}{3T^2} \|X_T - z\|^2 \leq \frac{2}{3}(f(X_0) - f(z)) + \frac{1}{3T^2} \|X_0 - z\|^2. \quad (4)$$

Suppose t is distributed according to density $p_t(t) = \frac{2}{T^2}(T-t)$ over $[0, T]$. Then, [Eq. \(4\)](#) implies

$$\mathbb{E}_{t \sim p_t} [f(X_t) - f(z)] + \frac{1}{3T^2} \|X_T - z\|^2 \leq \frac{2}{3}(f(X_0) - f(z)) + \frac{1}{3T^2} \|X_0 - z\|^2.$$

When $T \geq \sqrt{\frac{2\|X_0 - x^*\|^2}{f(X_0) - f(x^*)}}$, the above inequality implies that $f(X_t) - f(x^*) \leq \frac{5}{6}(f(X_0) - f(x^*))$ in expectation, taken over $t \sim p_t$. This novel randomized scheme is fundamentally different from the randomized Hamiltonian flow scheme studied in [Fu and Wibisono \(2025\)](#).

Additionally, [Corollary 1](#) can be generalized to different weighting schemes like the simple average; we elaborate on this in [Appendix E.2.2](#).

3.2. Provable minimization of α -strongly convex f

We begin by stating an implication of [Corollary 1](#) for functions satisfying α -quadratic growth.

Lemma 3 Assume f satisfies A_{cvx} and $A_{\alpha\text{-qg}}$. For any $T > 0$, the weighted average $X^{\text{avg}}(X_0; T)$ defined in [Eq. \(2\)](#) satisfies

$$f(X^{\text{avg}}(X_0; T)) - f(x^*) \leq \frac{2}{3} \left(1 + \frac{1}{\alpha T^2}\right) (f(X_0) - f(x^*)).$$

This is a direct consequence of the quadratic growth condition of f ; the complete proof is given in [Appendix E.2.4](#). Algorithmically, this lemma conveys that when given access to the Hamiltonian flow trajectory, aggregating the position according to the average $X^{\text{avg}}(X_0; T)$ for sufficiently long time results in an exponential contraction of the optimality gap. This suggests the algorithm [HFA](#) presented in [Algorithm 1](#) will converge exponentially fast; we present the formal guarantee in the following theorem. We note for a strongly convex objective function, we do not need any additional averaging (i.e., we set $\lambda = 0$ in [Algorithm 1](#)).

Theorem 1 Assume that f satisfies A_{cvx} and $A_{\alpha\text{-qg}}$. Let X_K be the output of [HFA \(Algorithm 1\)](#) for K iterations with parameter $\lambda = 0$ and the integration time sequence $T_k = T \geq \frac{C}{\sqrt{\alpha}}$ for all $k \in [K]$, for some constant $C > 0$. Then,

$$f(X_K) - f(x^*) \leq \left(\frac{2}{3} + \frac{2}{3C^2}\right)^K (f(X_0) - f(x^*)).$$

When $C > \sqrt{2}$, we have $\frac{2}{3} + \frac{2}{3C^2} < 1$. Moreover, when $C = 2$, the lower bound for T in [Theorem 1](#) implies, by the α -quadratic growth condition of f , the choice $T \geq \sqrt{\frac{4}{\alpha}} \geq \sqrt{\frac{2\|X_0 - x^*\|^2}{f(X_0) - f(x^*)}}$ discussed previously. From this theorem, we obtain a continuous-time convergence guarantee: an ε -accurate point can be obtained with total integration time scaling as $\alpha^{-1/2} \log(1/\varepsilon)$.

Corollary 2 Assume that f satisfies \mathbf{A}_{cvx} and $\mathbf{A}_{\alpha\text{-qg}}$. To obtain X_K that is an ε -accurate minimizer, it suffices to run HFA with parameter $\lambda = 0$ for K iterations and constant integration times $\{T_k\}_{k=1}^K$ where

$$K = \left\lceil 4 \log \frac{f(X_0) - f(x^*)}{\varepsilon} \right\rceil ; \quad T_k = \frac{5}{2\sqrt{\alpha}} \quad \forall k \in [K].$$

The total integration time is $T_{\text{tot}} = \frac{5}{2\sqrt{\alpha}} \left\lceil 4 \log \frac{f(X_0) - f(x^*)}{\varepsilon} \right\rceil$.

The proofs of the above theorem and corollary are stated in [Appendix E.2.3](#).

3.3. Provable minimization of convex f

We revisit [Eq. \(3\)](#) and the choice $\sqrt{\frac{2\|X_0 - x^*\|^2}{f(X_0) - f(x^*)}}$ which is sufficient for a contraction in the optimality gap. When f is convex, this ratio can be arbitrarily large, and therefore optimizing f using the averaged iterate can appear futile even when using exact Hamiltonian flow. We note that this lower bound for T stems from disregarding the $\frac{1}{3T^2}\|X_T - x^*\|^2$ term in [Eq. \(3\)](#). On the other hand, by including this term, it is unclear if there exists some \hat{X} that implies a decrease in optimality gap; this is due to the mismatch between $X^{\text{avg}}(X_0; T)$ passed to f and the X_T in the squared norm term. Interestingly, by using an additional averaging with the last iterate, we derive a version of [Eq. \(3\)](#) that explicitly defines a \hat{X} that leads to actionable algorithms in the convex setting.

Lemma 4 Let f satisfy \mathbf{A}_{cvx} . For any initial position $X_0 \in \mathbb{R}^d$, time $T > 0$, and parameter $\lambda > 0$, define

$$X^{\text{mix}}(X_0; T) = \frac{\lambda}{\lambda + 1} X_T + \frac{1}{1 + \lambda} X^{\text{avg}}(X_0; T).$$

Then, defining $C_\lambda = \frac{(3\lambda+1)^2}{9\lambda(1+\lambda)}$, we have

$$f(X^{\text{mix}}(X_0; T)) - f(x^*) + \frac{1}{3\lambda T^2} \|X^{\text{mix}}(X_0; T) - x^*\|^2 \leq C_\lambda (f(X_0) - f(x^*)) + \frac{1}{3\lambda T^2} \|X_0 - x^*\|^2.$$

Proof Sketch By the properties of the Hamiltonian flow,

$$f(X^{\text{avg}}(X_0; T)) - f(x^*) + \frac{1}{3T^2} \|X_T - x^*\|^2 \leq \frac{2}{3} (f(X_0) - f(x^*)) + \frac{1}{3T^2} \|X_0 - x^*\|^2 \quad (5a)$$

$$\frac{1}{3T^2} \|X^{\text{avg}}(X_0; T) - x^*\|^2 \leq \frac{1}{9} (f(X_0) - f(x^*)) + \frac{1}{3T^2} \|X_0 - x^*\|^2 \quad (5b)$$

$$f(X_T) - f(x^*) \leq f(X_0) - f(x^*). \quad (5c)$$

We simplify the weighted sum $\lambda \cdot \text{Eq. (5a)} + \text{Eq. (5b)} + \lambda^2 \cdot \text{Eq. (5c)}$, and use Jensen's inequality with the convexity of f and $\|\cdot\|^2$ to complete the proof. \blacksquare

We give the complete proof in [Appendix E.2.5](#). We highlight that $C_\lambda < 1$ for all $\lambda > \frac{1}{3}$; C_λ is minimized at $\lambda = 1$, where it takes the value $C_1 = \frac{8}{9}$; and X^{mix} for $\lambda = 1$ is a uniform average between X_T and $X^{\text{avg}}(X_0; T)$. The above lemma implies a convergence guarantee for minimizing convex functions using HFA, provided the integration times increase suitably at each iteration, and we output X^{mix} as the next iterate (which is the reason we introduce the parameter λ in HFA).

Theorem 2 Assume f satisfies \mathbf{A}_{cvx} . Suppose X_K is the output of HFA after K iterations with parameter $\lambda = 1$ and integration time sequence satisfying $T_0 = 1$ and $T_k^2 \geq \frac{9}{8}T_{k-1}^2$ for all $k \in [K]$. Then,

$$f(X_K) - f(x^*) \leq \left(\frac{8}{9}\right)^K \left(f(X_0) - f(x^*) + \frac{1}{3}\|X_0 - x^*\|^2 \right).$$

While the above theorem indicates that the optimality gap decays geometrically by the factor of $\frac{8}{9}$, the integration time increases geometrically at each iteration. However, since the *squared* integration time T_k^2 increases precisely by $\frac{9}{8}$, the inverse contraction factor, the total integration time to achieve a ε -accurate solution scales as $\frac{1}{\sqrt{\varepsilon}}$; this is presented in the following corollary, which is a counterpart to [Corollary 2](#) for convex f .

Corollary 3 Let f satisfy \mathbf{A}_{cvx} . To obtain X_K that is a ε -accurate minimizer, it suffices to run HFA with parameter $\lambda = 1$ for K iterations and integration time sequence $\{T_k\}_{k=1}^K$ such that

$$K = \left\lceil \frac{1}{\log(\frac{9}{8})} \cdot \log \frac{f(X_0) - f(x^*) + \frac{1}{3}\|X_0 - x^*\|^2}{\varepsilon} \right\rceil ; \quad T_k = \left(\frac{9}{8}\right)^{k/2}.$$

The total integration time is $T_{\text{tot}} \leq 18\sqrt{\frac{f(X_0) - f(x^*) + \frac{1}{3}\|X_0 - x^*\|^2}{\varepsilon}}$.

The proofs of the above theorem and corollary are stated in [Appendix E.2.6](#).

4. From exact Hamiltonian Flow to a discrete time method for minimizing convex f

The results of the prior section showcased the potential of an idealized iterative method based on the Hamiltonian flow for minimizing f at an accelerated rate while only having access to a gradient oracle of f and an exact solver for the Hamiltonian dynamics. In this section, we explore how to transfer the guarantees therein to an *implementable* algorithm via a practical discretization of the Hamiltonian flow, specifically using the *extragradient* integrator ([Fu and Wibisono, 2025](#), Sec. 4).

4.1. Properties of the extragradient integrator

A crucial property of the extragradient integrator that makes it a viable scheme for minimizing convex f is that, when the step size is sufficiently small, the Hamiltonian value *decreases* with every iteration; this is stated in the following lemma (see [Appendix F.1.1](#) for the proof).

Lemma 5 Let f satisfy \mathbf{A}_{cvx} and $\mathbf{A}_{L\text{-sm}}$. Choose $\eta \leq 1/\sqrt{L}$, and suppose (x_{n+1}, y_{n+1}) is produced by one step of the extragradient integrator (HF^{extg}) from (x_n, y_n) . Then

$$f(x_{n+1}) + \frac{1}{2}\|y_{n+1}\|^2 \leq f(x_n) + \frac{1}{2}\|y_n\|^2. \quad (6)$$

We contrast this with the property of the exact Hamiltonian flow along which the Hamiltonian value is *preserved*. Nonetheless, a consequence of this lemma is that running the extragradient integrator from $y_0 = \mathbf{0}$ and any position $x_0 \in \mathbb{R}^d$ results in a sequence of $\{x_n\}_{n \geq 0}$ such that $f(x_n) \leq f(x_0)$, mirroring what happens along the Hamiltonian flow. However, as in the continuous-time setting, this bound relative to the initial function value alone does not provide an explicit quantitative convergence rate. It is therefore natural to ask if the extragradient integrator satisfies a version of

Algorithm 2 Discretized Hamiltonian Flow for optimization with Averaging (dHFA-eg)

Input : Initialization: $x_0 \in \mathbb{R}^d$, Number of iterations: $K \in \mathbb{N}$, parameter: $\lambda \in \mathbb{R}_{\geq 0}$, step size: $\eta > 0$, discretization steps sequence: $\{N_1, \dots, N_K\} \in \mathbb{N}^K$

for $k = 1$ **to** K **do**

Obtain sequence $\{(x_n^{(k)}, y_n^{(k)})\}_{n=1}^{N_k}$ using **HF^{extg}** from $(x_0^{(k)}, y_0^{(k)}) = (x_{k-1}, \mathbf{0})$.

Compute $x^{\text{avg}}(x_{k-1}; N_k) = \frac{2}{N_k(N_k+1)} \sum_{n=1}^{N_k} (N_k - n + 1)x_n^{(k)}$.

Set $x_k = \frac{1}{\lambda+1}x^{\text{avg}}(x_{k-1}; N_k) + \frac{\lambda}{\lambda+1}x_{N_k}^{(k)}$.

end

return x_K

Corollary 1, and we answer this affirmatively. In particular, provided that the step size is sufficiently small, it is possible to control the optimality gap of an average of the iterates produced by the extragradient integrator. For $N > 0$, we define the weighted average

$$x^{\text{avg}}(x_0; N) = \frac{2}{N(N+1)} \sum_{n=1}^N (N-n+1)x_n.$$

Lemma 6 Assume f satisfies **A_{cvx}** and **A_{L-sm}**, and let $\{(x_n, y_n)\}_{n \geq 1}$ be the iterates of extragradient integrator (**HF^{extg}**) with step size $\eta \leq \frac{1}{\sqrt{L}}$, starting from $(x_0, \mathbf{0})$. Then, for any $N \geq 1$,

$$f(x^{\text{avg}}(x_0; N)) - f(x^*) + \frac{\|x_N - x^*\|^2}{3\eta^2 N(N+1)} \leq \frac{2}{3}(f(x_0) - f(x^*)) + \frac{\|x_0 - x^*\|^2}{3\eta^2 N(N+1)}. \quad (7)$$

We prove this lemma in Appendix F.1.2. The result above is motivated by the properties that we establish for the average iterate along the Hamiltonian flow trajectory in the exact Hamiltonian flow setting discussed in the prior section, which leads to the similarities between Eq. (7) and Eq. (3).

Lemma 6 provides the theoretical basis for the design of **dHFA-eg** (**Algorithm 2**) for minimizing convex and L -smooth functions. Contrary to **HFA** (which assumes we can simulate the exact Hamiltonian flow), **dHFA-eg** is implementable with access to a gradient oracle to f . From an implementation standpoint, obtaining the sequence $\{(x_n^{(k)}, y_n^{(k)})\}_{n=1}^{N_k}$ for every k would be performed in an inner loop. The computation of $x^{\text{avg}}(x_{k-1}; N_k)$ can be incorporated as part of this loop wherein it is computed as a running weighted sum (analogous to a streaming sum), and therefore does not require storing the entire sequence in memory for its computation; the presentation in **Algorithm 2** is intended for readability. It is also possible to replace **HF^{extg}** with **HF^{imp}**, which requires access to a proximal oracle for f (see details in Appendix F.2); essentially, this substitution results in quantitatively similar iteration complexities as using **HF^{extg}** but without assuming smoothness of f .

4.2. Provable minimization of α -strongly convex f

Theorem 3 Assume that f satisfies **A_{cvx}**, **A_{L-sm}** and **A _{α -qg}**. Suppose x_K is the output of running **dHFA-eg** for K iterations from initialization $x_0 \in \mathbb{R}^d$, with parameter $\lambda = 0$, step size $0 < \eta \leq \frac{1}{\sqrt{L}}$, and discretization steps sequence satisfying $N_k = N \geq \lceil \frac{c}{\eta\sqrt{\alpha}} \rceil$ for all $k \in [K]$, for any $c > 0$. Then,

$$f(x_K) - f(x^*) \leq \left(\frac{2}{3} + \frac{2}{3c^2} \right)^K (f(x_0) - f(x^*)).$$

From this, we obtain the following corollary on the iteration complexity of **dHFA-eg** for minimizing convex functions that satisfy both α -quadratic growth and L -smoothness. The proofs of the previous theorem and the following corollary are given in [Appendix F.1.3](#).

Corollary 4 *Assume that f satisfies \mathbf{A}_{cvx} , $\mathbf{A}_{L\text{-sm}}$ and $\mathbf{A}_{\alpha\text{-qg}}$. To obtain x_K that is an ε -accurate minimizer, it suffices to run **dHFA-eg** with parameter $\lambda = 0$ and step size $\eta = \frac{1}{\sqrt{L}}$ for K iterations and discretization step sequence $\{N_k\}_{k=1}^K$ such that*

$$K = \left\lceil 4 \log \frac{f(x_0) - f(x^*)}{\varepsilon} \right\rceil ; \quad N_k = \left\lceil \frac{5}{2} \sqrt{k} \right\rceil \quad \forall k \in [K].$$

The total number of gradient evaluations is $N_{\text{tot}} = 2 \cdot \left\lceil \frac{5}{2} \sqrt{K} \right\rceil \cdot \left\lceil 4 \log \frac{f(x_0) - f(x^*)}{\varepsilon} \right\rceil$.

4.3. Provable minimization of convex f

Owing to the similarity between [Eq. \(3\)](#) and [Eq. \(7\)](#), one can naturally expect that a combination of $x^{\text{avg}}(x_0; N)$ and x_N leads to a viable strategy to minimizing f . In the following lemma, we show that this is indeed the case, and this explains the necessity of the parameter λ in **dHFA-eg**. We provide the proof of this lemma in [Appendix F.1.4](#).

Lemma 7 *Assume that f satisfies \mathbf{A}_{cvx} and $\mathbf{A}_{L\text{-sm}}$. Let $\{(x_n, y_n)\}_{n \geq 1}$ be generated by the extra-gradient integrator with step size $\eta \leq \frac{1}{\sqrt{L}}$. For any $N \geq 4$ and initial position x_0 , define*

$$x^{\text{mix}}(x_0; N) = \frac{\lambda}{\lambda + 1} x_N + \frac{1}{1 + \lambda} x^{\text{avg}}(x_0; N).$$

Then, defining $c_\lambda = \frac{(6\lambda^2 + 4\lambda + 1)}{6\lambda(\lambda + 1)}$, we have

$$f(x^{\text{mix}}(x_0; N)) - f(x^*) + \frac{\|x^{\text{mix}}(x_0; N) - x^*\|^2}{3\lambda\eta^2 N(N+1)} \leq c_\lambda (f(x_0) - f(x^*)) + \frac{\|x_0 - x^*\|^2}{3\lambda\eta^2 N(N+1)}.$$

The factor c_λ is strictly less than 1 for $\lambda > \frac{1}{2}$, is minimized at $\lambda^* = \frac{\sqrt{3}+1}{2}$ (where it equals $c_{\lambda^*} = \frac{\sqrt{3}+1}{3}$), and this combination biases $x^{\text{mix}}(x_0; N)$ toward x_N . The discrepancy between C_λ in [Lemma 4](#) (for continuous time) and c_λ in the above lemma is due to a discrete sum appearing in the analysis instead of an integral.

Theorem 4 *Assume that f satisfies \mathbf{A}_{cvx} and $\mathbf{A}_{L\text{-sm}}$. Suppose x_K is the output of running **dHFA-eg** for K iterations with parameter $\lambda = \frac{\sqrt{3}+1}{2}$, step size $0 < \eta \leq \frac{1}{\sqrt{L}}$, and discretization steps sequence $\{N_k\}_{k=1}^K$ satisfying $N_0 = 4$ and $N_k(N_k + 1) \geq \frac{3}{\sqrt{3}+1} N_{k-1}(N_{k-1} + 1)$ for all $k \in [K]$. Then,*

$$f(x_K) - f(x^*) \leq \left(\frac{\sqrt{3} + 1}{3} \right)^K \left(f(x_0) - f(x^*) + \frac{\sqrt{3} - 1}{60\eta^2} \|x_0 - x^*\|^2 \right).$$

Drawing a comparison to [Theorem 2](#), we see a geometric rate of decrease over iterations K , while the number of discretization steps per iteration increases exponentially. However, due to the fact that the *squared* discretization steps increases by the reciprocal of the factor that the optimality gap decreases by at each iteration, we obtain a $\frac{1}{\sqrt{\varepsilon}}$ scaling in the total gradient complexity to obtain an ε -accurate solution. This is stated in the following corollary. The proofs of the previous theorem and the following corollary are given in [Appendix F.1.5](#).

Corollary 5 Let f satisfy A_{cvx} and $A_{L\text{-sm}}$. To obtain x_K that is an ε -accurate minimizer, it suffices to run **dHFA-eg** with parameter $\lambda = \frac{\sqrt{3}+1}{2}$ and step size $\eta = \frac{1}{\sqrt{L}}$ for K iterations and discretization step sequence $\{N_k\}_{k=1}^K$ such that

$$K = \left\lceil 2 \left(\log \frac{3}{1 + \sqrt{3}} \right)^{-1} \cdot \log \left(\sqrt{\frac{L}{\varepsilon}} \|x_0 - x^*\| \right) \right\rceil; \quad N_k = \left\lceil \sqrt{\frac{3}{1 + \sqrt{3}}} N_{k-1} + \frac{1}{2} \right\rceil; \quad N_0 = 4.$$

The total number of gradient evaluations satisfies $N_{\text{tot}} \leq 1620 \sqrt{\frac{L}{\varepsilon}} \|x_0 - x^*\|$.

5. Conclusion

In this work, we provide a new perspective on how the Hamiltonian flow can be used for convex optimization. Through a novel differential inequality along Hamiltonian flow for convex objective ([Lemma 2](#)), we show that a suitably chosen aggregate of the Hamiltonian flow trajectory is well-suited for optimization purposes, and leads to the idealized algorithm **HFA**. To make this practical in settings where **HF** cannot be simulated exactly, we propose a discretized version based on the extragradient integrator in **dHFA-eg**, which only requires access to the gradients of f . We prove **dHFA-eg** achieves accelerated convergence rates for smooth convex and strongly convex optimization.

Our results spur many interesting future directions. We note that our analysis predicts the same qualitative rate for different aggregation schemes such as the uniform average or the minimizer along the trajectory. We conjecture that choosing the next iterate to be the minimizer along the trajectory converges at a strictly better rate than other aggregation schemes. While we give guarantees for minimizing convex functions, it would be interesting to study whether **HFA** or **dHFA-eg** can provably optimize other classes of functions, such as those satisfying a Polyak-Łojaciewicz inequality, or even non-convex functions. In this work, we consider two integrators that dissipate the Hamiltonian energy, namely implicit integrator and extragradient integrator. However, our experiments suggest that the leapfrog integrator exhibits a convergence behavior similar to the implicit and extragradient integrators; it would be interesting to study this theoretically. Finally, while the results of this paper concern optimization, it would be interesting to see if the broader techniques developed in this work can be translated to developed HMC-based algorithms that perform *accelerated sampling*.

Acknowledgments. Qiang Fu, Siddharth Mitra, Xiuyuan Wang, and Andre Wibisono were supported by NSF awards CCF-2403391 and CAREER CCF-2443097. Vishwak Srinivasan and Ashia Wilson were supported in part by Assicurazioni Generali S.p.A. through MIT Award 036189-00006.

References

- Zeyuan Allen-Zhu and Lorenzo Orecchia. Linear Coupling: An Ultimate Unification of Gradient and Mirror Descent. In *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 3:1–3:22. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017.
- Jason M Altschuler and Pablo A Parrilo. Acceleration by stepsize hedging: Multi-step descent and the silver stepsize schedule. *Journal of the ACM*, 72(2):1–38, 2025a.
- Jason M Altschuler and Pablo A Parrilo. Acceleration by stepsize hedging: Silver stepsize schedule for smooth convex optimization. *Mathematical Programming*, 213(1):1105–1118, 2025b.
- Hedy Attouch, Zaki Chbani, Jalal Fadili, and Hassan Riahi. First-order optimization algorithms via inertial systems with Hessian driven damping. *Mathematical Programming*, pages 1–43, 2022.
- Michael Betancourt, Michael I. Jordan, and Ashia C. Wilson. On Symplectic Optimization. *arXiv preprint arXiv:1802.03653*, 2026.
- Nawaf Bou-Rabee and Andreas Eberle. Mixing time guarantees for unadjusted Hamiltonian Monte Carlo. *Bernoulli*, 29(1):75–104, 2023.
- Nawaf Bou-Rabee and J. M. Sanz-Serna. Geometric integrators and the Hamiltonian Monte Carlo method. *Acta Numerica*, 27:113–206, 2018. ISSN 1474-0508.
- Nawaf Bou-Rabee, Siddharth Mitra, and Andre Wibisono. Tail-Sensitive KL and Rényi Convergence of Unadjusted Hamiltonian Monte Carlo via One-Shot Couplings. *arXiv preprint arXiv:2601.09019*, 2026.
- Sébastien Bubeck, Yin Tat Lee, and Mohit Singh. A geometric alternative to Nesterov’s accelerated gradient descent. *arXiv preprint arXiv:1506.08187*, 2015.
- Bob Carpenter, Andrew Gelman, Matthew D Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of statistical software*, 76:1–32, 2017.
- Shi Chen, Qin Li, Oliver Tse, and Stephen J. Wright. Accelerating optimization over the space of probability measures. *Journal of Machine Learning Research*, 26(31):1–40, 2025.
- Arnak Dalalyan. Further and stronger analogy between sampling and optimization: Langevin Monte Carlo and gradient descent. In *Conference on Learning Theory*, pages 678–689. PMLR, 2017a.
- Arnak S. Dalalyan. Theoretical guarantees for approximate sampling from smooth and log-concave densities. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(3): 651–676, 2017b.
- G Bruno De Luca, Alice Gatti, and Eva Silverstein. Improving energy conserving descent for machine learning: Theory and practice. *arXiv preprint arXiv:2306.00352*, 2023.
- Jelena Diakonikolas and Michael I Jordan. Generalized momentum-based methods: A Hamiltonian perspective. *SIAM Journal on Optimization*, 31(1):915–944, 2021.

- Yoel Drori and Marc Teboulle. Performance of first-order methods for smooth convex minimization: a novel approach. *Mathematical Programming*, 145(1):451–482, 2014.
- Dmitriy Drusvyatskiy, Maryam Fazel, and Scott Roy. An optimal first order method based on optimal quadratic averaging. *SIAM Journal on Optimization*, 28(1):251–271, 2018.
- Simon Duane, Anthony D Kennedy, Brian J Pendleton, and Duncan Roweth. Hybrid Monte Carlo. *Physics letters B*, 195(2):216–222, 1987.
- Alain Durmus, Szymon Majewski, and Błażej Miasojedow. Analysis of Langevin Monte Carlo via convex optimization. *Journal of Machine Learning Research*, 20(73):1–46, 2019.
- Valentin Duruisseaux and Melvin Leok. Practical perspectives on symplectic accelerated optimization. *Optimization Methods and Software*, 38(6):1230–1268, 2023.
- Alexandre d’Aspremont, Damien Scieur, and Adrien Taylor. Acceleration methods. *Foundations and Trends in Optimization*, 5(1-2):1–245, 2021.
- Mathieu Even, Raphaël Berthier, Francis Bach, Nicolas Flammarion, Hadrien Hendrikx, Pierre Gaillard, Laurent Massoulié, and Adrien Taylor. Continuiized accelerations of deterministic and stochastic gradient descents, and of gossip algorithms. In *Advances in Neural Information Processing Systems*, volume 34, pages 28054–28066. Curran Associates, Inc., 2021.
- Qiang Fu and Andre Wibisono. Hamiltonian descent algorithms for optimization: Accelerated rates via randomized integration time. In *Advances in Neural Information Processing Systems*, volume 38, pages 93628–93685. Curran Associates, Inc., 2025.
- Benjamin Grimmer. Provably faster gradient descent via long steps. *SIAM Journal on Optimization*, 34(3):2588–2608, 2024.
- Ernst Hairer, Marlis Hochbruck, Arieh Iserles, and Christian Lubich. Geometric numerical integration. *Oberwolfach Reports*, 3(1):805–882, 2006.
- Matthew D Hoffman and Andrew Gelman. The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623, 2014.
- Bin Hu and Laurent Lessard. Dissipativity theory for Nesterov’s accelerated method. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 1549–1557. PMLR, 2017.
- Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In Paolo Frasconi, Niels Landwehr, Giuseppe Manco, and Jilles Vreeken, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 795–811, Cham, 2016. Springer International Publishing.
- Benedict Leimkuhler and Sebastian Reich. *Simulating Hamiltonian Dynamics*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2005.

- Wu Lin, Valentin Duruisseaux, Melvin Leok, Frank Nielsen, Mohammad Emtiyaz Khan, and Mark Schmidt. Simplifying momentum-based positive-definite submanifold optimization with applications to deep learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 21026–21050. PMLR, 2023.
- Yi-An Ma, Niladri S. Chatterji, Xiang Cheng, Nicolas Flammarion, Peter L. Bartlett, and Michael I. Jordan. Is there an analog of Nesterov acceleration for gradient-based MCMC? *Bernoulli*, 27(3): 1942 – 1992, 2021.
- Chris J Maddison, Daniel Paulin, Yee Whye Teh, Brendan O’Donoghue, and Arnaud Doucet. Hamiltonian descent methods. *arXiv preprint arXiv:1809.05042*, 2018.
- Michael Muehlebach and Michael Jordan. A dynamical systems perspective on Nesterov acceleration. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 4656–4662. PMLR, 2019.
- Radford M Neal. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11):2, 2011.
- Yurii Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. In *Dokl akad nauk Sssr*, volume 269, page 543, 1983.
- Yurii Nesterov. *Lectures on convex optimization*, volume 137 of *Springer Optimization and Its Applications*. Springer, Cham, second edition, 2018.
- Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and trends® in Optimization*, 1(3):127–239, 2014.
- Adil Salim, Dmitry Kovalev, and Peter Richtarik. Stochastic proximal langevin algorithm: Potential splitting and nonasymptotic rates. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.
- A. Scagliotti and P. Colli Franzone. A piecewise conservative method for unconstrained convex optimization. *Computational Optimization and Applications*, 81(1):251–288, 2021. ISSN 1573-2894.
- Bin Shi, Simon S Du, Michael I Jordan, and Weijie J Su. Understanding the acceleration phenomenon via high-resolution differential equations. *Mathematical Programming*, pages 1–70, 2022.
- Vishwak Srinivasan, Andre Wibisono, and Ashia Wilson. High-accuracy sampling from constrained spaces with the metropolis-adjusted preconditioned langevin algorithm. In *Proceedings of The 36th International Conference on Algorithmic Learning Theory*, volume 272, pages 1169–1220. PMLR, 2025.
- Weijie Su, Stephen Boyd, and Emmanuel J. Candès. A Differential Equation for Modeling Nesterov’s Accelerated Gradient Method: Theory and Insights. *Journal of Machine Learning Research*, 17(153):1–43, 2016.

- Andrew R Teel, Jorge I Poveda, and Justin Le. First-order optimization algorithms with resets and Hamiltonian flows. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 5838–5843. IEEE, 2019.
- Brian Kha Tran and Melvin Leok. Geometric methods for adjoint systems. *Journal of Nonlinear Science*, 34(1), 2023. ISSN 1432-1467.
- Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature methods*, 17(3):261–272, 2020.
- Jun-Kun Wang. Frictionless Hamiltonian Descent and Coordinate Hamiltonian Descent for Strongly Convex Quadratic Problems and Beyond. *arXiv preprint arXiv:2402.13988*, 2025.
- Andre Wibisono. Sampling as optimization in the space of measures: The langevin dynamics as a composite optimization problem. In *Proceedings of the 31st Conference On Learning Theory*, volume 75, pages 2093–3027. PMLR, 2018.
- Andre Wibisono, Ashia C Wilson, and Michael I Jordan. A variational perspective on accelerated methods in optimization. *Proceedings of the National Academy of Sciences*, 113(47):E7351–E7358, 2016.
- Andre Wibisono, Molei Tao, and Georgios Piliouras. Alternating mirror descent for constrained min-max games. In *Advances in Neural Information Processing Systems*, volume 35, pages 35201–35212. Curran Associates, Inc., 2022.
- Ashia Wilson, Lester Mackey, and Andre Wibisono. Accelerating rescaled gradient descent: Fast optimization of smooth functions. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.
- Ashia C Wilson, Ben Recht, and Michael I Jordan. A Lyapunov analysis of accelerated methods in optimization. *Journal of Machine Learning Research*, 22(113):1–34, 2021.
- Hui Zhang and Wotao Yin. Gradient methods for convex minimization: better rates under weaker conditions. *arXiv preprint arXiv:1303.4645*, 2013.
- Kelvin Shuangjian Zhang, Gabriel Peyré, Jalal Fadili, and Marcelo Pereyra. Wasserstein Control of Mirror Langevin Monte Carlo. In *Proceedings of Thirty Third Conference on Learning Theory*, volume 125, pages 3814–3841. PMLR, 2020.
- Heng Zhu and Jun-Kun Wang. Frictionless hamiltonian descent with discretization and parallel optimization. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856.

Appendix

Table of Contents

A	Additional Related Works	20
B	Background on Hamiltonian Dynamics	22
B.1	Properties of the Hamiltonian flow	22
B.2	Discussion of integrators	24
B.2.1	Explicit and implicit integrators	24
B.2.2	Leapfrog integrator	26
C	A Review of Hamiltonian Monte Carlo for Sampling	27
D	Lower Bound for HFopt	28
E	Extended discussion of Section 3	29
E.1	Examples	29
E.1.1	A Quadratic Setting	29
E.1.2	A Strictly Convex Setting	30
E.2	Proofs of main lemmas in Section 3	31
E.2.1	Proof of Corollary 1	31
E.2.2	Versions of Lemma 2 for the average and minimizer over the trajectory	33
E.2.3	Proof of Theorem 1 and Corollary 2	34
E.2.4	Proof of Lemma 3	35
E.2.5	Proof of Lemma 4	35
E.2.6	Proof of Theorem 2 and Corollary 3	36
F	Extended discussion of Section 4	37

F.1	Proofs of the statements in Section 4	37
F.1.1	Proof of Lemma 5	37
F.1.2	Proof of Lemma 6	37
F.1.3	Proofs of Theorem 3 and corollary 4	40
F.1.4	Proof of Lemma 7	40
F.1.5	Proofs of Theorem 4 and Corollary 5	42
F.1.6	Proof of Lemma 15	43
F.2	Discretized Hamiltonian Flow for optimization with Averaging using HF^{imp}	44
F.2.1	Proof of Lemma 16	47
F.2.2	Proof of Lemma 17	47
F.2.3	Proof of Lemma 18	48
G	Numerical Experiments	49
G.1	Comparison with gradient descent and accelerated gradient descent	49
G.2	Alternative averaging for iterates	50
G.3	Varying the integrator in dHFA-eg	50

Appendix A. Additional Related Works

Accelerated gradient flow and accelerated gradient descent. The accelerated gradient method of [Nesterov \(1983\)](#) is the following update:

$$\begin{cases} x_{k+1} = y_k - \eta \nabla f(y_k), \\ y_{k+1} = x_{k+1} + \gamma_k (x_{k+1} - x_k). \end{cases} \quad (\text{AGD})$$

The dynamical systems perspective of [AGD](#) presented in [Su et al. \(2016\)](#) spawned many follow-up works pertaining to algorithm design and generalizations of the techniques. [Wibisono et al. \(2016\)](#) develop a variational perspective of accelerated methods in optimization via the principle of least action using the Bregman Lagrangian, which recover known accelerated methods and derive new and faster accelerated methods beyond [AGD](#), as well as the discrete-time algorithms with matching convergence rates. Additional papers that adopt the dynamical perspective of acceleration include [Hu and Lessard \(2017\)](#); [Maddison et al. \(2018\)](#); [Muehlebach and Jordan \(2019\)](#); [Even et al. \(2021\)](#); [Attouch et al. \(2022\)](#); [Shi et al. \(2022\)](#). This continues to be an active area of research.

Hamiltonian dynamics for optimization. Taking this dynamical perspective further, the *Hamiltonian flow* ([HF](#)) corresponds to an undamped second-order system that conserves total energy and therefore does not converge to a minimizer in continuous time. While there have been many works that study *damped* Hamiltonian systems (such as those derived from accelerated gradient methods), to the best of our knowledge, only a small number of prior works have investigated optimization algorithms based on *conservative* Hamiltonian dynamics, as we study in this paper.

As mentioned in [Section 1](#), [Teel et al. \(2019\)](#) propose [HFopt](#), and in their scheme, the velocity is reset to zero either when the trajectory approaches the boundary of the region $\{(x, y) \in \mathbb{R}^d \times \mathbb{R}^d : \langle \nabla f(x), y \rangle \leq 0, \|y\|^2 \geq \|\nabla f(x)\|^2 / L\}$ where L is the smoothness constant, or when a fixed timer expires; see the pseudocode below. They establish uniform global stability and (non-accelerated) convergence guarantees for minimizing smooth and strongly convex functions, albeit only in the idealized setting assuming we can simulate the continuous-time Hamiltonian flow.

Algorithm 3 Hybrid Hamiltonian Algorithm with Resets ([Teel et al. \(2019\)](#))

Input : Initialization: $X_0 \in \mathbb{R}^d$, number of iterations: $K \in \mathbb{N}$, maximum integration time: $T \in (0, \infty)$, threshold parameter: $\bar{L} \geq 0$

for $k = 1$ **to** K **do**

Solve HF from $(X_{k-1}, \mathbf{0})$ for time T to obtain trajectory $((X_t^{(k)}, Y_t^{(k)}))_{t \in [0, T]}$.
Compute $T_k = \inf \left\{ t \in [0, T] : \langle \nabla f(X_t^{(k)}), Y_t^{(k)} \rangle = 0 \text{ and } \ Y_t^{(k)}\ ^2 \geq \frac{\ \nabla f(X_t^{(k)})\ ^2}{L} \right\}$, with the convention that $T_k = T$ if this set is empty.
Set $X_k = X_{T_k}^{(k)}$.

end

return X_K

[Scagliotti and Colli Franzone \(2021\)](#) propose a different restart rule for the same conservative dynamics ([HFopt](#)). Starting from $(X_{k-1}, \mathbf{0})$, they evolve the Hamiltonian flow until the *mean dissipation* $t \mapsto \frac{\|Y_t^{(k)}\|^2}{2t}$ attains a local maximum. Along the Hamiltonian flow, this stopping rule

can be written as

$$T_k = \inf \left\{ t > 0 : -t \langle \nabla f(X_t^{(k)}), Y_t^{(k)} \rangle - \frac{1}{2} \|Y_t^{(k)}\|^2 < 0 \right\},$$

see the pseudocode below. They proved that this restart time is finite for coercive objectives, and uniformly bounded for α -strongly convex and L -smooth f . As a consequence, the resulting piecewise conservative trajectory converges linearly:

$$f(\tilde{X}(t)) - f(x^*) \leq \left(1 + \frac{\alpha}{L}\right)^{-\lfloor t/T_R \rfloor} (f(X_0) - f(x^*)), \quad T_R = \frac{32L}{\alpha\sqrt{\alpha}},$$

and in particular

$$f(X_k) - f(x^*) \leq \left(1 + \frac{\alpha}{L}\right)^{-k} (f(X_0) - f(x^*)).$$

Algorithm 4 Hamiltonian Method of Scagliotti and Colli Franzone (2021)

Input : Initialization: $X_0 \in \mathbb{R}^d$, number of iterations: $K \in \mathbb{N}$

for $k = 1$ **to** K **do**

Solve (HF) from $(X_{k-1}, \mathbf{0})$ until the first time $T_k = \inf \left\{ t > 0 : \frac{d}{dt} \left(\frac{\|Y_t^{(k)}\|^2}{2t} \right) < 0 \right\}$.

Set $X_k = X_{T_k}^{(k)}$.

end

return X_K

Diakonikolas and Jordan (2021) analyze a class of *generalized* Hamiltonian dynamics driven by a time-dependent Hamiltonian $H(x, y, \tau) = h(\tau) f(x/\tau) + \psi^*(y)$, where $h(\tau) > 0$ is a function of the scaled time τ , and ψ^* is a strongly convex and differentiable function. They show that along the Hamiltonian flow, the averaged gradient $\left\| \frac{1}{t} \int_0^t \nabla f(x_\tau) d\tau \right\|$ decays at an t^{-1} rate. Moreover, they demonstrate that a broad family of momentum methods in both Euclidean and non-Euclidean geometries can be derived from these generalized dynamics, encompassing classical algorithms such as AGD and other accelerated methods (Wibisono et al., 2016; Wilson et al., 2021). These generalized constructions are conceptually distinct from the energy-conserving Hamiltonian dynamics studied in this work, where we use the standard Hamiltonian function which is time-independent with a quadratic kinetic energy.

De Luca et al. (2023) propose a Hamiltonian-based method with Hamiltonian function $H(x, y) = \lambda \log(F(x) - F_0) + \log(\|y\|^2)$, where λ, F_0 are user-specified parameters. They perform an empirical evaluation of this algorithm, but do not provide a convergence rate guarantee.

Accelerated optimization via Hamiltonian dynamics. A recent work by Wang (2025) studies HFopt for minimizing strongly convex quadratics $f(x) = \frac{1}{2}x^\top Ax - b^\top x$ with $\alpha I \preceq A \preceq LI$. In this quadratic setting, the Hamiltonian dynamics HF admit a closed-form solution, which enables a sharp analysis. By selecting a sequence of integration times $\{T_k\}$ according to the roots of Chebyshev polynomials, Wang (2025) proves that HFopt converges to the minimizer of f at a rate of $\exp(-k/\sqrt{\kappa})$ after k iterations, with a total integration time to achieve an ε -accurate solution that scales as $1/\sqrt{\alpha}$, matching accelerated time complexity in Table 1. A more systematic investigation of HF as a foundation for first-order optimization was conducted by Fu and Wibisono (2025),

wherein they study different integration schedules and discretization schemes, both for the general convex and strongly convex objectives. In particular, they show that HF with a short integration time recovers the convergence rate of gradient descent. Moreover, by randomizing the integration time, the resulting dynamics exhibit acceleration convergence rates analogous to the accelerated gradient flow. The corresponding discretized algorithm, termed randomized Hamiltonian gradient descent (RHGD), achieves accelerated convergence rates that match the rates of AGD. These results demonstrate that both non-accelerated and accelerated first-order methods can be viewed as arising from Hamiltonian dynamics under different integration and discretization schemes. Whereas the work of Fu and Wibisono (2025) provides accelerated rates for randomized integration times, so the guarantees only hold in expectation or in high probability, in this work we provide deterministic accelerated rates via long integration times.

Appendix B. Background on Hamiltonian Dynamics

We present an extended discussion about Hamiltonian dynamics and numerical integrators. In Section 2.1, we restricted our attention to the energy function $H(x, y) = f(x) + \frac{1}{2}\|y\|^2$ as it was more pertinent to optimization. However, the Hamiltonian can be defined more generally over the phase space \mathbb{R}^{2d} . A key definition in the Hamiltonian dynamics is the *symplectic matrix* $\Omega \in \mathbb{R}^{2d \times 2d}$:

$$\Omega := \begin{bmatrix} \mathbf{0} & \mathbf{I}_d \\ -\mathbf{I}_d & \mathbf{0} \end{bmatrix}$$

where \mathbf{I}_d is the identity matrix. Note that Ω is a skew-symmetric matrix, i.e., $\Omega^\top = -\Omega$.

Definition 3 Let $H: \mathbb{R}^{2d} \rightarrow \mathbb{R}$ be a continuously differentiable Hamiltonian function. The Hamiltonian flow with respect to H is the system of differential equations on the phase space given by

$$\dot{Z}_t = \Omega \nabla H(Z_t). \tag{HF_{gen}}$$

When the Hamiltonian function is separable, $H(x, y) = f(x) + g(y)$, the Hamiltonian dynamics in HF_{gen} reduces to

$$\dot{X}_t = \nabla g(Y_t), \quad \dot{Y}_t = -\nabla f(X_t).$$

Note that when $g(y) = \frac{1}{2}\|y\|^2$, this recovers HF in Section 2.1. Throughout this paper, we assume that the Hamiltonian flow admits a unique solution for any initial condition $Z_0 = (X_0, Y_0) = (x, y)$, and the solution at time t is given by $Z_t = \text{HF}((x, y); t)$ as introduced in Section 2.1. See e.g., Leimkuhler and Reich (2005, §3.1, §3.2) for additional details and examples of Hamiltonian functions and dynamics in physical contexts.

B.1. Properties of the Hamiltonian flow

The Hamiltonian flow satisfies desirable properties in addition to the conservation property stated as Lemma 1 (stated for a specific Hamiltonian, but holds more generally). These properties also play a crucial role in the development of Hamiltonian Monte Carlo algorithm for sampling. Throughout, we assume that the Hamiltonian is a continuously differentiable function over the phase space.

Conservation of Hamiltonian

Lemma 8 For any time $t > 0$ and $x, y \in \mathbb{R}^d$, $H(\text{HF}((x, y); t)) = H(x, y)$.

Proof Let $Z_t = \text{HF}((x, y); t)$. Since H is continuously differentiable, by the chain rule we have

$$\frac{d}{dt}H(Z_t) = \nabla H(Z_t)^\top \dot{Z}_t = \nabla H(Z_t)^\top \Omega \nabla H(Z_t) = 0$$

where we use the fact that $v^\top \Omega v = 0$ for any vector v , since Ω is skew-symmetric. \blacksquare

Symplecticity and preservation of volume

Definition 4 A differentiable map $\Phi: \mathbb{R}^{2d} \rightarrow \mathbb{R}^{2d}$ is called symplectic if for every $z \in \mathbb{R}^{2d}$, its Jacobian $\nabla \Phi(z)$ satisfies

$$\nabla \Phi(z)^\top \Omega \nabla \Phi(z) = \Omega .$$

Definition 5 A differentiable map $\Phi: \mathbb{R}^{2d} \rightarrow \mathbb{R}^{2d}$ is called volume-preserving if for every $z \in \mathbb{R}^{2d}$, its Jacobian $\nabla \Phi(z)$ satisfies

$$|\det(\nabla \Phi(z))| = 1 .$$

The volume-preservation property is named as such since for any measurable set $A \in \mathbb{R}^{2d}$ and Φ that satisfies $|\det(\nabla \Phi(z))| = 1$, we have $\text{vol}(\Phi(A)) = \text{vol}(A)$. This follows by applying the change-of-variables formula, which gives:

$$\text{vol}(\Phi(A)) = \int_A |\det(\nabla \Phi(z_0))| dz_0 = \int_A dz_0 = \text{vol}(A).$$

Lemma 9 For any $t > 0$, the Hamiltonian flow map $(x, y) \mapsto \text{HF}((x, y); t)$ associated with a twice continuously Hamiltonian H is both symplectic and volume-preserving.

Proof For any $z_0 \in \mathbb{R}^{2d}$, define $J(t) := \nabla_z \text{HF}(z; t)|_{z=z_0}$. Since $\dot{z}_t = \Omega \nabla H(z_t)$ along Hamiltonian flow HF_{gen} , we have

$$\dot{J}(t) = \Omega \nabla^2 H(z_t) J(t) ; \quad J(0) = I_{2d} .$$

Define $M(t) := J(t)^\top \Omega J(t)$. By the product rule,

$$\begin{aligned} \dot{M}(t) &= \dot{J}(t)^\top \Omega J(t) + J(t)^\top \Omega \dot{J}(t) \\ &= J(t)^\top \nabla^2 H(z_t) \Omega^\top \Omega J(t) + J(t)^\top \Omega^2 \nabla^2 H(z_t) J(t) \\ &= -J(t)^\top \nabla^2 H(z_t) J(t) + J(t)^\top \nabla^2 H(z_t) J(t) = 0 . \end{aligned}$$

In the third equality, we used the fact that $\Omega^2 = -I_{2d}$ and $\Omega^\top \Omega = I_{2d}$. In simpler terms, for $M(t)$ is a constant function. Consequently,

$$M(t) = M(0) \Rightarrow J(t)^\top \Omega J(t) = \Omega .$$

Taking determinants on both sides and using the product rule results in

$$\det(J(t))^2 = \det(\Omega)^2 = 1 .$$

However since $\det(J(0)) = 1$ and since $t \mapsto J(t)$ is continuous, $\det J(t) = 1$. \blacksquare

Consequently, for any measurable set $A \subseteq \mathbb{R}^{2d}$ and for all $t > 0$:

$$\text{vol}(\text{HF}(A; t)) = \text{vol}(A).$$

Time reversibility

Lemma 10 *Assume that the Hamiltonian H satisfies $H(x, y) = H(x, -y)$ for all x, y , and let $\text{HF}(\cdot; t)$ denote the corresponding Hamiltonian flow map as defined by HF_{gen} . If $(x_t, y_t) = \text{HF}((x_0, y_0); t)$ for any time $t \geq 0$ and initial conditions (x_0, y_0) , then $\text{HF}((x_t, -y_t); t) = (x_0, -y_0)$.*

Proof Let $(X_s, Y_s)_{s \geq 0}$ be the unique solution of HF_{gen} with $(X_0, Y_0) = (x_0, y_0)$, and suppose that $(X_t, Y_t) = (x_t, y_t)$ for some $t \geq 0$. Define a new pair of curves on $[0, t]$ by

$$\tilde{X}_s := X_{t-s}, \quad \tilde{Y}_s := -Y_{t-s}.$$

Note that the derivative of an even function is odd. The assumption about H states that for any $x, y \mapsto H(x, y)$ is an even function, and hence $\nabla_y H(x, y) = -\nabla_y H(x, -y)$. Therefore, by the chain rule,

$$\dot{\tilde{X}}_s = -\dot{X}_{t-s} = -\nabla_y H(X_{t-s}, Y_{t-s}) = \nabla_y H(X_{t-s}, -Y_{t-s}) = \nabla_y H(\tilde{X}_s, \tilde{Y}_s).$$

Similarly,

$$\dot{\tilde{Y}}_s = -(-\dot{Y}_{t-s}) = \dot{Y}_{t-s} = -\nabla_x H(X_{t-s}, Y_{t-s}) = -\nabla_x H(X_{t-s}, -Y_{t-s}) = -\nabla_x H(\tilde{X}_s, \tilde{Y}_s).$$

So $(\tilde{X}_s, \tilde{Y}_s)$ also satisfies the Hamiltonian dynamics $\dot{X}_s = Y_s, \dot{Y}_s = -\nabla f(X_s)$ on $[0, t]$. Moreover, $(\tilde{X}_0, \tilde{Y}_0) = (X_t, -Y_t) = (x_t, -y_t)$ and $(\tilde{X}_t, \tilde{Y}_t) = (X_0, -Y_0) = (x_0, -y_0)$. By uniqueness of solutions, the solution starting from $(x_t, -y_t)$ must equal $(\tilde{X}_s, \tilde{Y}_s)$ on $[0, t]$, and in particular at time t we obtain $\text{HF}((x_t, -y_t); t) = (x_0, -y_0)$. \blacksquare

B.2. Discussion of integrators

As discussed in [Section 1](#), the Hamiltonian dynamics HF_{gen} is exactly solvable only in special cases, for example, when $H(x, y) = x^\top Ax + y^\top By$. In general, we need to *discretize* the dynamics. Here we discuss several elementary discretization schemes; see [Hairer et al. \(2006, Chap. 2\)](#), [Leimkuhler and Reich \(2005, Chaps. 2, 4\)](#) for a more detailed exposition.

B.2.1. EXPLICIT AND IMPLICIT INTEGRATORS

Definition 6 *The explicit integrator / forward Euler integrator of the Hamiltonian flow (HF_{gen}) from initial values (x_0, y_0) with step size $\eta > 0$ generates the sequence $\{(x_n, y_n)\}_{n \geq 0}$ where (x_n, y_n) satisfies the recursion*

$$\begin{aligned} x_{n+1} - x_n &= \eta \cdot \nabla_y H(x_n, y_n), \\ y_{n+1} - y_n &= -\eta \cdot \nabla_x H(x_n, y_n). \end{aligned} \tag{8}$$

More succinctly, using $z_n := (x_n, y_n)$,

$$z_{n+1} - z_n = \eta \cdot \Omega \nabla H(z_n).$$

Definition 7 The implicit integrator / backward Euler integrator of the Hamiltonian flow (HF_{gen}) from initial values (x_0, y_0) with step size $\eta > 0$ generates the sequence $\{(x_n, y_n)\}_{n \geq 0}$ where (x_n, y_n) satisfies the recursion

$$\begin{aligned} x_{n+1} - x_n &= \eta \cdot \nabla_y H(x_{n+1}, y_{n+1}), \\ y_{n+1} - y_n &= -\eta \cdot \nabla_x H(x_{n+1}, y_{n+1}). \end{aligned} \quad (9)$$

More succinctly, using $z_n := (x_n, y_n)$,

$$z_{n+1} - z_n = \eta \cdot \Omega \nabla H(z_{n+1}).$$

Remark 2 The explicit integrator is implementable whenever ∇H is accessible. The implicit integrator, however, requires solving a nonlinear system. Consider the special case $H(x, y) = f(x) + \frac{1}{2}\|y\|^2$, for which $\nabla_y H(x, y) = y$ and $\nabla_x H(x, y) = \nabla f(x)$. Then the implicit updates read:

$$\begin{aligned} x_{n+1} - x_n &= \eta y_{n+1}, \\ y_{n+1} - y_n &= -\eta \nabla f(x_{n+1}) \end{aligned}$$

which is precisely HF^{imp} . Eliminating y_{n+1} in the update for x_{n+1} in HF^{imp} yields the implicit equation

$$x_{n+1} = x_n + \eta y_n - \eta^2 \nabla f(x_{n+1}),$$

or equivalently,

$$x_{n+1} = \arg \min_{x \in \mathbb{R}^d} \left\{ f(x) + \frac{1}{2\eta^2} \|x - (x_n + \eta y_n)\|^2 \right\} = \text{Prox}_{\eta^2 f}(x_n + \eta y_n)$$

where $\text{Prox}_{\eta^2 f}$ is the proximal operator for $x \mapsto \eta^2 f(x)$ (Parikh et al., 2014). The update for y_{n+1} is the same as in HF^{imp} . Thus, when a proximal oracle for f is available, the implicit integrator admits a simple implementation

$$x_{n+1} = \text{Prox}_{\eta^2 f}(x_n + \eta y_n), \quad y_{n+1} = y_n - \eta \nabla f(x_{n+1}).$$

The extragradient integrator is formed by replacing the call to the proximal oracle for f by a gradient descent step. See Definition 2 for more detail.

Monotonicity properties for convex Hamiltonian. A key property about the explicit (resp. implicit) integrator is that it produces iterates along which the Hamiltonian values are monotonically increasing (resp. decreasing) when the Hamiltonian H is a convex function in the phase space. We formalize this in following lemmas. We note the following results only use the property that the Hamiltonian flow is a skew-gradient flow (i.e., the velocity vector field is orthogonal to the gradient); see also the discussion in Wibisono et al. (2022, Appendix B).

Lemma 11 Assume that the Hamiltonian H is convex. Suppose (x_{n+1}, y_{n+1}) is produced by one step of the explicit integrator (Eq. (8)) from (x_n, y_n) with step size $\eta > 0$. Then,

$$H(x_{n+1}, y_{n+1}) \geq H(x_n, y_n).$$

Proof We use the notation z_n and z_{n+1} to denote the updates in phase space for convenience. By convexity of H we have

$$\begin{aligned} H(z_{n+1}) &\geq H(z_n) + \langle \nabla H(z_n), z_{n+1} - z_n \rangle \\ &= H(z_n) + \eta \langle \nabla H(z_n), \Omega \nabla H(z_n) \rangle \\ &= H(z_n) \end{aligned}$$

where the final step uses the fact that $\langle v, \Omega v \rangle = 0$ since Ω is skew-symmetric. ■

Lemma 12 *Assume that the Hamiltonian H is convex. Suppose (x_{n+1}, y_{n+1}) is produced by one step of the implicit integrator (Eq. (9)) from (x_n, y_n) with step size $\eta > 0$. Then,*

$$H(x_{n+1}, y_{n+1}) \leq H(x_n, y_n) .$$

Proof We use the notation z_n and z_{n+1} to denote the updates in phase space for convenience. By convexity of H we have

$$\begin{aligned} H(z_{n+1}) &\leq H(z_n) + \langle \nabla H(z_{n+1}), z_{n+1} - z_n \rangle \\ &= H(z_n) + \eta \langle \nabla H(z_{n+1}), \Omega \nabla H(z_{n+1}) \rangle \\ &= H(z_n) \end{aligned}$$

where the final step uses the fact that $\langle v, \Omega v \rangle = 0$ since Ω is skew-symmetric. ■

B.2.2. LEAPFROG INTEGRATOR

We describe another type of integrator called the leapfrog integrator. This differs from the design of the explicit and implicit integrator due to interleaved updates for y and x as described below.

Definition 8 *The leapfrog integrator of the Hamiltonian flow (HF_{gen}) from initial values (x_0, y_0) with step size $\eta > 0$ generates the sequence $\{(x_n, y_n)\}_{n \geq 0}$ where (x_n, y_n) satisfies the recursion*

$$\begin{aligned} y_{n+\frac{1}{2}} &= y_n - \frac{\eta}{2} \nabla_x H(x_n, y_n) \\ x_{n+1} &= x_n + \eta \nabla_y H(x_n, y_{n+\frac{1}{2}}) \\ y_{n+1} &= y_{n+\frac{1}{2}} - \frac{\eta}{2} \nabla_x H(x_{n+1}, y_{n+\frac{1}{2}}). \end{aligned}$$

Remark 3 *When H is separable, i.e. of the form $H(x, y) = f(x) + g(y)$, the updates for y in the leapfrog integrator can be combined, and the resulting update rule is known as the symplectic Euler integrator for the separable Hamiltonian, which performs the update for the x and y variable in an alternating way:*

$$x_{n+1} = x_n + \eta \nabla_y H(x_n, y_n), \quad y_{n+1} = y_n - \eta \nabla_x H(x_{n+1}, y_n) .$$

While the explicit and implicit Euler schemes result in monotonic sequences of Hamiltonian values (Lemmas 11 and 12), the Hamiltonian $H(x_n, y_n)$ along the iterates of the leapfrog integrator does *not* change monotonically. Instead, it enjoys a higher-order *one-step conservation error of Hamiltonian*: under mild smoothness assumptions,

$$|H(x_{n+1}, y_{n+1}) - H(x_n, y_n)| = \mathcal{O}(\eta^3) \quad (\eta \rightarrow 0),$$

whereas the explicit/implicit Euler schemes have a one-step defect of order $\mathcal{O}(\eta^2)$. We refer readers to Section 3.3 of Bou-Rabee and Sanz-Serna (2018) for a detailed discussion.

Another important property of leapfrog integrator is the preservation of symplecticity of Hamiltonian flow (see Lemma 9); the symplecticity of the leapfrog integrator can be found in Bou-Rabee and Sanz-Serna (2018, Theorem 4.1). In the continuous-time analysis of the present work, however, symplecticity does not play a direct role, and therefore we did not use leapfrog integrator in our discretization analysis. That said, a number of previous works on Hamiltonian optimization, including Betancourt et al. (2026); Tran and Leok (2023); Duruisseaux and Leok (2023); Lin et al. (2023); Zhu and Wang (2026), have shown that symplectic integrators can substantially improve the stability of the discrete trajectory, which in turn may lead to better convergence behavior. In the context of Hamiltonian Monte Carlo, the use of symplectic integrators is even more fundamental, as it is closely tied to the preservation of the target probability distribution. We refer to Bou-Rabee and Sanz-Serna (2018) for a thorough discussion.

Appendix C. A Review of Hamiltonian Monte Carlo for Sampling

We review the Hamiltonian Monte Carlo (HMC) framework for sampling (Duane et al., 1987; Neal, 2011). Given a function $f: \mathbb{R}^d \rightarrow \mathbb{R}$, the goal in sampling is to draw approximate samples from the target distribution $\nu(x) \propto e^{-f(x)}$ on \mathbb{R}^d . We denote the set of probability distributions on \mathbb{R}^d by $\mathcal{P}(\mathbb{R}^d)$. Assuming that the Hamiltonian dynamics HF can be solved exactly, the exact HMC algorithm to sample from ν is defined in eHMC (Algorithm 5).

Algorithm 5 Exact Hamiltonian Monte Carlo (eHMC)

Input : Initial distribution $\mu \in \mathcal{P}(\mathbb{R}^d)$, Integration time $T > 0$, Number of iterations $K \in \mathbb{N}$
 Sample $X_0 \sim \mu$
for $k = 1$ **to** K **do**
 | Sample $\xi \sim \mathcal{N}(0, I)$
 | Define $X_k := \Pi_1 \circ \text{HF}((X_{k-1}, \xi); T)$
end
return X_K

The difference between eHMC for sampling and HFopt for optimization is that for sampling, the velocity is periodically refreshed to be an independent draw from a standard Gaussian $\xi \sim \mathcal{N}(0, I_d)$, instead of deterministically set to $\mathbf{0} \in \mathbb{R}^d$ for optimization.

Since HF can be implemented exactly only in special cases, a discretization scheme of the Hamiltonian dynamics has to be utilized, leading to the unadjusted HMC algorithm (Bou-Rabee and Eberle, 2023; Bou-Rabee et al., 2026); several discretization schemes of the Hamiltonian dynamics are discussed in Appendix B with a popular method being the Störmer–Verlet or leapfrog integrator.

Define $\pi = \nu \otimes \mathcal{N}(0, I) \in \mathcal{P}(\mathbb{R}^{2d})$. Therefore, π is a distribution defined on the phase space and note that

$$\pi(x, y) \propto e^{-H(x, y)}$$

where the Hamiltonian energy function $H : \mathbb{R}^{2d} \rightarrow \mathbb{R}$ is defined in Section 2.1. To see that ν is in fact the stationary distribution for eHMC, we will show that the Hamiltonian flow (HF) preserves π , i.e., for any $t \geq 0$

$$\text{HF}(\cdot; t)_{\#}\pi = \pi.$$

Let $\tilde{\pi} = \text{HF}(\cdot; t)_{\#}\pi$. Then, by the change of variable formula, we have that for any $(x, y) \in \mathbb{R}^{2d}$,

$$\pi(x, y) = \tilde{\pi}(\text{HF}((x, y); t)) |\det(\nabla \text{HF}((x, y); t))| = \tilde{\pi}(\text{HF}((x, y); t))$$

where the last equality is because the Hamiltonian flow preserves volume; see Appendix B. Thus,

$$\tilde{\pi}(\text{HF}((x, y); t)) = \pi(x, y) = Z^{-1}e^{-H(x, y)} = Z^{-1}e^{-H(\text{HF}((x, y); t))}$$

where Z is a normalizing constant and the final equality is due to conservation of the Hamiltonian along the Hamiltonian flow (see Appendix B). Hence $\tilde{\pi} = \pi$, and therefore, the position marginal ν is stationary for eHMC, when the velocity marginal is $\mathcal{N}(0, I_d)$ and is independent with ν .

Further properties of eHMC can be found in Bou-Rabee et al. (2026, Section A) and a discussion on how eHMC and HFopt fit into a general Lift-Conserve-Project (LCP) scheme can be found in Fu and Wibisono (2025, Section B). For a discussion on prior works studying eHMC and unadjusted HMC, see e.g., Bou-Rabee et al. (2026, Section 1.3).

Appendix D. Lower Bound for HFopt

Lemma 13 *Fix $c > 0$. For every $\kappa \geq \max\{1, c^2\}$, there exists an α -strongly convex and $L = \kappa\alpha$ -smooth quadratic function*

$$f(x) = \frac{\alpha}{2}x_1^2 + \frac{L}{2}x_2^2, \quad x = (x_1, x_2)^\top,$$

such that the algorithmic update (HFopt) with integration time $T = \frac{c}{\sqrt{L}}$, initialized at $x_0 = (x_{0,1}, 0)$, requires $k = \Omega(\kappa \log(1/\varepsilon))$ many iterations to ensure that $f(x_k) - f(x^*) \leq \varepsilon$. The total integration time $T \times k$ is therefore $\Omega(\frac{c\sqrt{L}}{\alpha} \log(1/\varepsilon))$.

Proof As f is separable, the dynamics (HF) decouple along both coordinates and are given by

$$\ddot{X}_t = -\alpha X_t, \quad \ddot{Y}_t = -L Y_t.$$

As the initial velocity in Step (ii) of HFopt is $\mathbf{0} \in \mathbb{R}^2$, we have that $X'_0 = Y'_0 = 0$. Note that in the proof of this lemma, $X, Y \in \mathbb{R}$ and denote the coordinates of the position space \mathbb{R}^2 .

For any $k \in \mathbb{N}$, denote the k th iterate of the algorithm (HFopt) by $x_k = (x_{k,1}, x_{k,2}) \in \mathbb{R}^2$. Given integration time $T > 0$, the update is therefore

$$x_{k+1,1} = x_{k,1} \cos(\sqrt{\alpha}T), \quad x_{k+1,2} = x_{k,2} \cos(\sqrt{L}T),$$

which can be rewritten as

$$x_{k+1,1} = x_{k,1} \cos\left(\frac{c}{\sqrt{\kappa}}\right), \quad x_{k+1,2} = x_{k,2} \cos(c).$$

Considering the initialization $x_0 = (a, 0)$, we get that $x_{k,2} \equiv 0$ for all $k \geq 0$ and that

$$x_{k,1} = x_{0,1} \cdot \cos^k\left(\frac{c}{\sqrt{\kappa}}\right).$$

Therefore,

$$f(x_k) - f(x^*) = \frac{\alpha}{2} x_{k,1}^2 = \frac{\alpha}{2} x_{0,1}^2 \cos^{2k}\left(\frac{c}{\sqrt{\kappa}}\right) = \cos^{2k}\left(\frac{c}{\sqrt{\kappa}}\right) (f(x_0) - f(x^*)).$$

As $\kappa \geq c^2$, we have that $0 \leq \frac{c}{\sqrt{\kappa}} \leq 1$. Therefore, using

$$\cos(u) \geq 1 - \frac{u^2}{2}$$

for $0 \leq u \leq 1$ implies

$$f(x_k) - f(x^*) \geq \left(1 - \frac{c^2}{2\kappa}\right)^{2k} (f(x_0) - f(x^*)),$$

which implies the claimed iteration complexity. ■

Appendix E. Extended discussion of [Section 3](#)

E.1. Examples

E.1.1. A QUADRATIC SETTING

As an example, we consider a quadratic objective function and illustrate how the final endpoint of Hamiltonian flow trajectories compares against averaged trajectories. We take the objective function to be $f(x) = \frac{1}{2}x^\top \Lambda x$ where Λ is the 2×2 diagonal matrix $\Lambda = \begin{pmatrix} 2 & 0 \\ 0 & 10 \end{pmatrix}$. In this setting, the solution to the Hamiltonian dynamics is available in closed form. We set the initial position to be $X_0 = [1, 1]^\top$, integration time to be $T = 4$, and plot the three trajectories:

- the position $(X_t)_{t \in [0, T]}$,
- the aggregated position $(X^{\text{avg}}(X_0; t))_{t \in [0, T]}$ as defined in [Eq. \(2\)](#),
- the aggregated position $(X^{\text{s-avg}}(X_0; t))_{t \in [0, T]}$ according to $X^{\text{s-avg}}(X_0; t) = \frac{1}{t} \int_0^t X_s \, ds$.

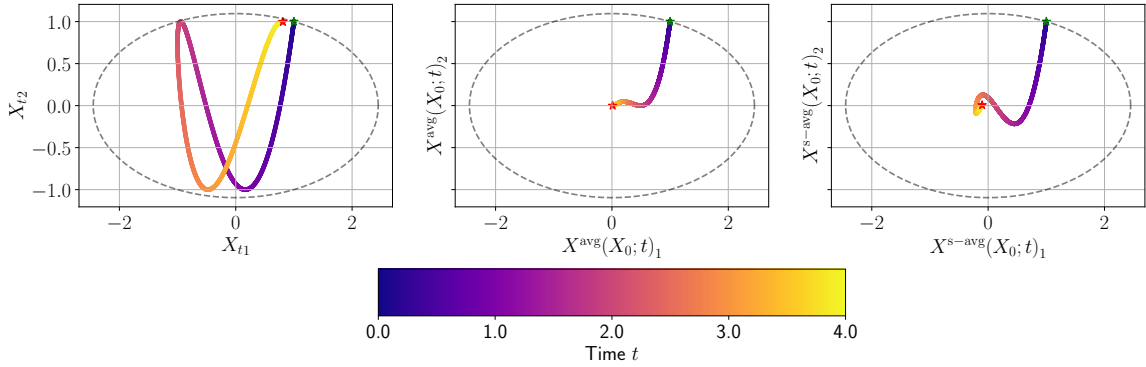


Figure 1: The dotted lines represent the level set of f , and green and red stars highlight the position at time $t = 0$ and $t = T$ respectively.

The minimizer of f is at $[0, 0]^\top$ and Fig. 1 shows that the position X_t exhibits oscillatory behavior whereas the averages $X^{s\text{-avg}}(X_0; t)$ and $X^{\text{avg}}(X_0; t)$ approach the minimizer. In Fig. 2 we plot the function values corresponding to the positions plotted in Fig. 1, and also plot the following function values:

- $f^{\text{avg}}(X_0; t) := \frac{2}{t^2} \int_0^t (t-s)f(X_s)ds$,
- $f^{s\text{-avg}}(X_0; t) := \frac{1}{t} \int_0^t f(X_s)ds$.

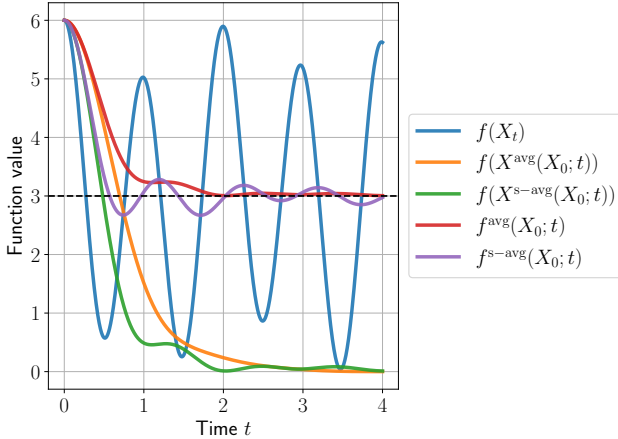


Figure 2: Evolution of function values with time t .

We plot f^{avg} and $f^{s\text{-avg}}$ because the former is the quantity that we bound; see Remark 1, and the latter is the function value analogue of $X^{s\text{-avg}}$. We note two key observations from this plot: (a) the function values along the aggregated position decay to 0, and (b) $f^{\text{avg}}(X_0; t)$, $f^{s\text{-avg}}(X_0; t)$ oscillates about $\frac{1}{2}f(X_0)$ with varying amplitudes. The first observation speaks to the nature of oscillatory behaviour of X_t shown in Fig. 1, and the second observation suggests our analysis might be able to be tightened, but it would not qualitatively affect the convergence rates.

E.1.2. A STRICTLY CONVEX SETTING

In this example, we consider a 1 dimensional function $f(x) = e^x - x - 1$. This is a strictly convex function on account of $\frac{d^2}{dx^2}f(x) = e^x > 0$. This instructive example is meant to demonstrate that (a) the time averages of the position does not necessarily converge to the minimizer which is at 0 with $f^* = 0$, and (b) the optimality gap of the function averages $f^{\text{avg}}(X_0; t)$ and $f^{s\text{-avg}}(X_0; t)$ do not necessarily oscillate about $\frac{1}{2}(f(X_0) - f^*)$.

Here, we do not have access to a closed form solution for HF, and instead use a numerical integrator – specifically the DOP853 integrator of Dormand and Prince as implemented in the `dop853` integrator in `SciPy` (Virtanen et al., 2020).

We initialize $X_0 = 1$, and set $T = 20$. We plot same three trajectories as before, but the aggregated positions are approximated due to the unavailability of closed form solutions. Specifically, for $\eta = 10^{-3}$ and $k \in \{0, \dots, T/\eta\}$

$$X^{\text{avg}}(X_0; k\eta) \approx \frac{2}{k(k+1)} \sum_{i=0}^k \sum_{j=0}^i X_{j\eta},$$

$$X^{\text{s-avg}}(X_0; k\eta) \approx \frac{1}{k+1} \sum_{i=0}^k X_{i\eta}.$$

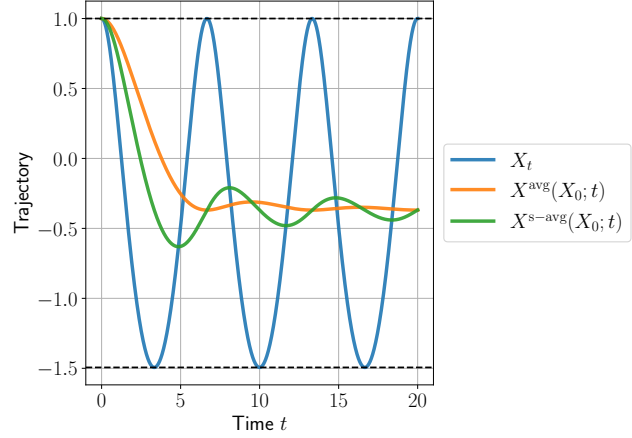


Figure 3: The dotted lines are the boundaries of the sublevel set of f corresponding to $f(x) \leq f(X_0)$.

In Figure 3, we observe a similar oscillatory behavior for X_t , while the averaged positions oscillate less so. Notably, these do not oscillate close to 0, which we recall is the minimizer of this function. We also plot the function values associated with these trajectories, and f^{avg} and $f^{\text{s-avg}}$ as described in the previous subsection – see Figure 4. Again, due to the unavailability of closed form solution, we approximate the integrals in f^{avg} and $f^{\text{s-avg}}$ as a finite-sum in the same manner as done for the approximations of the aggregated positions.

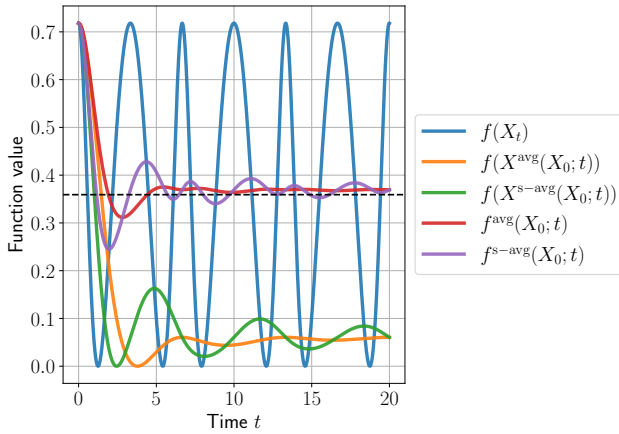


Figure 4: Evolution of function values with time t .

Here, the dotted line represents $\frac{1}{2}f(X_0)$. We see that the time averages of the function values $f^{\text{avg}}(X_0; t)$ and $f^{\text{s-avg}}(X_0; t)$ oscillates about a value that is slightly higher than this line with varying amplitudes. This does not invalidate the result of Corollary 1 which suggests the level $\frac{2}{3}f(X_0) \approx 0.48$. Consistent with the variation shows in Figure 3, we see that the function values of the aggregated positions do not converge to 0, which is in contrast to what was observed in the quadratic setting (Figure 2).

E.2. Proofs of main lemmas in Section 3

E.2.1. PROOF OF COROLLARY 1

To prove Corollary 1, we first state and prove the following more general lemma.

Lemma 14 *Let f satisfy \mathbf{A}_{cvx} . Consider a non-negative weighting function $\mathbf{w}(t)$ over $[0, T]$ such that $\int_0^T \mathbf{w}(t)dt = 1$, and define the \mathbf{w} -weighted average of $(\text{HF}((X_0, \mathbf{0}); \tau))_{\tau \in [0, T]}$ as*

$$X^{\mathbf{w}}(X_0; T) := \int_0^T \mathbf{w}(t)X_t dt .$$

Then,

$$\begin{aligned} f(X^{\mathbf{w}}(X_0; T)) - f(z) &\leq \int_0^T \mathbf{w}(t)(f(X_t) - f(z))dt \\ &\leq \frac{2}{3}(f(X_0) - f(z)) - \frac{1}{3}\mathbf{w}(T)\langle X_T - z, Y_T \rangle + \frac{1}{3} \int_0^T \dot{\mathbf{w}}(t)\langle X_t - z, Y_t \rangle dt . \end{aligned}$$

Proof Recall that $h(t) = \langle X_t - z, Y_t \rangle$ and $h(0) = 0$. Multiplying both sides of the result of [Lemma 2](#) by $\mathbf{w}(t)$ which is non-negative, we obtain

$$\mathbf{w}(t)(f(X_t) - f(z)) \leq \frac{2}{3}\mathbf{w}(t)(f(X_0) - f(z)) - \frac{1}{3}\mathbf{w}(t)\dot{h}(t) .$$

Integrating both sides from 0 to T , we get the following:

LHS: By convexity of f ,

$$\int_0^T \mathbf{w}(t)(f(X_t) - f(z)) dt \geq f\left(\int_0^T \mathbf{w}(t)X_t dt\right) - f(z) = f(X^{\mathbf{w}}(X_0; T)) - f(z) .$$

The inequality is due to Jensen's inequality as $\int_0^T \mathbf{w}(t)dt = 1$.

RHS: Using integration-by-parts,

$$-\frac{1}{3} \int_0^T \mathbf{w}(t)\dot{h}(t)dt = -\frac{1}{3} \{w(T)h(T) - w(0)h(0)\} + \frac{1}{3} \int_0^T \dot{\mathbf{w}}(t)h(t)dt .$$

This completes the proof. ■

We now provide the proof of [Corollary 1](#).

Proof of Corollary 1 As defined in [Eq. \(2\)](#), $X^{\text{avg}}(X_0; T)$ is the \mathbf{w} -weighted average with weight $\mathbf{w}(t) = \frac{2}{T^2}(T - t)$, so $\dot{\mathbf{w}}(t) = -\frac{2}{T^2}$ and $\mathbf{w}(T) = 0$. Instantiating [Lemma 14](#) with this weight gives

$$f(X^{\text{avg}}(X_0; T)) - f(z) \leq \frac{2}{3}(f(X_0) - f(z)) - \frac{2}{3T^2} \int_0^T \langle X_t - z, Y_t \rangle dt .$$

Since $\frac{d}{dt}\|X_t - z\|^2 = 2\langle X_t - z, Y_t \rangle$, we can simplify the above to

$$f(X^{\text{avg}}(X_0; T)) - f(z) \leq \frac{2}{3}(f(X_0) - f(z)) - \frac{1}{3T^2}\|X_T - z\|^2 + \frac{1}{3T^2}\|X_0 - z\|^2$$

which proves the corollary. ■

E.2.2. VERSIONS OF [LEMMA 2](#) FOR THE AVERAGE AND MINIMIZER OVER THE TRAJECTORY

Corollary 6 Consider the setting of [Lemma 2](#). Let $X^{\text{s-avg}}(X_0; T)$ be the simple average of X_t along the Hamiltonian flow trajectory $(\text{HF}((X_0, \mathbf{0}); \tau))_{\tau \in [0, T]}$. Then,

$$f(X^{\text{s-avg}}(X_0; T)) - f(x^*) \leq \frac{2}{3}(f(X_0) - f(x^*)) + \frac{1}{3T} \|X_T - x^*\| \cdot \sqrt{2(f(X_0) - f(X_T))}.$$

Moreover, when f satisfies [A \$_{\alpha}\$ -qg](#),

$$f(X^{\text{s-avg}}(X_0; T)) - f(x^*) \leq \left(\frac{2}{3} + \frac{2}{3T\sqrt{\alpha}} \right) (f(X_0) - f(x^*)).$$

Proof We instantiate [Lemma 14](#) with $w(t) = \frac{1}{T}$. This gives

$$f(X^{\text{s-avg}}(X_0; T)) - f(z) \leq \frac{2}{3}(f(X_0) - f(z)) - \frac{1}{3T} \langle X_T - z, Y_T \rangle.$$

The inner product can be bounded from above using Cauchy-Schwarz inequality as

$$-\langle X_T - z, Y_T \rangle \leq \|X_T - z\| \cdot \|Y_T\| \leq \|X_T - z\| \cdot \sqrt{2(f(X_0) - f(X_T))}.$$

where the last inequality uses the energy conservation along the Hamiltonian flow trajectory to bound $\|Y_T\|$.

When f satisfies [A \$_{\alpha}\$ -qg](#), using the energy conservation along the Hamiltonian flow trajectory, we have

$$\|X_T - x^*\| \leq \sqrt{\frac{2(f(X_T) - f(x^*))}{\alpha}} \leq \sqrt{\frac{2(f(X_0) - f(x^*))}{\alpha}}.$$

Substituting this in the first statement of the corollary along with $f(x^*) \leq f(X_T)$ by definition yields

$$f(X^{\text{s-avg}}(X_0; T)) - f(x^*) \leq \frac{2}{3}(f(X_0) - f(x^*)) + \frac{2}{3T\sqrt{\alpha}}(f(X_0) - f(x^*)).$$

■

The corollary for the minimizer along the trajectory is simpler to obtain owing to the fact that the minimum is at most any weighted average.

Corollary 7 Consider the setting of [Lemma 2](#). Let $X^{\text{min}}(X_0; T)$ denote the minimizer of f along the Hamiltonian flow trajectory $(\text{HF}((X_0, \mathbf{0}); \tau))_{\tau \in [0, T]}$, i.e., $f(X^{\text{min}}(X_0; T)) \leq f(X_t)$ for all $t \in [0, T]$. Then,

$$f(X^{\text{min}}(X_0; T)) - f(x^*) + \frac{1}{3T^2} \|X_T - x^*\|^2 \leq \frac{2}{3}(f(X_0) - f(x^*)) + \frac{1}{3T^2} \|X_0 - x^*\|^2.$$

Moreover, when f is satisfies [A \$_{\alpha}\$ -qg](#),

$$f(X^{\text{min}}(X_0; T)) - f(x^*) \leq \left(\frac{2}{3} + \frac{2}{3T^2\alpha} \right) (f(X_0) - f(x^*)).$$

Proof We use the following corollary of [Lemma 2](#), which we obtain from integrating both sides of [Eq. \(3\)](#) twice:

$$\frac{2}{T^2} \int_0^T (T-t)(f(X_t) - f(z))dt + \frac{1}{3T^2} \|X_T - z\|^2 \leq \frac{2}{3}(f(X_0) - f(z)) + \frac{1}{3T^2} \|X_0 - z\|^2.$$

Since $f(X^{\min}(X_0; T)) \leq f(X_t)$, letting $z = x^*$ gives:

$$\frac{2}{T^2} \int_0^T (T-t)(f(X^{\min}(X_0; T)) - f(x^*))dt + \frac{1}{3T^2} \|X_T - x^*\|^2 \leq \frac{2}{3}(f(X_0) - f(x^*)) + \frac{1}{3T^2} \|X_0 - x^*\|^2.$$

Since $\int_0^T (T-t)dt = \frac{T^2}{2}$, this further simplifies into:

$$f(X^{\min}(X_0; T)) - f(x^*) + \frac{1}{3T^2} \|X_T - x^*\|^2 \leq \frac{2}{3}(f(X_0) - f(x^*)) + \frac{1}{3T^2} \|X_0 - x^*\|^2.$$

When f is satisfies [A \$_{\alpha}\$ -qg](#), $\|X_0 - x^*\|^2 \leq \frac{2}{\alpha}(f(X_0) - f(x^*))$. Applying this bound and dropping the $\frac{1}{3T^2} \|X_T - x^*\|^2$ -term on the left gives:

$$f(X^{\min}(X_0; T)) - f(x^*) \leq \left(\frac{2}{3} + \frac{2}{3T^2\alpha} \right) (f(X_0) - f(x^*)).$$

■

E.2.3. PROOF OF [THEOREM 1](#) AND [COROLLARY 2](#)

Proof of [Theorem 1](#) Notice that at each iteration k , $X_k = X^{\text{avg}}(X_{k-1}; T_k)$. Therefore, from [Lemma 3](#) and the lower bound for T_k , we have

$$f(X_k) - f(x^*) \leq \frac{2}{3} \left(1 + \frac{1}{C^2} \right) (f(X_{k-1}) - f(x^*)).$$

Iterating this inequality, we obtain the statement of the lemma. ■

Proof of [Corollary 2](#) From [Theorem 1](#), we have for $T_k = T = \frac{5}{2\sqrt{\alpha}}$ that

$$f(X_K) - f(x^*) \leq \left(\frac{58}{75} \right)^K \cdot (f(X_0) - f(x^*)).$$

Since $K \geq 4 \log \frac{f(X_0) - f(x^*)}{\varepsilon} \geq \left(\log \frac{75}{58} \right)^{-1} \log \frac{f(X_0) - f(x^*)}{\varepsilon}$,

$$f(X_K) - f(x^*) \leq \varepsilon.$$

The total integration time is $\sum_{k=1}^K T_k = \left\lceil 4 \log \frac{f(X_0) - f(x^*)}{\varepsilon} \right\rceil \frac{5}{2\sqrt{\alpha}}$. ■

E.2.4. PROOF OF LEMMA 3

Proof From Corollary 1, we have

$$f(X^{\text{avg}}(X_0; T)) - f(x^*) \leq \frac{2}{3}(f(X_0) - f(x^*)) + \frac{1}{3T^2} \|X_0 - x^*\|^2.$$

Since f satisfies $A_{\alpha\text{-qg}}$, we have that $\|X_0 - x^*\|^2 \leq \frac{2}{\alpha}(f(X_0) - f(x^*))$. Substituting this in the inequality above results in the statement of the lemma. \blacksquare

E.2.5. PROOF OF LEMMA 4

Proof Corollary 1 and the energy conservation of the Hamiltonian dynamics guarantee Eqs. (5a) and (5c) respectively. It remains to prove Eq. (5b). Lemma 2 states that for $h(t) := \langle X_t - z, Y_t \rangle$,

$$\dot{h}(t) \leq -3(f(X_t) - f(z)) + 2(f(X_0) - f(z)).$$

When $z \leftarrow x^*$, we have $f(X_t) - f(x^*) \geq 0$, and thus from the above, $\dot{h}(t) \leq 2(f(X_0) - f(x^*))$. Integrating both sides gives

$$h(t) - h(0) \leq 2t(f(X_0) - f(x^*)).$$

Since $h(0) = 0$ and $h(t) = \frac{1}{2} \frac{d}{dt} \|X_t - x^*\|^2$, integrating again yields

$$\frac{1}{2} \|X_t - x^*\|^2 - \frac{1}{2} \|X_0 - x^*\|^2 \leq t^2 \cdot (f(X_0) - f(x^*)).$$

Multiplying both sides by $\frac{2}{T^2}(T-t)$ and integrating from $t = 0$ to $t = T$ gives

$$\frac{1}{2} \int_0^T \frac{2(T-t)}{T^2} \|X_t - x^*\|^2 dt - \frac{1}{2} \|X_0 - x^*\|^2 \leq \frac{T^2}{6} (f(X_0) - f(x^*)).$$

By Jensen's inequality,

$$\frac{1}{2} \|X^{\text{avg}}(X_0; T) - x^*\|^2 \leq \frac{1}{2} \int_0^T \frac{2(T-t)}{T^2} \|X_t - x^*\|^2 dt$$

and hence

$$\frac{1}{2} \|X^{\text{avg}}(X_0; T) - x^*\|^2 - \frac{1}{2} \|X_0 - x^*\|^2 \leq \frac{T^2}{6} (f(X_0) - f(x^*))$$

which proves Eq. (5b). The weighted sum $\lambda \cdot \text{Eq. (5a)} + \text{Eq. (5b)} + \lambda^2 \cdot \text{Eq. (5c)}$ (stated below in order for convenience)

$$\begin{aligned} f(X^{\text{avg}}(X_0; T)) - f(x^*) + \frac{1}{3T^2} \|X_T - x^*\|^2 &\leq \frac{2}{3}(f(X_0) - f(x^*)) + \frac{1}{3T^2} \|X_0 - x^*\|^2 \\ \frac{1}{3T^2} \|X^{\text{avg}}(X_0; T) - x^*\|^2 &\leq \frac{1}{9}(f(X_0) - f(x^*)) + \frac{1}{3T^2} \|X_0 - x^*\|^2 \\ f(X_T) - f(x^*) &\leq f(X_0) - f(x^*) \end{aligned}$$

yields **LHS** \leq **RHS** where:

LHS:

$$\lambda \cdot (f(X^{\text{avg}}(X_0; T)) - f(x^*)) + \lambda^2 \cdot (f(X_T) - f(x^*)) \geq \lambda(1 + \lambda) \cdot (f(X^{\text{mix}}(X_0; T)) - f(x^*)),$$

$$\frac{\lambda}{3T^2} \cdot \|X_T - x^*\|^2 + \frac{1}{3T^2} \|X^{\text{avg}}(X_0; T) - x^*\|^2 \geq \frac{(1 + \lambda)}{3T^2} \cdot \|X^{\text{mix}}(X_0; T) - x^*\|^2$$

and the inequalities are due to Jensen's inequality.

RHS: The RHS simplifies to

$$\left(\lambda^2 + \frac{2\lambda}{3} + \frac{1}{9} \right) (f(X_0) - f(x^*)) + \frac{1}{3T^2} (\lambda + 1) \|X_0 - x^*\|^2.$$

Dividing both sides by $\lambda(1 + \lambda)$, we get the claimed result:

$$f(X^{\text{mix}}(X_0; T)) - f(x^*) + \frac{1}{3T^2\lambda} \|X^{\text{mix}}(X_0; T) - x^*\|^2 \leq C_\lambda \cdot (f(X_0) - f(x^*)) + \frac{1}{3T^2\lambda} \|X_0 - x^*\|^2. \quad \blacksquare$$

E.2.6. PROOF OF THEOREM 2 AND COROLLARY 3

Proof of Theorem 2 By definition of X_k , for any $k \in [K]$ it holds that $X_k = X^{\text{mix}}(X_{k-1}; T_k)$. From Lemma 4, we know that

$$\begin{aligned} f(X_k) - f(x^*) + \frac{1}{3T_k^2} \|X_k - x^*\|^2 &= f(X^{\text{mix}}(X_{k-1}; T_k)) - f(x^*) + \frac{1}{3T_k^2} \|X_k - x^*\|^2 \\ &\leq \frac{8}{9} \cdot (f(X_{k-1}) - f(x^*)) + \frac{1}{3T_k^2} \|X_{k-1} - x^*\|^2 \\ &\leq \frac{8}{9} \left\{ f(X_{k-1}) - f(x^*) + \frac{1}{3T_{k-1}^2} \|X_{k-1} - x^*\|^2 \right\} \end{aligned}$$

where we use the condition that $T_k^2 \geq \frac{9}{8} T_{k-1}^2$. Iterating this inequality, we get

$$f(X_K) - f(x^*) \leq \left(\frac{8}{9} \right)^K \left\{ f(X_0) - f(x^*) + \frac{1}{3} \|X_0 - x^*\|^2 \right\}. \quad \blacksquare$$

Proof of Corollary 3 For the specified choice of K , we obtain from Theorem 2 that $f(X_K) - f(x^*) \leq \varepsilon$. The total integration time is bounded by:

$$\begin{aligned} \sum_{k=1}^K T_k &= \frac{3}{3 - \sqrt{8}} \cdot \left(\frac{9^{K/2}}{8} - 1 \right) \leq \frac{3}{3 - \sqrt{8}} \cdot \left(\sqrt{\frac{f(X_0) - f(x^*) + \frac{1}{3} \|X_0 - x^*\|^2}{\varepsilon}} - 1 \right) \\ &\leq 3(3 + \sqrt{8}) \left(\sqrt{\frac{f(X_0) - f(x^*) + \frac{1}{3} \|X_0 - x^*\|^2}{\varepsilon}} \right) \\ &\leq 18 \left(\sqrt{\frac{f(X_0) - f(x^*) + \frac{1}{3} \|X_0 - x^*\|^2}{\varepsilon}} \right). \quad \blacksquare \end{aligned}$$

Appendix F. Extended discussion of Section 4

F.1. Proofs of the statements in Section 4

F.1.1. PROOF OF LEMMA 5

Proof Consider the difference in the Hamiltonian at successive iterations:

$$\begin{aligned}
 & f(x_{n+1}) + \frac{1}{2}\|y_{n+1}\|^2 - f(x_n) - \frac{1}{2}\|y_n\|^2 \\
 & \leq \langle \nabla f(x_{n+1}), x_{n+1} - x_n \rangle + \frac{1}{2}\|y_n - \eta \nabla f(x_{n+1})\|^2 - \frac{1}{2}\|y_n\|^2 \\
 & = \langle \nabla f(x_{n+1}), \eta y_n - \eta^2 \nabla f(x_{n+\frac{1}{2}}) \rangle + \frac{\eta^2}{2}\|\nabla f(x_{n+1})\|^2 - \eta \langle y_n, \nabla f(x_{n+1}) \rangle \\
 & = -\eta^2 \langle \nabla f(x_{n+1}), \nabla f(x_{n+\frac{1}{2}}) \rangle + \frac{\eta^2}{2}\|\nabla f(x_{n+1})\|^2 \\
 & = \frac{\eta^2}{2} \left(\|\nabla f(x_{n+1}) - \nabla f(x_{n+\frac{1}{2}})\|^2 - \|\nabla f(x_{n+\frac{1}{2}})\|^2 \right).
 \end{aligned}$$

Since $x_{n+1} = x_{n+\frac{1}{2}} - \eta^2 \nabla f(x_{n+\frac{1}{2}})$, we have by smoothness of f that

$$\|\nabla f(x_{n+1}) - \nabla f(x_{n+\frac{1}{2}})\|^2 \leq L^2 \|x_{n+1} - x_{n+\frac{1}{2}}\|^2 = L^2 \eta^4 \|\nabla f(x_{n+\frac{1}{2}})\|^2.$$

When $\eta \leq \frac{1}{\sqrt{L}}$,

$$f(x_{n+1}) + \frac{1}{2}\|y_{n+1}\|^2 - f(x_n) - \frac{1}{2}\|y_n\|^2 \leq \frac{\eta^2(L^2\eta^4 - 1)}{2} \|\nabla f(x_{n+\frac{1}{2}})\|^2 \leq 0.$$

■

F.1.2. PROOF OF LEMMA 6

To prove Lemma 6, we first discuss some parallels between the continuous-time and discrete-time analysis that we highlight to provide intuition.

Table 2: Parallels between continuous-time and discrete-time analysis.

	Continuous time	Discrete time
Evolution rule	Hamiltonian flow (HF)	Extragradient Integrator (HF ^{extg})
Hamiltonian property	Conservation (Lemma 1)	Dissipation (Lemma 5)
Boundary term	$h_t = \langle X_t - z, Y_t \rangle$	$h_n = \langle x_n - z, y_n \rangle$
Strategy	Bound on \dot{h}_t in terms of optimality gap (Lemma 2)	Bound on $h_{n+1} - h_n$ in terms of optimality gap (Lemma 15)
Average	$X^{\text{avg}}(X_0; T) = \frac{2}{T^2} \int_0^T (T-t) X_t dt$	$x^{\text{avg}}(x_0; N) = \frac{2}{N(N+1)} \sum_{n=1}^N (N-n+1) x_n$

Lemma 15 Assume f satisfies A_{cvx} and A_{L-sm} . Assume f is convex and L -smooth. Let $\{(x_n, y_n)\}_{n \geq 1}$ be generated by the extragradient integrator (HF^{extg}) from any initial $x_0 \in \mathbb{R}^d$ and $y_0 = \mathbf{0}$, and step size $\eta \leq \frac{1}{\sqrt{L}}$. Then, for any $N \geq 1$ and reference point $z \in \mathbb{R}^d$, the following holds:

$$\langle x_N - z, y_N \rangle \leq 2\eta N(f(x_0) - f(z)) - 3\eta \sum_{n=1}^N (f(x_n) - f(z)).$$

We prove this lemma in [Appendix F.1.6](#). [Lemma 6](#) builds on this lemma, proven below.

Proof of Lemma 6 We begin by bounding the difference $\Delta_n := \|x_{n+1} - z\|^2 - \|x_n - z\|^2$, and recall that $h_n = \langle x_n - z, y_n \rangle$. Using the relation between x_{n+1}, y_{n+1} from x_n, y_n as given in HF^{extg} , we have

$$\begin{aligned} \Delta_n &= \|x_{n+1} - x_n\|^2 + 2\langle x_{n+1} - x_n, x_n - z \rangle \\ &= \|\eta y_n - \eta^2 \nabla f(x_{n+\frac{1}{2}})\|^2 + 2\langle \eta y_n - \eta^2 \nabla f(x_{n+\frac{1}{2}}), x_n - z \rangle \\ &= 2\eta h_n - 2\eta^2 \langle \nabla f(x_{n+\frac{1}{2}}), x_n - z \rangle - 2\eta^3 \langle y_n, \nabla f(x_{n+\frac{1}{2}}) \rangle + \eta^2 \|y_n\|^2 + \eta^4 \|\nabla f(x_{n+\frac{1}{2}})\|^2 \\ &= 2\eta h_n - 2\eta^2 \langle \nabla f(x_{n+\frac{1}{2}}), x_n + \eta y_n - z \rangle + \eta^2 \|y_n\|^2 + \eta^4 \|\nabla f(x_{n+\frac{1}{2}})\|^2 \\ &= 2\eta h_n - 2\eta^2 \langle \nabla f(x_{n+\frac{1}{2}}), x_{n+\frac{1}{2}} - z \rangle + \eta^2 \|y_n\|^2 + \eta^4 \|\nabla f(x_{n+\frac{1}{2}})\|^2 \\ &\leq 2\eta h_n + 2\eta^2 (f(z) - f(x_{n+\frac{1}{2}})) + \eta^2 \|y_n\|^2 + \eta^4 \|\nabla f(x_{n+\frac{1}{2}})\|^2, \end{aligned}$$

where the final inequality above uses the convexity of f . Since $x_{n+1} = x_{n+\frac{1}{2}} - \eta^2 \nabla f(x_{n+\frac{1}{2}})$, by smoothness of f ,

$$f(x_{n+1}) \leq f(x_{n+\frac{1}{2}}) - \eta^2 \left(1 - \frac{L\eta^2}{2}\right) \|\nabla f(x_{n+\frac{1}{2}})\|^2.$$

When $\eta \leq \frac{1}{\sqrt{L}}$, this implies

$$\frac{\eta^2}{2} \|\nabla f(x_{n+\frac{1}{2}})\|^2 \leq \eta^2 \left(1 - \frac{L\eta^2}{2}\right) \|\nabla f(x_{n+\frac{1}{2}})\|^2 \leq f(x_{n+\frac{1}{2}}) - f(x_{n+1}).$$

Substituting this bound and using [Lemma 5](#) to write $\|y_n\|^2 \leq 2(f(x_0) - f(x_n))$, we get

$$\begin{aligned} \Delta_n &\leq 2\eta h_n + 2\eta^2 (f(z) - f(x_{n+\frac{1}{2}})) + 2\eta^2 \cdot (f(x_0) - f(x_n)) + 2\eta^2 (f(x_{n+\frac{1}{2}}) - f(x_{n+1})) \\ &= 2\eta h_n + 2\eta^2 (f(x_0) - f(x_n)) + 2\eta^2 (f(z) - f(x_{n+1})). \end{aligned} \tag{10}$$

Recall [Lemma 15](#) guarantees that

$$\begin{aligned} h_n &\leq 2\eta n (f(x_0) - f(z)) - 3\eta \sum_{n'=1}^n (f(x_{n'}) - f(z)) \\ &= 2(n+1)\eta (f(x_0) - f(z)) - 3\eta \sum_{n'=1}^{n+1} (f(x_{n'}) - f(z)) + \eta (3f(x_{n+1}) - 2f(x_0) - f(z)). \end{aligned}$$

Substituting this in the bound for Δ_n , we get

$$\begin{aligned}
 \Delta_n &\leq 2\eta h_n + 2\eta^2(f(x_0) - f(x_n)) + 2\eta^2(f(z) - f(x_{n+1})) \\
 &= 4\eta^2(n+1)(f(x_0) - f(z)) - 6\eta^2 \sum_{n'=1}^{n+1} (f(x_{n'}) - f(z)) + 2\eta^2(3f(x_{n+1}) - 2f(x_0) - f(z)) \\
 &\quad + 2\eta^2(f(x_0) - f(x_n)) + 2\eta^2(f(z) - f(x_{n+1})) \\
 &= 4\eta^2(n+1)(f(x_0) - f(z)) - 6\eta^2 \sum_{n'=1}^{n+1} (f(x_{n'}) - f(z)) + 2\eta^2(f(x_{n+1}) - f(x_n)) \\
 &\quad + 2\eta^2(f(x_{n+1}) - f(x_0)) \\
 &\leq 4\eta^2(n+1)(f(x_0) - f(z)) - 6\eta^2 \sum_{n'=1}^{n+1} (f(x_{n'}) - f(z)) + 2\eta^2(f(x_{n+1}) - f(x_n)),
 \end{aligned}$$

where the last inequality is due to [Lemma 5](#). We now sum this inequality from $n = 0$ to $n = N - 1$:

$$\begin{aligned}
 &\|x_N - z\|^2 - \|x_0 - z\|^2 \\
 &= \sum_{n=0}^{N-1} \Delta_n \\
 &\leq \sum_{n=0}^{N-1} 4\eta^2(n+1)(f(x_0) - f(z)) - 6\eta^2 \sum_{n=0}^{N-1} \sum_{n'=1}^{n+1} (f(x_{n'}) - f(z)) + 2\eta^2 \sum_{n=0}^{N-1} (f(x_{n+1}) - f(x_n)) \\
 &= \sum_{n=1}^N 4\eta^2 n(f(x_0) - f(z)) - 6\eta^2 \sum_{n=1}^N \sum_{n'=1}^n (f(x_{n'}) - f(z)) + 2\eta^2(f(x_N) - f(x_0)) \\
 &= 2N(N+1)\eta^2(f(x_0) - f(z)) - 6\eta^2 \sum_{n=1}^N (N-n+1)(f(x_n) - f(z)) + 2\eta^2(f(x_N) - f(x_0)) \\
 &\leq 2N(N+1)\eta^2(f(x_0) - f(z)) - 6\eta^2 \sum_{n=1}^N (N-n+1)(f(x_n) - f(z))
 \end{aligned}$$

where the last inequality again uses [Lemma 5](#). Dividing both sides by $3\eta^2 N(N+1)$ and rearranging terms, we get

$$\begin{aligned}
 &\frac{2}{N(N+1)} \sum_{n=1}^N (N-n+1)(f(x_n) - f(z)) + \frac{1}{3\eta^2 N(N+1)} \|x_N - z\|^2 \\
 &\leq \frac{2}{3}(f(x_0) - f(z)) + \frac{1}{3\eta^2 N(N+1)} \|x_0 - z\|^2.
 \end{aligned}$$

Since $\frac{2}{N(N+1)} \sum_{n=1}^N (N-n+1) = 1$, we have from Jensen's inequality and the definition of $x^{\text{avg}}(x_0; N)$ that

$$f(x^{\text{avg}}(x_0; N)) - f(z) + \frac{1}{3\eta^2 N(N+1)} \|x_N - z\|^2 \leq \frac{2}{3}(f(x_0) - f(z)) + \frac{1}{3\eta^2 N(N+1)} \|x_0 - z\|^2.$$

■

F.1.3. PROOFS OF [THEOREM 3](#) AND [COROLLARY 4](#)

Proof of [Theorem 3](#) Recall that x_k produced by [dHFA-eg](#) satisfies $x_k = x^{\text{avg}}(x_{k-1}; N_k)$. From [Lemma 6](#),

$$\begin{aligned} f(x_k) - f(x^*) &= f(x^{\text{avg}}(x_{k-1}; N_k)) - f(x^*) \\ &\leq \frac{2}{3}(f(x_{k-1}) - f(x^*)) + \frac{1}{3\eta^2 N_k(N_k + 1)} \|x_{k-1} - x^*\|^2 \\ &\stackrel{(a)}{\leq} \left(\frac{2}{3} + \frac{2}{3\eta^2 \alpha N_k(N_k + 1)} \right) (f(x_{k-1}) - f(x^*)) \\ &\stackrel{(b)}{\leq} \left(\frac{2}{3} + \frac{2}{3c^2} \right) \cdot (f(x_{k-1}) - f(x^*)). \end{aligned}$$

In step (a), we use the fact that f satisfies [A \$_{\alpha}\$ -gg](#), and in step (b) we use the condition that $N_k \geq \lceil \frac{c}{\eta\sqrt{\alpha}} \rceil \geq \frac{c}{\eta\sqrt{\alpha}}$. Iterating this inequality,

$$f(x_K) - f(x^*) \leq \left(\frac{2}{3} + \frac{2}{3c^2} \right)^K \cdot (f(x_0) - f(x^*)).$$

■

Proof of [Corollary 4](#) Since $N_k = \lceil \frac{5}{2}\sqrt{k} \rceil \geq \frac{5}{2}\sqrt{k}$, from [Theorem 3](#) we have

$$f(x_K) - f(x^*) \leq \left(\frac{58}{75} \right)^K \cdot (f(x_0) - f(x^*)).$$

Since $K = \lceil 4 \log \frac{f(x_0) - f(x^*)}{\varepsilon} \rceil \geq (\log \frac{75}{58})^{-1} \frac{f(x_0) - f(x^*)}{\varepsilon}$,

$$f(x_K) - f(x^*) \leq \varepsilon.$$

Since at each iteration of the extragradient integrator we use 2 gradient calls, we have the total gradient complexity as

$$2 \cdot \left\lceil \frac{5}{2} \sqrt{\frac{L}{\alpha}} \right\rceil \cdot \left\lceil 4 \log \frac{f(x_0) - f(x^*)}{\varepsilon} \right\rceil.$$

■

 F.1.4. PROOF OF [LEMMA 7](#)

Proof From [Eq. \(10\)](#), we have that $\Delta_n := \|x_{n+1} - x^*\|^2 - \|x_n - x^*\|^2$ satisfies

$$\Delta_{n'} \leq 2\eta h_{n'} + 2\eta^2 (f(x_0) - f(x_{n'})).$$

Setting $z = x^*$ in [Lemma 15](#), we could bound $h_{n'}$ using:

$$\langle x_{n'} - x^*, y_{n'} \rangle \leq 2\eta n' (f(x_0) - f(x^*)) - 3\eta \sum_{n=1}^{n'} (f(x_n) - f(x^*)) \leq 2\eta n' (f(x_0) - f(x^*)).$$

Plugging this bound into the upper bound for $\Delta_{n'}$ gives:

$$\Delta_{n'} \leq 4\eta^2 n' (f(x_0) - f(x^*)) + 2\eta^2 (f(x_0) - f(x_{n'})) \leq 4\eta^2 n' (f(x_0) - f(x^*)) + 2\eta^2 (f(x_0) - f(x^*)).$$

Summing both sides from $n' = 0$ to $n - 1$,

$$\|x_n - x^*\|^2 - \|x_0 - x^*\|^2 \leq 2\eta^2 n^2 (f(x_0) - f(x^*)).$$

Multiplying both sides by $N - n + 1$ and summing from $n = 1$ to $n = N$, we get

$$\sum_{n=1}^N (N - n + 1) \|x_n - x^*\|^2 \leq \frac{N(N+1)}{2} \|x_0 - x^*\|^2 + \frac{\eta^2 N(N+1)^2(N+2)}{6} (f(x_0) - f(x^*)).$$

Dividing both sides by $N(N+1)$, we get

$$\frac{2}{N(N+1)} \sum_{n=1}^N (N - n + 1) \frac{\|x_n - x^*\|^2}{2} \leq \frac{\|x_0 - x^*\|^2}{2} + \frac{\eta^2 (N+1)(N+2)}{6} (f(x_0) - f(x^*)).$$

Note that $\sum_{n=1}^N (N - n + 1) = \frac{N(N+1)}{2}$, and therefore by Jensen's inequality and the definition of $x^{\text{avg}}(x_0; N)$ we get

$$\frac{1}{2} \|x^{\text{avg}}(x_0; N) - x^*\|^2 \leq \frac{1}{2} \|x_0 - x^*\|^2 + \frac{\eta^2 (N+1)(N+2)}{6} (f(x_0) - f(x^*)).$$

Since $(N+1)(N+2) \leq \frac{3}{2}N(N+1)$ for $N \geq 4$, we have

$$\begin{aligned} \frac{1}{2} \|x^{\text{avg}}(x_0; N) - x^*\|^2 &\leq \frac{1}{2} \|x_0 - x^*\|^2 + \frac{\eta^2 N(N+1)}{4} (f(x_0) - f(x^*)) \\ \frac{1}{3\eta^2 N(N+1)} \|x^{\text{avg}}(x_0; N) - x^*\|^2 &\leq \frac{1}{3\eta^2 N(N+1)} \|x_0 - x^*\|^2 + \frac{1}{6} (f(x_0) - f(x^*)). \end{aligned} \quad (11)$$

Like in the proof of [Lemma 4](#), we consider a weighted sum of the following inequalities:

$$f(x^{\text{avg}}(x_0; N)) - f(x^*) + \frac{\|x_N - x^*\|^2}{3\eta^2 N(N+1)} \leq \frac{2}{3} (f(x_0) - f(x^*)) + \frac{\|x_0 - x^*\|^2}{3\eta^2 N(N+1)} \quad (\text{Eq. (7)})$$

$$\frac{1}{3\eta^2 N(N+1)} \|x^{\text{avg}}(x_0; N) - x^*\|^2 \leq \frac{1}{3\eta^2 N(N+1)} \|x_0 - x^*\|^2 + \frac{1}{6} (f(x_0) - f(x^*)) \quad (\text{Eq. (11)})$$

$$f(x_N) - f(x^*) \leq f(x_0) - f(x^*) \quad (\text{Eq. (6)})$$

with weights $\lambda, 1, \lambda^2$ respectively. This yields **LHS** \leq **RHS** where:

LHS:

$$\begin{aligned} &\lambda \cdot (f(x^{\text{avg}}(x_0; N)) - f(x^*)) + \lambda^2 \cdot (f(x_N) - f(x^*)) \\ &+ \frac{\lambda}{3\eta^2 N(N+1)} \|x_N - x^*\|^2 + \frac{1}{3\eta^2 N(N+1)} \|x^{\text{avg}}(x_0; N) - x^*\|^2 \\ &\geq \lambda(1 + \lambda) \cdot (f(x^{\text{mix}}(x_0; N)) - f(x^*)) + \frac{(1 + \lambda)}{3\eta^2 N(N+1)} \cdot \|x^{\text{mix}}(x_0; N) - x^*\|^2. \end{aligned}$$

RHS:

$$\left(\lambda^2 + \frac{2\lambda}{3} + \frac{1}{6}\right)(f(x_0) - f(x^*)) + \frac{1 + \lambda}{3\eta^2 N(N+1)} \|x_0 - x^*\|^2.$$

After dividing both sides by $\lambda(1 + \lambda)$, this finally returns

$$\begin{aligned} f(x^{\text{mix}}(x_0; N)) - f(x^*) + \frac{1}{3\lambda\eta^2 N(N+1)} \|x^{\text{mix}}(x_0; N) - x^*\|^2 \\ \leq c_\lambda(f(x_0) - f(x^*)) + \frac{1}{3\lambda\eta^2 N(N+1)} \|x_0 - x^*\|^2 \end{aligned}$$

which is precisely the statement of the lemma. ■

F.1.5. PROOFS OF THEOREM 4 AND COROLLARY 5

Proof of Theorem 4 Note that for any $k \geq 1$, $x_k = x^{\text{mix}}(x_{k-1}; N_k)$ with $\lambda = \lambda^* = \frac{\sqrt{3+1}}{2}$. Consequently, from Lemma 7, we have

$$\begin{aligned} f(x_k) - f(x^*) + \frac{1}{3\lambda^*\eta^2 N_k(N_k+1)} \|x_k - x^*\|^2 \\ = f(x^{\text{mix}}(x_{k-1}; N_k)) - f(x^*) + \frac{1}{3\lambda^*\eta^2 N_k(N_k+1)} \|x^{\text{mix}}(x_{k-1}; N_k) - x^*\|^2 \\ \leq c_{\lambda^*}(f(x_{k-1}) - f(x^*)) + \frac{1}{3\lambda^*\eta^2 N_k(N_k+1)} \|x_{k-1} - x^*\|^2 \\ = c_{\lambda^*} \left\{ f(x_{k-1}) - f(x^*) + \frac{1}{3\lambda^*\eta^2 c_{\lambda^*} N_k(N_k+1)} \|x_{k-1} - x^*\|^2 \right\} \\ \leq c_{\lambda^*} \left\{ f(x_{k-1}) - f(x^*) + \frac{1}{3\lambda^*\eta^2 N_{k-1}(N_{k-1}+1)} \|x_{k-1} - x^*\|^2 \right\}. \end{aligned}$$

The final inequality uses the fact that $N_k(N_k+1) \geq \frac{1}{c_{\lambda^*}} N_{k-1}(N_{k-1}+1)$. Iterating this from $k = 1$ to $k = K$, we get

$$f(x_K) - f(x^*) \leq \left(\frac{\sqrt{3+1}}{3}\right)^K \left\{ f(x_0) - f(x^*) + \frac{1}{3\lambda^*\eta^2 N_0(N_0+1)} \|x_0 - x^*\|^2 \right\}.$$

Setting $N_0 = 4$ and $\lambda^* = \frac{\sqrt{3+1}}{2}$ completes the proof. ■

Proof of Corollary 5 We first show that the desired inequality between N_k and N_{k+1} holds under the specified $\{N_k\}$ sequence. Define $q = 1/\sqrt{c_\lambda^*} = \sqrt{\frac{3}{1+\sqrt{3}}} \leq 2$. By properties of $\lceil \cdot \rceil$, we calculate that

$$\begin{aligned} N_{k+1}^2 + N_{k+1} &= \lceil qN_k + \frac{1}{2} \rceil^2 + \lceil qN_k + \frac{1}{2} \rceil \\ &\geq (qN_k + \frac{3}{2})(qN_k + \frac{1}{2}) \\ &= q^2 N_k^2 + 2qN_k + \frac{3}{4} \\ &\geq q^2(N_k^2 + N_k), \end{aligned}$$

where the last inequality followed because $q \leq 2$. Additionally, note that

$$\begin{aligned} K &\geq \left[\left(\log \frac{3}{1 + \sqrt{3}} \right)^{-1} \cdot \log \frac{\frac{L}{2} \|x_0 - x^*\|^2 + \frac{(\sqrt{3}-1)L}{60} \|x_0 - x^*\|^2}{\varepsilon} \right] \\ &\geq \left(\log \frac{3}{1 + \sqrt{3}} \right)^{-1} \cdot \log \frac{f(x_0) - f(x^*) + \frac{1}{3\lambda^*\eta^2 N_0(N_0+1)} \|x_0 - x^*\|^2}{\varepsilon}. \end{aligned}$$

Therefore, from [Theorem 4](#),

$$f(x_K) - f(x^*) \leq \varepsilon.$$

To obtain the number of gradient evaluations, recall that by definition, $N_{k+1} \leq qN_k + \frac{3}{2}$. Consequently,

$$N_k \leq q^k N_0 + \frac{3}{2} \left(\frac{q^k - 1}{q - 1} \right) \leq q^k \left(N_0 + \frac{3}{2(q-1)} \right).$$

Then, summing both sides from $k = 1$ to $k = K$,

$$\sum_{k=1}^K N_k \leq \left(\frac{N_0 q}{q-1} + \frac{3q}{2(q-1)^2} \right) (q^K - 1) \leq \left(\frac{N_0 q}{q-1} + \frac{3q}{2(q-1)^2} \right) q^K.$$

The quantity q^K can be simplified as

$$\begin{aligned} q^K &= \exp(K \log q) \\ &\leq \exp \left(\frac{1}{2} \log \left(\frac{3}{1 + \sqrt{3}} \right) \cdot \left(2 \log \left(\frac{3}{1 + \sqrt{3}} \right)^{-1} \cdot \log \left(\sqrt{\frac{L}{\varepsilon}} \|x_0 - x^*\| \right) + 1 \right) \right) \\ &= \sqrt{\frac{L}{\varepsilon}} \|x_0 - x^*\| \cdot q. \end{aligned}$$

Substituting the values for N_0, q , the constant $\frac{N_0 q^2}{q-1} + \frac{3q^2}{2(q-1)^2} \leq 810$, which completes the proof. \blacksquare

F.1.6. PROOF OF [LEMMA 15](#)

Proof Denote $h_n = \langle x_n - z, y_n \rangle$. We first compute the difference

$$\begin{aligned} h_{n+1} - h_n &= \langle x_{n+1} - z, y_{n+1} \rangle - \langle x_n - z, y_n \rangle \\ &= \langle x_{n+1} - z, y_{n+1} - y_n \rangle + \langle x_{n+1} - z, y_n \rangle - \langle x_n - z, y_n \rangle \\ &= -\eta \langle x_{n+1} - z, \nabla f(x_{n+1}) \rangle + \langle x_{n+1} - x_n, y_n \rangle. \end{aligned}$$

For the first term on the right hand side, we use the convexity of f :

$$-\eta \langle x_{n+1} - z, \nabla f(x_{n+1}) \rangle \leq \eta (f(z) - f(x_{n+1})).$$

For the second term on the right hand side, we additionally use the smoothness of f and the form of the extragradient integrator to get

$$\langle x_{n+1} - x_n, y_n \rangle = \langle \eta y_n - \eta^2 \nabla f(x_{n+\frac{1}{2}}), y_n \rangle$$

$$\begin{aligned}
 &= \eta \|y_n\|^2 - \eta \langle \nabla f(x_{n+\frac{1}{2}}), x_{n+\frac{1}{2}} - x_n \rangle \\
 &\stackrel{(a)}{\leq} 2\eta(f(x_0) - f(x_n)) + \eta(f(x_n) - f(x_n + \eta y_n)) \\
 &\leq 2\eta f(x_0) - \eta f(x_n) - \eta f(x_{n+\frac{1}{2}}) \\
 &\stackrel{(b)}{\leq} 2\eta f(x_0) - \eta f(x_n) - \eta f(x_{n+1}).
 \end{aligned}$$

Steps (a) and (b) are due to the choice of $\eta \leq \frac{1}{\sqrt{L}}$ from [Lemma 5](#) and the fact that $f(x_{n+1}) \leq f(x_{n+\frac{1}{2}})$ respectively. Combining the bounds, we get

$$h_{n+1} - h_n \leq 2\eta(f(x_0) - f(z)) - \eta(f(x_n) - f(z)) - 2\eta(f(x_{n+1}) - f(z)).$$

Summing both sides from $n = 0$ to $n = N - 1$ and noting that $h_0 = 0$,

$$\begin{aligned}
 h_N &\leq 2\eta N(f(x_0) - f(z)) - 3\eta \sum_{n=1}^{N-1} (f(x_n) - f(z)) - 2\eta(f(x_N) - f(z)) - \eta(f(x_0) - f(z)) \\
 &\leq 2\eta N(f(x_0) - f(z)) - 3\eta \sum_{n=1}^N (f(x_n) - f(z))
 \end{aligned}$$

where in the final inequality we use the fact that $f(x_N) \leq f(x_0)$. ■

F.2. Discretized Hamiltonian Flow for optimization with Averaging using **HF^{imp}**

Here, we discuss how the implicit integrator is also capable of achieving accelerated rates. As discussed previously, the implicit integrator assumes access to the proximal oracle for f . The resulting algorithm **dHFA-im** is provided in [Algorithm 6](#). By [Lemma 12](#), when f is convex, the iterates generated by the implicit integrator produce a sequence of non-increasing Hamiltonians, mirroring [Lemma 5](#); notably, this holds without requiring the smoothness of f . As a consequence of this monotonicity property, the iterates generated by the implicit integrator also satisfies the same guarantee as in [Eq. \(7\)](#); see the lemma below, proven in [Appendix F.2.1](#).

Lemma 16 *Assume that f satisfies \mathbf{A}_{cvx} , and let $\{(x_n, y_n)\}_{n \geq 1}$ be the iterates of the implicit integrator (**HF^{imp}**) with any step size $\eta > 0$, starting from $(x_0, \mathbf{0})$. Then, for any $N \geq 1$ and reference point $z \in \mathbb{R}^d$*

$$f(x^{\text{avg}}(x_0; N)) - f(z) + \frac{1}{3\eta^2 N(N+1)} \|x_N - z\|^2 \leq \frac{2}{3}(f(x_0) - f(z)) + \frac{1}{3\eta^2 N(N+1)} \|x_0 - z\|^2. \quad (12)$$

Now, we state the analogues of [Theorem 3](#) and [Corollary 4](#) for **dHFA-im**.

Theorem 5 *Assume that f satisfies \mathbf{A}_{cvx} and $\mathbf{A}_{\alpha\text{-qg}}$. Suppose x_K is the output of running **dHFA-im** for K iterations with parameter $\lambda = 0$, any step size $\eta > 0$, and discretization steps sequence satisfying $N_k = N \geq \lceil \frac{c}{\eta\sqrt{\alpha}} \rceil$ for all $k \in [K]$ and initialization $x_0 \in \mathbb{R}^d$. Then,*

$$f(x_K) - f(x^*) \leq \left(\frac{2}{3} + \frac{2}{3c^2} \right)^K \cdot (f(x_0) - f(x^*)).$$

Algorithm 6 Discretized Hamiltonian Flow for optimization with Aggregation using implicit integrator (dHFA-im)

Input : Initialization: $x_0 \in \mathbb{R}^d$, Number of iterations: $K \in \mathbb{N}$, parameter: $\lambda \in \mathbb{R}_{\geq 0}$, step size $\eta > 0$, discretization steps sequence: $\{N_1, \dots, N_K\} \in \mathbb{N}^K$, oracle for implicit integrator

HFimp

for $k = 1$ **to** K **do**

Obtain sequence $\{x_n^{(k)}, y_n^{(k)}\}_{n=1}^{N_k}$ using **HFimp** from $(x_0^{(k)}, y_0^{(k)}) = (x_{k-1}, \mathbf{0})$.

Compute $x^{\text{avg}}(x_{k-1}; N_k) = \frac{2}{N_k(N_k+1)} \sum_{n=1}^{N_k} (N_k - n + 1)x_n^{(k)}$.

Set $x_k = \frac{1}{\lambda+1} x^{\text{avg}}(x_{k-1}; N_k) + \frac{\lambda}{\lambda+1} x_{N_k}^{(k)}$.

end

return x_K

Proof Recall that x_k produced by **dHFA-im** satisfies $x_k = x^{\text{avg}}(x_{k-1}; N_k)$. From [Lemma 16](#),

$$\begin{aligned} f(x_k) - f(x^*) &= f(x^{\text{avg}}(x_{k-1}; N_k)) - f(x^*) \\ &\leq \frac{2}{3} (f(x_{k-1}) - f(x^*)) + \frac{1}{3\eta^2 N_k(N_k+1)} \|x_{k-1} - x^*\|^2 \\ &\stackrel{(a)}{\leq} \left(\frac{2}{3} + \frac{2}{3\eta^2 \alpha N_k(N_k+1)} \right) (f(x_{k-1}) - f(x^*)) \\ &\stackrel{(b)}{\leq} \left(\frac{2}{3} + \frac{2}{3c^2} \right) \cdot (f(x_{k-1}) - f(x^*)). \end{aligned}$$

In step (a), we use the fact that f satisfies $\mathbf{A}_{\alpha\text{-qg}}$, and in step (b) we use the condition that $N_k \geq \lceil \frac{c}{\eta\sqrt{\alpha}} \rceil \geq \frac{c}{\eta\sqrt{\alpha}}$. Iterating this inequality,

$$f(x_K) - f(x^*) \leq \left(\frac{2}{3} + \frac{2}{3c^2} \right)^K \cdot (f(x_0) - f(x^*)).$$

■

Corollary 8 Assume that f satisfies \mathbf{A}_{cvx} and $\mathbf{A}_{\alpha\text{-qg}}$. To obtain x_K that is a ε -accurate minimizer, it suffices to run **dHFA-im** with parameter $\lambda = 0$ and any step size $\eta > 0$ for K iterations and discretization step sequence $\{N_k\}_{k=1}^K$ with

$$K = \left\lceil 4 \log \frac{f(x_0) - f(x^*)}{\varepsilon} \right\rceil; \quad N_k = \left\lceil \frac{5}{2\eta\sqrt{\alpha}} \right\rceil \quad \forall k \in [K].$$

The total number of calls to the implicit discretization oracle is

$$N_{\text{tot}} = \left\lceil \frac{5}{2\eta\sqrt{\alpha}} \right\rceil \cdot \left\lceil 4 \log \frac{f(x_0) - f(x^*)}{\varepsilon} \right\rceil.$$

Proof Since $N_k = \lceil \frac{5}{2\eta\sqrt{\alpha}} \rceil \geq \frac{5}{2\eta\sqrt{\alpha}}$, from [Theorem 5](#) we have

$$f(x_K) - f(x^*) \leq \left(\frac{58}{75} \right)^K \cdot (f(x_0) - f(x^*)).$$

$$\text{Since } K = \left\lceil 4 \log \frac{f(x_0) - f(x^*)}{\varepsilon} \right\rceil \geq \left(\log \frac{75}{58} \right)^{-1} \frac{f(x_0) - f(x^*)}{\varepsilon},$$

$$f(x_K) - f(x^*) \leq \varepsilon.$$

Therefore, the total number of calls to the implicit discretization oracle is

$$\left\lceil \frac{5}{2\eta\sqrt{\alpha}} \right\rceil \cdot \left\lceil 4 \log \frac{f(x_0) - f(x^*)}{\varepsilon} \right\rceil.$$

■

To obtain the result for minimizing convex functions, we first state the analogous version of [Lemma 7](#) for [HF^{imp}](#). A proof is provided in [Appendix F.2.2](#).

Lemma 17 *Assume that f satisfies [A_{cvx}](#). Let $\{(x_n, y_n)\}_{n \geq 1}$ be generated by the implicit integrator with step size $\eta > 0$. For any $N \geq 4$ and initial position x_0 , define*

$$x^{\text{mix}}(x_0; N) = \frac{\lambda}{\lambda + 1} x_N + \frac{1}{1 + \lambda} x^{\text{avg}}(x_0; N).$$

Then, defining $c_\lambda = \frac{(6\lambda^2 + 4\lambda + 1)}{6\lambda(\lambda + 1)}$, we have

$$f(x^{\text{mix}}(x_0; N)) - f(x^*) + \frac{\|x^{\text{mix}}(x_0; N) - x^*\|^2}{3\lambda\eta^2 N(N + 1)} \leq c_\lambda (f(x_0) - f(x^*)) + \frac{\|x_0 - x^*\|^2}{3\lambda\eta^2 N(N + 1)}.$$

Since the conclusion of the [Lemma 17](#) is the same as the conclusion of [Lemma 7](#), and [dHFA-im](#) is algorithmically the same as [dHFA-eg](#) with just [HF^{imp}](#) in lieu of [HF^{extg}](#), we have the following theorem and corollary for minimizing convex f , analogous to [Theorem 4](#) and [Corollary 5](#). We state them without proof, as they can be derived from [Lemma 17](#) arguments similar to that in [Appendix F.1.5](#).

Theorem 6 *Assume that f satisfies [A_{cvx}](#). Suppose x_K is the output of running [dHFA-im](#) for K iterations with parameter $\lambda = \frac{\sqrt{3} + 1}{2}$, any step size $\eta > 0$, and discretization steps sequence $\{N_k\}_{k=1}^K$ satisfying $N_k(N_k + 1) \geq \frac{3}{\sqrt{3} + 1} N_{k-1}(N_{k-1} + 1)$ for all $k \in [K]$ with $N_0 = 4$. Then,*

$$f(x_K) - f(x^*) \leq \left(\frac{\sqrt{3} + 1}{3} \right)^K \left(f(x_0) - f(x^*) + \frac{\sqrt{3} - 1}{60\eta^2} \|x_0 - x^*\|^2 \right).$$

Corollary 9 *Assume that f satisfies [A_{cvx}](#). To obtain x_K that is a ε -accurate minimizer, it suffices to run [dHFA-im](#) with parameter $\lambda = \frac{\sqrt{3} + 1}{2}$ and any step size $\eta > 0$ for K iterations and discretization step sequence $\{N_k\}_{k=1}^K$ such that*

$$K = \left\lceil \left(\log \left(\frac{3}{1 + \sqrt{3}} \right) \right)^{-1} \cdot \log \left(\frac{f(x_0) - f(x^*) + \frac{\sqrt{3} - 1}{60\eta^2} \|x_0 - x^*\|^2}{\varepsilon} \right) \right\rceil,$$

$$N_k = \left\lceil \sqrt{\frac{3}{1 + \sqrt{3}} N_{k-1} + \frac{1}{2}} \right\rceil; \quad N_0 = 4.$$

The total number of numerical integration steps satisfies

$$N_{\text{tot}} \leq 810 \sqrt{\frac{f(x_0) - f(x^*) + \frac{\sqrt{3} - 1}{60\eta^2} \|x_0 - x^*\|^2}{\varepsilon}}.$$

F.2.1. PROOF OF LEMMA 16

Similar to the proof of Lemma 6, we first begin with following helper lemma whose proof is provided in Appendix F.2.3.

Lemma 18 *Assume that f satisfies \mathbf{A}_{cvx} . Let $\{(x_n, y_n)\}_{n \geq 1}$ be generated by the implicit integrator (\mathbf{HF}^{imp}) from any initial $x_0 \in \mathbb{R}^d$ and $y_0 = \mathbf{0}$. Then, for any $N \geq 1$, the following holds:*

$$\langle x_N - z, y_N \rangle \leq 2\eta N(f(x_0) - f(z)) - 3\eta \sum_{n=1}^N (f(x_n) - f(z)).$$

Proof of Lemma 16 We adopt the strategy to prove Lemma 6 and consider the difference $\Delta_n := \|x_{n+1} - z\|^2 - \|x_n - z\|^2$. By definition of \mathbf{HF}^{imp} , x_{n+1}, y_{n+1} from x_n, y_n

$$\begin{aligned} \Delta_n - 2\eta h_n &= \|x_{n+1} - z\|^2 - \|x_n - z\|^2 - 2\eta \langle x_n - z, y_n \rangle \\ &= \|x_{n+1} - x_n\|^2 + 2\langle x_{n+1} - x_n, x_n - z \rangle - 2\eta \langle x_n - z, y_n \rangle \\ &= \eta^2 \|y_{n+1}\|^2 + 2\eta \langle y_{n+1} - y_n, x_n - z \rangle \\ &= \eta^2 \|y_{n+1}\|^2 - 2\eta^2 \langle \nabla f(x_{n+1}), x_n - z \rangle \\ &= \eta^2 \|y_{n+1}\|^2 - 2\eta^2 \langle \nabla f(x_{n+1}), x_{n+1} - z \rangle - 2\eta^2 \langle \nabla f(x_{n+1}), x_n - x_{n+1} \rangle \\ &= \eta^2 \|y_{n+1}\|^2 - 2\eta^2 \langle \nabla f(x_{n+1}), x_{n+1} - z \rangle + 2\eta^3 \langle \nabla f(x_{n+1}), y_{n+1} \rangle \\ &= \eta^2 \|y_{n+1} + \eta \nabla f(x_{n+1})\|^2 - 2\eta^2 \langle \nabla f(x_{n+1}), x_{n+1} - z \rangle - \eta^4 \|\nabla f(x_{n+1})\|^2 \\ &= \eta^2 \|y_n\|^2 - 2\eta^2 \langle \nabla f(x_{n+1}), x_{n+1} - z \rangle - \eta^4 \|\nabla f(x_{n+1})\|^2 \\ &\leq 2\eta^2 (f(x_0) - f(x_n)) + 2\eta^2 (f(z) - f(x_{n+1})). \end{aligned} \quad (13)$$

The final step uses the convexity of f . Note that Eq. (13) takes the same form as Eq. (10) shown for the extragradient integrator. Since the implicit integrator also satisfies a bound on h_n from Lemma 18 that matches the bound for the h_n term for the extragradient integrator in Lemma 15, and the non increasing property of the Hamiltonian (Lemma 12), the rest of the proof follows according to the proof of Lemma 6. \blacksquare

F.2.2. PROOF OF LEMMA 17

Proof Since Eq. (13) for implicit integrator is identical with Equation (10) for extragradient integrator and Lemma 18 for implicit integrator is identical with Lemma 15 for extragradient integrator, the same derivation in the proof of Lemma 7 shows the following for iterations of implicit integrator, whenever $N \geq 4$:

$$\frac{1}{2} \|x^{\text{avg}}(x_0; N) - x^*\|^2 \leq \frac{1}{2} \|x_0 - x^*\|^2 + \frac{\eta^2 N(N+1)}{4} (f(x_0) - f(x^*)). \quad (14)$$

Therefore, like in the proof of Lemma 4 and Lemma 7, we consider a weighted sum of the following inequalities from the equations: Eqs. (12) and (14) and the non-increasing property of the implicit integrator (Lemma 12).

$$f(x^{\text{avg}}(x_0; N)) - f(z) + \frac{\|x_N - z\|^2}{3\eta^2 N(N+1)} \leq \frac{2}{3} (f(x_0) - f(z)) + \frac{\|x_0 - z\|^2}{3\eta^2 N(N+1)}$$

$$\begin{aligned} \frac{1}{3\eta^2 N(N+1)} \|x^{\text{avg}}(x_0; N) - x^*\|^2 &\leq \frac{1}{3\eta^2 N(N+1)} \|x_0 - x^*\|^2 + \frac{1}{6} (f(x_0) - f(x^*)) \\ f(x_N) - f(x^*) &\leq f(x_0) - f(x^*) \end{aligned}$$

with weights $\lambda, 1, \lambda^2$ respectively. Identical simplification as in the proof of [Lemma 7](#) then gives:

$$\begin{aligned} f(x^{\text{mix}}(x_0; N)) - f(x^*) + \frac{1}{3\lambda\eta^2 N(N+1)} \|x^{\text{mix}}(x_0; N) - x^*\|^2 \\ \leq c_\lambda (f(x_0) - f(x^*)) + \frac{1}{3\lambda\eta^2 N(N+1)} \|x_0 - x^*\|^2 \end{aligned}$$

which is precisely the statement of the lemma. ■

F.2.3. PROOF OF [LEMMA 18](#)

Proof Denote $h_n = \langle x_n - z, y_n \rangle$. We first compute the difference

$$\begin{aligned} h_{n+1} - h_n &= \langle x_{n+1} - z, y_{n+1} \rangle - \langle x_n - z, y_n \rangle \\ &= \langle x_{n+1} - z, y_{n+1} - y_n \rangle + \langle x_{n+1} - z, y_n \rangle - \langle x_n - z, y_n \rangle \\ &= -\eta \langle x_{n+1} - z, \nabla f(x_{n+1}) \rangle + \langle x_{n+1} - x_n, y_n \rangle. \end{aligned}$$

For the first term on the left hand side, we use convexity of f :

$$-\eta \langle x_{n+1} - z, \nabla f(x_{n+1}) \rangle \leq \eta (f(z) - f(x_{n+1})).$$

For the second term on the left hand side, we use the update rule and convexity of f :

$$\begin{aligned} \langle x_{n+1} - x_n, y_n \rangle &= \eta \langle y_{n+1}, y_n \rangle \\ &= \eta \langle y_n - \eta \nabla f(x_{n+1}), y_n \rangle \\ &= \eta \|y_n\|^2 - \eta \langle \nabla f(x_{n+1}), \eta y_n \rangle \\ &\stackrel{(a)}{=} \eta \|y_n\|^2 - \eta \langle \nabla f(x_{n+1}), x_{n+1} - x_n + \eta^2 \nabla f(x_{n+1}) \rangle \\ &= \eta \|y_n\|^2 - \eta^3 \|\nabla f(x_{n+1})\|^2 - \eta \langle x_{n+1} - x_n, \nabla f(x_{n+1}) \rangle \\ &\stackrel{(b)}{\leq} \eta \|y_n\|^2 + \eta (f(x_n) - f(x_{n+1})) \\ &\stackrel{(c)}{\leq} 2\eta f(x_0) - 2\eta f(x_n) + \eta (f(x_n) - f(x_{n+1})) \\ &= 2\eta f(x_0) - \eta f(x_n) - \eta f(x_{n+1}). \end{aligned}$$

In step (a), we use the property that $x_{n+1} = x_n + \eta y_{n+1} = x_n + \eta(y_n - \eta \nabla f(x_{n+1}))$. In step (b) we used the convexity of f . In step (c) we use the fact that along the iterates of implicit integrator, $f(x_n) + \frac{1}{2} \|y_n\|^2 \leq f(x_0)$.

Combining the bounds, we get

$$h_{n+1} - h_n \leq 2\eta (f(x_0) - f(z)) - \eta (f(x_n) - f(z)) - 2\eta (f(x_{n+1}) - f(z)).$$

Summing both sides from $n = 0$ to $n = N - 1$ and noting that $h_0 = 0$,

$$\begin{aligned} h_N &\leq 2\eta N(f(x_0) - f(z)) - 3\eta \sum_{n=1}^{N-1} (f(x_n) - f(z)) - 2\eta(f(x_N) - f(z)) - \eta(f(x_0) - f(z)) \\ &\leq 2\eta N(f(x_0) - f(z)) - 3\eta \sum_{n=1}^N (f(x_n) - f(z)) \end{aligned}$$

where in the final inequality we use the fact that $f(x_N) \leq f(x_0)$. ■

Appendix G. Numerical Experiments

In this section, we perform numerical experiments* to corroborate our theoretical findings in the preceding sections. We specifically assess the performance on two classical tasks: linear regression and logistic regression. We use (A, b) to denote the dataset where A is a $\mathbb{R}^{n \times d}$ matrix representing n covariates of dimension d each, and $b \in \mathcal{Y}^n$ is the vector of associated responses. For linear regression $\mathcal{Y} = \mathbb{R}$, and for logistic regression $\mathcal{Y} \in \{-1, +1\}$. The objectives for linear and logistic regression are

$$\begin{aligned} f_{\text{linear}}(x) &:= \frac{1}{2n} \|Ax - b\|^2 + \frac{\beta}{2} \|x\|^2, \\ f_{\text{logistic}}(x) &:= \frac{1}{n} \sum_{i=1}^n \log(1 + \exp(-b_i A_i^\top x)) + \frac{\beta}{2} \|x\|^2, \end{aligned}$$

respectively. In both settings, $\beta > 0$ is the regularization parameter.

Linear Regression We set $n = 100$ and $d = 200$ (an underdetermined system) where $\lambda_{\min}(A^\top A) = 0$. The matrix A is constructed by sampling each entry uniformly in $[-2/\sqrt{d}, 2/\sqrt{d}]$, and setting b as $Ax^* + \xi$ where $x^* = \mathbf{1}_d$ and ξ is a d -dimensional standard normal vector. The Hessian is computable in closed form as $\frac{1}{n} A^\top A + \beta \cdot I_d$, and hence the strong convexity and smoothness parameters are $\alpha = \beta + \frac{1}{n} \lambda_{\min}(A^\top A) = \beta$ and $L = \beta + \frac{1}{n} \lambda_{\max}(A^\top A)$ respectively and $\kappa = \frac{L}{\alpha}$.

Logistic Regression We set $n = 500$ and $d = 100$. The matrix A is constructed in the same way as done for linear regression above. The response is $b_i = \text{sign}(A_i^\top x^* + \xi)$ where $x^* = \mathbf{1}_d$ and ξ is a d -dimensional standard normal vector. The smoothness and strongly convexity parameters cannot be computed in closed form for this problem, however the strong convexity parameter is at least β owing to the convexity of the data fidelity and the regularization terms.

G.1. Comparison with gradient descent and accelerated gradient descent

For both linear regression and logistic regression, we consider two settings of β : $\beta \in \{0, 10^{-3}\}$. When $\beta = 0$, both problems are convex. For Nesterov's accelerated gradient method (AGD), the momentum parameter γ_k at iteration is set to $\frac{k-1}{k+2}$ in this setting. For **dHFA-eg**, $\lambda = \frac{1+\sqrt{3}}{2}$ and choose $q = \frac{1}{\sqrt{c_\lambda}}$ to determine the sequence $\{N_k\}_{k \geq 0}$ according to **Corollary 5**. On the other hand, when $\beta = 10^{-3}$, both problems are strongly convex. Here, for AGD, the momentum parameter

*. Code for this can be found at: <https://github.com/vishwakftw/dHFA>.

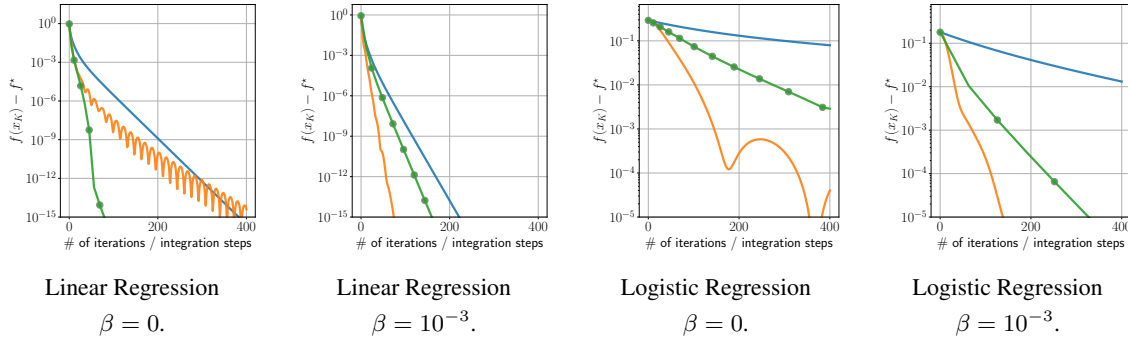


Figure 5: Comparison between GD (—), AGD (—), and dhFA-eg (—●—).

γ_k is set to $\frac{1-\sqrt{\eta\alpha}}{1+\sqrt{\eta\alpha}}$ and for dhFA-eg, $\lambda = 0$ and $N_k = \lceil \frac{2}{\eta\sqrt{\alpha}} \rceil$ for all iterations k according to Corollary 4. The step size η to set to $\frac{1}{L}$ for GD and AGD, and to $\frac{1}{\sqrt{L}}$ for dhFA-eg for the linear regression problem. For the logistic regression problem, we performed grid search on a collection of 25 uniformly log-spaced values from 10^{-5} to 1 to determine the optimal step size η , since we do not have exact access to the smoothness parameter L .

In Fig. 5, we plot these curves for each of the four settings. In linear regression with $\beta = 0$, we observe that dhFA-eg is clearly better than the rest, while in the other settings, AGD converges faster. This is likely due to a suboptimal setting of parameters, which we note can be further tuned. Regardless, we see clear acceleration by dhFA-eg relative to GD. Interestingly, in the logistic regression experiments, the best-performing step size on our search grid was $\eta = 1$ for all algorithms, which was the largest value considered.

G.2. Alternative averaging for iterates

The weighted average of the iterates x^{avg} used in dhFA-eg (Line 3) is motivated by the continuous-time average X^{avg} . Here, we investigate the effect of replacing this with the simple average of the iterates generated by extragradient integrator $x^{\text{s-avg}}(x_0; N) = \frac{1}{N} \sum_{n=1}^N x_n$. With the same setting of step size η for the empirical setups described previously, we run dhFA-eg with $x^{\text{s-avg}}$ in lieu of x^{avg} , and compare it with the original dhFA-eg; the results are plotted in Fig. 6. From these, interestingly we see an advantage of using the simple average in all settings except for linear regression with $\beta = 0$, which is a simple convex setting. We also observe that when $\beta = 10^{-3}$ in both the linear and logistic regression setups, the performance closely matches that of AGD.

G.3. Varying the integrator in dhFA-eg

dhFA-eg is a practically viable version of HFA which is made possible by using the extragradient integrator (HF^{extg}) for the Hamiltonian dynamics. Coincidentally, this integrator also satisfies desirable properties, which lead to the theoretical guarantees of dhFA-eg discussed in the previous section. In this subsection, we investigate how different integrators perform empirically in comparison to the extragradient integrator in dhFA-eg. We specifically focus on the explicit and leapfrog integrators (see Appendix B.2 for definitions) as these only require access to the gradients of f . The empirical results are plotted in Fig. 7.

We use the same empirical setups as described previously. The step size for these methods are set to the same value; for linear regression, this is set as $\eta = \frac{1}{\sqrt{L}}$, whereas for logistic regression,

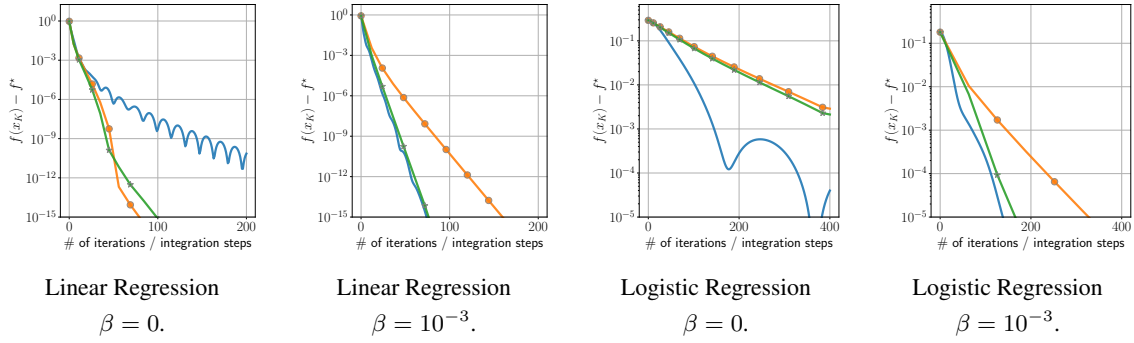


Figure 6: Comparison between averaging schemes. The plots depict three curves per setting: dhFA-eg with x^{avg} (—●—), dhFA-eg with $x^{\text{s-avg}}$ (—★—), and AGD (—).

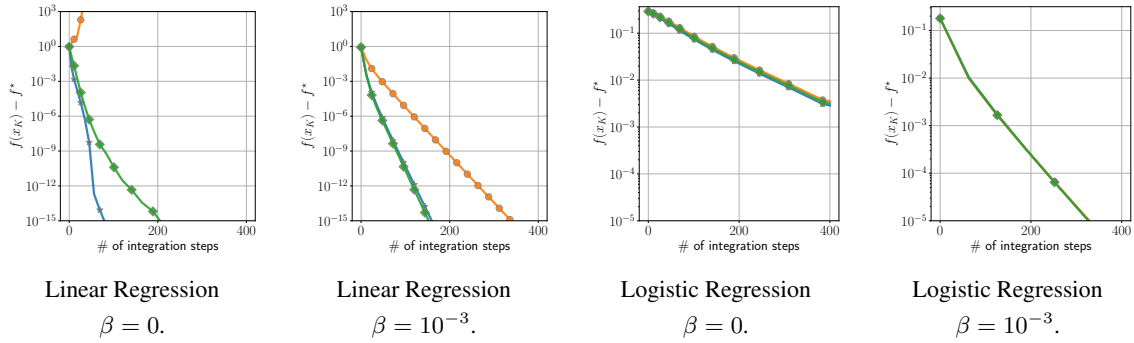


Figure 7: Comparison between integrators. The plots depict three curves per setting: dhFA-ex (—●—), dhFA-lf (—◆—), dhFA-eg (—★—). dhFA-ex and dhFA-lf are dhFA-eg but with the extragradient integrator replaced by the explicit and leapfrog integrators respectively.

this is set as $\eta = 1$. From the plots on the right, we see that the extragradient integrator is the best performing integrator in all settings. Interestingly, for the linear regression setup with $\beta = 0$, the explicit integrator diverges rapidly, while for the logistic regression problem, the performance matches closely with the others.