Optimization algorithms inspired by the geometry of dissipative systems

Alessandro Bravetti[1], Maria L. Daza-Torres[2], Hugo Flores-Arguedas[3], and Michael Betancourt[4]

[1]Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas (IIMAS), Universidad Nacional Autónoma de México, A.P. 70-543, 04510 Ciudad de México, México
alessandro.bravetti@iimas.unam.mx
[2]Universidad de Guadalajara, Guadalajara, México,
luisa.daza@academicos.udg.mx
[3]Centro de Investigación en Matemáticas (CIMAT), Guanajuato, México,
hugo.flores@cimat.mx
[4]Symplectomorphic LLC, New York, USA,
betan@symplectomorphic.com

Abstract

Accelerated gradient methods are a powerful optimization tool in machine learning and statistics but their development has traditionally been driven by heuristic motivations. Recent research, however, has demonstrated that these methods can be derived as discretizations of dynamical systems, which in turn has provided a basis for more systematic investigations, especially into the structure of those dynamical systems and their structure–preserving discretizations. In this work we introduce dynamical systems defined through a *contact geometry* which are not only naturally suited to the optimization goal but also subsume all previous methods based on geometric dynamical systems. These contact dynamical systems also admit a natural, robust discretization through geometric *contact integrators*. We demonstrate these features in paradigmatic examples which show that we can indeed obtain optimization algorithms that achieve oracle lower bounds on convergence rates while also improving on previous proposals in terms of stability.

Keywords: optimization, accelerated gradient, dynamical systems, geometric integrators, contact geometry

1 Introduction

Despite their practical utility and explicit convergence bounds, accelerated gradient methods have long been difficult to motivate from a fundamental theory. This

lack of understanding limits the theoretical foundations of the methods, which in turns hinders the development of new and more principled schemes. Recently a progression of work has studied the continuum limit of accelerated gradient methods, demonstrating that these methods can be derived as discretizations of continuous dynamical systems. Shifting focus to the structure and discretization of these latent dynamical systems provides a foundation for the systematic development and implementation of new accelerated gradient methods.

This recent direction of research began with [21] which found a continuum limit of Nesterov's accelerated gradient method (NAG)

$$X_k = P_{k-1} - \tau \nabla f(P_{k-1}) \tag{1}$$

$$P_k = X_k + \frac{k-1}{k+2}(X_k - X_{k-1})\,, \tag{2}$$

by discretizing the ordinary differential equation

$$\ddot{X} + \frac{3}{t}\dot{X} + \nabla f(X) = 0\,, \tag{3}$$

for $t > 0$ with the initial conditions $X(0) = X_0$ and $\dot{X}(0) = 0$. By generalizing the ordinary differential equation (3) they were then able to derive new accelerated gradient methods that achieved comparable convergence rates.

[23] continued in this direction by showing that NAG can also be derived as a discretization of a more structured *variational* dynamical system specified with a time–dependent Lagrangian, or equivalent Hamiltonian. Consider an objective function $f : \mathcal{X} \to \mathbb{R}$ which is continuously differentiable, convex, and has a unique minimizer $X^* \in \mathcal{X}$. Moreover assume that $\mathcal{X}$ is a convex set endowed with a distance–generating function $h : \mathcal{X} \to \mathbb{R}$ that is also convex and essentially smooth. From the *Bregman divergence* induced by h,

$$D_h(Y, X) = h(Y) - h(X) - \langle \nabla h(X), Y - X \rangle \tag{4}$$

they derived the *Bregman Lagrangian*

$$L(X, V, t) = \mathrm{e}^{\alpha+\gamma} \left(D_h(X + \mathrm{e}^{-\alpha} V, X) - \mathrm{e}^{\beta} f(X) \right)\,, \tag{5}$$

where α, β, and γ are continuously differentiable functions of time. They then proved that under the ideal scaling conditions

$$\dot{\beta} \leq \mathrm{e}^{\alpha} \tag{6}$$

$$\dot{\gamma} = \mathrm{e}^{\alpha}\,, \tag{7}$$

the solutions of the resulting Euler–Lagrange equations

$$\ddot{X} + (\mathrm{e}^{\alpha} - \dot{\alpha})\,\dot{X} + \mathrm{e}^{2\alpha+\beta} \left[\nabla^2 h(X + \mathrm{e}^{-\alpha}\dot{X}) \right]^{-1} \nabla f(X) = 0 \tag{8}$$

satisfy [23, Theorem 1.1]

$$f(X) - f(X^*) \leq \mathcal{O}(\mathrm{e}^{-\beta})\,. \tag{9}$$

From a physical perspective the two terms in equation (5) play the role of a kinetic and a potential energies, respectively. At the same time the explicit time–dependence of the Lagrangian (5) is a necessary ingredient in order for the dynamical

system to dissipate energy and relax to a minimum of the potential, and hence to a minimum of the objective function. Moreover, by (6), the optimal convergence rate is achieved by choosing $\dot{\beta} = \mathrm{e}^{\alpha}$, i.e. $\beta = \int_{t_0}^{t} \mathrm{e}^{\alpha(s)} ds$, and we observe that in the Euclidean case, $h(X) = \frac{1}{2} \|X\|^2$, the Hessian is the identity matrix and thus (8) simplifies to

$$\ddot{X} + (\mathrm{e}^{\alpha} - \dot{\alpha}) \dot{X} + \mathrm{e}^{2\alpha+\beta} \nabla f(X) = 0 \,. \tag{10}$$

Finally the authors developed a heuristic discretization of (8) that yielded optimization algorithms matching the continuous convergence rates.

[3] considered more systematic discretizations of these variational dynamical systems that exploited the fact that they are well suited for numerical discretizations that preserve their geometric structure [16]. In particular they Legendre transformed the Bregman Lagrangian (5) to derive the *Bregman Hamiltonian*,

$$H(X, P, t) = \mathrm{e}^{\alpha+\gamma} \left(D_{h^*}(\mathrm{e}^{-\gamma} P + \nabla h(X), \nabla h(X)) + \mathrm{e}^{\beta} f(X) \right) , \tag{11}$$

where $h^*(P) = \sup_V P \cdot V - h(X)$ is the Legendre transform of $h(X)$, and then argued that a principled way to obtain reliable and rate–matching discretizations of the resulting dynamical system

$$\dot{X} = \nabla_P H(X, P) \tag{12}$$

$$\dot{P} = -\nabla_X H(X, P) \,, \tag{13}$$

is with *symplectic integrators*. They numerically demonstrated in the Euclidean case that a standard leapfrog integrator yields an optimization algorithm that achieves polynomial convergence rates and showed how the introduction of a gradient flow could achieve late–time exponential convergence rates matching those seen empirically in other accelerated gradient methods.

Perhaps most importantly these variational methods allowed the focus to shift from existing accelerated gradient methods to the structure of the latent dynamical systems. Although the variational dynamical systems were initially derived to reproduce existing accelerated gradient methods, their existence suggests the introduction of new, principled dynamical systems that promise entirely new methods.

For example we can replace variational dynamical systems that exploit heuristic time–dependencies to achieve dissipation with explicitly *dissipative* dynamical systems. [19] and [12] considered a dynamical systems perspective on these systems, showing how relatively simple dissipations can achieve state-of-the-art convergence. [13] took a more geometric perspective, replacing the time–dependent Hamiltonian geometry of the variational systems with a *conformally symplectic* geometry that generates dynamical systems of the form

$$\dot{X} = \nabla_P H(X, P) \tag{14}$$

$$\dot{P} = -\nabla_X H(X, P) - c\, P \,, \tag{15}$$

with $c \in \mathbb{R}$ a constant. Being a geometric dynamical system this approach also admits effective geometric integrators similar to [3]. These conformally symplectic dynamical systems, however, are less general than the time–dependent variational dynamical systems; in particular NAG cannot be exactly recovered in this framework [13].

Another relevant aspect that has been uncovered by studying optimization algorithms from a variational or Hamiltonian analysis is the focus on a very important

degree of freedom, the choice of the kinetic energy, that plays a fundamental role in the construction of fast and stable algorithms that can possibly escape local minima, in direct analogy with what happens in Hamiltonian Monte Carlo methods [1, 2, 17]. In particular, first [18] and then [13] have motivated that a careful choice of the kinetic energy term can stabilize the dynamical systems when the objective function is rapidly changing, similar to the regularization induced by trust regions. Indeed, like the popular Adam, AdaGrad and RMSprop algorithms, the resulting *Relativistic Gradient Descent* (RGD) algorithm regularizes the dynamical velocities to achieve a faster convergence and improved stability.

Finally [18] introduced another way of incorporating dissipative terms into Hamilton's equations (12)–(13) (see also [20]). Their *Hamiltonian descent* algorithm is derived from the equations of motion

$$\dot{X} = \nabla_P H(X,P) + X^* - X \tag{16}$$

$$\dot{P} = -\nabla_X H(X,P) + P^* - P\,, \tag{17}$$

where $(X^*, P^*) = \operatorname{argmin} H(X,P)$. Because the dynamics are defined using terms only linear in X and P they converge to the optimal solution exponentially quickly under mild conditions on H [20]. That said, this exponential convergence requires already knowing the optimum (X^*, P^*) in order to generate the dynamics. Additionally this particular dynamical system lies outside of the variational and conformal symplectic families of dynamical systems and so can not take advantage of the geometric integrators.

In this work we show that all of the above–mentioned dynamical systems can be incorporated into a single family of *contact Hamiltonian systems* [4, 6] endowed with *a contact geometry*. The geometric foundation provides a powerful set of tools for both studying the convergence of the continuous dynamics as well as generating structure–preserving discretizations. We also produce a new version of RGD, *Contact Relativistic Gradient Descent* (CRGD), and provide numerical experiments that show how it improves over RGD in terms of stability and rates of convergence.

The structure of this work is as follows: in Section 2 we introduce contact Hamiltonian systems and show that all systems corresponding to Equations (10), (14)–(15), and (16)–(17) can be easily recovered as particular examples. In Section 3 we provide the basics of the geometric theory of time–discretization of contact systems by means of splitting, thus deriving optimization algorithms similar in spirit to, but more general than, those introduced in [3, 13]. Then in Section 4 and Section 5 we show numerically that our algorithms can improve both the speed of convergence and the stability with respect to the ones previously proposed. Finally, in Section 6 we summarize the results and discuss future directions.

2 Continuous–time contact optimization

Contact geometry is a rich subject, but to ease exposition we will consider here only how contact geometries manifest in Euclidean spaces. For a treatment of the more general theory see [5, 6, 9, 10, 11, 14].

In the Euclidean context a contact state space is odd–dimensional, $\mathbb{R}^{2n+1}$, and coordinated by the variables (X, P, S), where the $X \in \mathbb{R}^n$ play the role of *generalized coordinates*, the $P \in \mathbb{R}^n$ the corresponding *momenta* and $S \in \mathbb{R}$ the *action* of the system.

A contact geometry is defined by a *contact structure*. On a Euclidean state space there are two common ways to specify such a structure.

Example 1 (The standard contact structure in canonical coordinates)**.** The standard structure is defined as the kernel of the 1–form

$$\eta_{\text{std1}} := dS - PdX\,. \tag{18}$$

We use "standard" because one can show that a contact structure on any manifold looks like this one locally [15].

Example 2 (The standard contact structure in non–canonical coordinates)**.** This structure is defined as the kernel of the 1–form

$$\eta_{\text{std2}} := dS - \frac{1}{2}PdX + \frac{1}{2}XdP\,. \tag{19}$$

Although this appears different from the structure in Example 1 they define equivalent geometries.

Transformations that preserve the contact structure, and hence the contact geometry, play a special role on these spaces.

Definition 1. *A* contact transformation *or* contactomorphism $F : (M, \eta_1) \to (N, \eta_2)$ *is a map that preserves the contact structure*

$$F^*\eta_2 = \alpha_F \eta_1, \tag{20}$$

where F^* is the pullback induced by F, and $\alpha_F : M \to \mathbb{R}$ is a nowhere–vanishing function.

Remark 1. *From Definition 1 and Examples 1 and 2 we see that a contact map re–scales the contact* 1*–form by multiplying it by a nowhere–vanishing function. Indeed, such multiplication preserves the kernel of the* 1*–form, and hence the resulting geometry.*

Remark 2. *We can explicitly construct a contact transformation between η_{std1} and η_{std2} above. The map*

$$F : (X, P, S) \mapsto \left(X + P, \frac{P - X}{2}, S - \frac{XP}{2}\right) \tag{21}$$

satisfies $F^*\eta_{\text{std2}} = \eta_{\text{std1}}$. Consequently the two structures defined in Examples 1 and 2 are equivalent. The superficial difference arises only because they are written in different coordinates.

We can now define dynamical systems that generalize the Hamiltonian systems arising in symplectic geometries.

Definition 2 (Contact Hamiltonian systems)**.** *Given a possibly time–dependent differentiable function $\mathcal{H}(X, P, S, t)$ on the contact state space $(\mathbb{R}^{2n+1}, \eta)$, we define the* contact Hamiltonian vector field associated to $\mathcal{H}$ *as the vector field $X_{\mathcal{H}}$ satisfying*

$$\pounds_{X_{\mathcal{H}}}\eta = f_H \eta \qquad \eta(X_{\mathcal{H}}) = -\mathcal{H}\,, \tag{22}$$

where $\pounds_{X_{\mathcal{H}}}\eta$ denotes the Lie derivative of η with respect to $X_{\mathcal{H}}$ and η can be either η_{std1} or η_{std2} respectively. We denote the flow of $X_{\mathcal{H}}$ the contact Hamiltonian system associated to $\mathcal{H}$.

Remark 3. *The first condition in (22) simply ensures that the flow of $X_{\mathcal{H}}$ generates contact transformations, while the second condition requires the vector field to be generated by a Hamiltonian function.*

Lemma 1 (Contact Hamiltonian systems: std1)**.** *Given a (possibly time–dependent) differentiable function $\mathcal{H}(X, P, S, t)$ on the contact state space $(\mathbb{R}^{2n+1}, \eta_{\text{std1}})$, the associated contact Hamiltonian system is the following dynamical system*

$$\dot{X} = \nabla_P \mathcal{H} \tag{23}$$

$$\dot{P} = -\nabla_X \mathcal{H} - P\frac{\partial \mathcal{H}}{\partial S} \tag{24}$$

$$\dot{S} = \nabla_P \mathcal{H}\, P - \mathcal{H}\,. \tag{25}$$

Lemma 2 (Contact Hamiltonian system: std2)**.** *Given a (possibly time–dependent) differentiable function $\mathcal{H}(X, P, S, t)$ on the contact state space $(\mathbb{R}^{2n+1}, \eta_{\text{std2}})$, the associated contact Hamiltonian system is the following dynamical system*

$$\dot{X} = \nabla_P \mathcal{H} - \frac{1}{2} X\frac{\partial \mathcal{H}}{\partial S} \tag{26}$$

$$\dot{P} = -\nabla_X \mathcal{H} - \frac{1}{2} P\frac{\partial \mathcal{H}}{\partial S} \tag{27}$$

$$\dot{S} = \frac{1}{2}\left(X\nabla_X \mathcal{H} + P\nabla_P \mathcal{H}\right) - \mathcal{H}\,. \tag{28}$$

The proofs of the above lemmas follow from writing explicitly the conditions in (22) for η_{std1} and η_{std2} respectively, using Cartan's identity for the Lie derivative of a 1–form, and then contracting the first identity in (22) with the vector field $\partial/\partial S$ to show that the factor $f_{\mathcal{H}}$ has to be $\partial\mathcal{H}/\partial S$ in both cases.

Remark 4 (Lagrangian formulation)**.** *Contact systems can alternatively be introduced starting from the Lagrangian function $\mathcal{L}(X, V, S, t)$ and its corresponding generalized Euler–Lagrange equations*

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial V}\right) - \frac{\partial \mathcal{L}}{\partial X} - \frac{\partial \mathcal{L}}{\partial V}\frac{\partial \mathcal{L}}{\partial S} = 0\,, \tag{29}$$

together with the action equation

$$\dot{S} = \mathcal{L}(X, V, S, t)\,. \tag{30}$$

Indeed, for regular Lagrangians it can be shown that (23)–(25) are equivalent to the system (29)–(30).

We arrive at our main result with a direct calculation using equations (23)–(25) and (26)–(28),

Proposition 1 (Recovering previous frameworks)**.** *All the previously–mentioned frameworks for describing continuous–time optimization methods can be recovered as follows:*

i) If $\mathcal{H} = H(X, P, t)$, that is, if $\mathcal{H}$ does not depend explicitly on S, then from equations (23)–(24) we obtain the standard Hamiltonian equations (12)–(13), with (25) completely decoupled from the system. Consequently we recover all the results for the symplectic Hamiltonian analysis of the continuous–time Bregman dynamics considered in [3] as particular cases.

ii) If $\mathcal{H} = H(X,P,t) + c\,S$, then from equations (23)–(24) we obtain the standard equations for conformally symplectic systems (14)–(15), with (25) once again completely decoupled from the system. Consequently we recover all the results for the conformally symplectic analysis of continuous–time optimization dynamics considered in [13] as particular cases.

iii) If $\mathcal{H} = \frac{1}{2}\|P\|^2 + \mathrm{e}^{2\alpha+\beta} f(X) + (\mathrm{e}^{\alpha} - \dot{\alpha})\, S$, then from equations (23)–(24) we obtain the Euler–Lagrange equations (10) for the Euclidean Bregman Lagrangian. As in the first two cases (25) completely decouples from the system. Here we recover all the results obtained in [23] by exploiting this variational formulation in continuous time.

iv) If $\mathcal{H} = H(X,P,t) + X^*P - P^*X + 2S$, then from equations (26)–(27) we obtain the Hamiltonian descent equations (16)–(17), with (28) decoupled from the system. Consequently we recover all the results for the Hamiltonian descent analysis of continuous–time optimization dynamics considered in [18] and [20] as particular cases.

Remark 5 ($\mathcal{H}$ and Lyapunov functions)**.** *Although in Proposition 1.iii) we start with the Hamiltonian formulation and recover equation (10) directly, in principle, we can alternatively start by defining the Euclidean* contact Bregman Lagrangian*,*

$$\mathcal{L}(X,V,S,t) = \frac{\mathrm{e}^{-2\alpha}}{2}\|V\|^2 - \mathrm{e}^{\beta} f(X) - (\mathrm{e}^{\alpha} + \dot{\alpha})\, S\,, \tag{31}$$

and check that its associated generalized Euler–Lagrange equations (29) coincide with (10).

Moreover, if we compute the momenta using the standard Legendre transform,

$$P = \nabla_V L = \mathrm{e}^{-2\alpha} V\,, \tag{32}$$

then we obtain the corresponding contact Hamiltonian

$$\mathcal{H} = \frac{\mathrm{e}^{2\alpha}}{2}\|P\|^2 + \mathrm{e}^{\beta} f(X) + (\mathrm{e}^{\alpha} + \dot{\alpha})\, S\,. \tag{33}$$

The Hamiltonians in Proposition 1.iii) and in (33) are equivalent, in the sense that both lead to the same dynamical systems (10). (33), however, can be further manipulated into the equivalent form

$$\mathcal{H} = \mathcal{E}_t + (\mathrm{e}^{\alpha} + \dot{\alpha})\, S\,, \tag{34}$$

where

$$\mathcal{E}_t = \frac{\mathrm{e}^{2\alpha}}{2}\|P\|^2 + \mathrm{e}^{\beta}(f(X) - f(X^*))$$

is the Lyapunov function used in [23] to prove the rate of convergence of the dynamics (10).

This suggests constructing principled contact Hamiltonians for optimization purposes by taking

$$\mathcal{H} = \mathcal{E}_t + g(t)\, G(S) \tag{35}$$

and properly engineering the functions $g(t)$ and $G(S)$ to ensure that the dynamical system converges to the desired optimum.

Remark 6 (Continuous versus discrete equivalence)**.** *Although the different formulations of* (10) *lead to the same continuous dynamical systems they will not, in general, lead to the same discretized dynamical system. Having multiple formulations may make it easier to identify optimal discretizations and hence the most effective optimization algorithms.*

Remark 7 (Avoiding Optimum–Dependent Hamiltonians)**.** *In Proposition 1.iv) the contact Hamiltonian and the resulting equations of motion suffer from the same problem as the Hamiltonian descent case, namely that one needs to know the optimum in order to define the dynamical systems. Besides the possibility of using the techniques introduced in [20], we can also use the geometry of contact Hamiltonian systems to investigate equivalent contact Hamiltonians that do not require knowing the optimum a priori.*

The possibilities discussed in these remarks will be explored in future works. We conclude this section with an important corollary of Proposition 1.

Proposition 2 (Recovering NAG)**.** *Continuous–time NAG is contact.*

Proof. (3) is a particular case of (10) and therefore the continuous limit of NAG is a contact system. One possible contact Hamiltonian that generates these dynamics is

$$\mathcal{H} = \frac{1}{2}\|P\|^2 + f(X) + \frac{3}{t}S\,. \tag{36}$$

□

Remark 8. *Indeed, all the ODEs giving rise to the generalized Nesterov's schemes proposed in [21] are analogously seen to be contact systems.*

3 Discrete–time contact optimization

Before considering new optimization algorithms that stem from the discretization of contact Hamiltonian systems with geometric integrators, we will first prove the discrete–time analogue of Proposition 2 and show that discrete–time NAG is not given by a dynamical system alone but rather a composition of a contact map and a gradient descent. This result is inspired by the conjecture put forward in [3], who argued that symplectic maps followed by gradient descent steps can generate the exponential convergence near convex optima empirically observed in discrete–time NAG. Here we provide an actual proof that NAG is based on the composition of a contact map and a gradient step.

Proposition 3 (Recovering NAG)**.** *Discrete–time NAG,* (1)–(2)*, is given by the composition of a contact map and a gradient descent step.*

Proof. First we recall from Definition 1 that a contact transformation for the contact structure given by (19) is a map that satisfies

$$dS_{k+1} - \frac{1}{2}P_{k+1}dX_{k+1} + \frac{1}{2}X_{k+1}dP_{k+1} = f(X_k, P_k, S_k)\left(dS_k - \frac{1}{2}P_k dX_k + \frac{1}{2}X_k dP_k\right), \tag{37}$$

for some function $f(X_k, P_k, S_k)$ that is nowhere 0. Then we claim that NAG can be exactly decomposed in the contact state space as the composition of the map

$$X_{k+1} = P_k \tag{38}$$

$$P_{k+1} = X_{k+1} + \frac{k-1}{k+2}(X_{k+1} - X_k)\,, \tag{39}$$

$$S_{k+1} = \frac{k-1}{k+2} S_k\,, \tag{40}$$

which is readily seen to be a contact transformation satisfying

$$dS_{k+1} - \frac{1}{2}P_{k+1}dX_{k+1} + \frac{1}{2}X_{k+1}dP_{k+1} = \frac{k-1}{k+2}\left[dS_k - \frac{1}{2}P_k dX_k + \frac{1}{2}X_k dP_k\right]\,, \tag{41}$$

followed by a standard gradient descent map,

$$X_{k+1} = X_k - \tau\nabla f(X_k) \tag{42}$$

$$P_{k+1} = P_k \tag{43}$$

$$S_{k+1} = S_k\,. \tag{44}$$

□

Remark 9. *The fact that NAG has a latent geometric nature is already a step forward towards understanding its effectiveness. It is also of interest that we have been able to prove exactly the conjecture in [3] that discrete–time NAG can be obtained by composing structure–preserving maps, in this case a contact transformation, with gradient descent steps. In light of our result, it seems that this can indeed be the intrinsic mechanism responsible for the late–stage exponential convergence so often seen in NAG. We will not pursue this direction here, leaving it to future work.*

We now review a more systematic procedure to discretize contact Hamiltonian systems, which, when applied to equation (3) or to the more general (10), leads to new optimization algorithms. First we introduce the following lemmas from [8] which show how and when we can construct contact integrators of any even order.

Lemma 3 (Second–order contact integrator)**.** *Let the possibly time–dependent contact Hamiltonian be separable into the sum of functions*

$$\mathcal{H}(X, P, S, t) = \sum_{j=1}^{n} \phi_j(X, P, S, t), \tag{45}$$

such that each of the associated contact Hamiltonian vector fields X_{ϕ_j} are exactly integrable. Then the integrator

$$S_2(\tau) = \mathrm{e}^{\frac{\tau}{2}\partial_t}\mathrm{e}^{\frac{\tau}{2}X_{\phi_1}}\mathrm{e}^{\frac{\tau}{2}X_{\phi_2}}\cdots\mathrm{e}^{\tau X_{\phi_n}}\cdots\mathrm{e}^{\frac{\tau}{2}X_{\phi_2}}\mathrm{e}^{\frac{\tau}{2}X_{\phi_1}}\mathrm{e}^{\frac{\tau}{2}\partial_t} \tag{46}$$

is a second–order contact integrator.

Lemma 4 (Higher–order integrator with exact coefficients)**.** *If $S_{2n}(\tau)$ is an integrator of order $2n$ then the map*

$$S_{2n+2}(\tau) = S_{2n}(z_1\tau)S_{2n}(z_0\tau)S_{2n}(z_1\tau), \tag{47}$$

with z_0 *and* z_1 *given by*

$$z_0(n) = -\frac{2^{\frac{1}{2n+1}}}{2 - 2^{\frac{1}{2n+1}}}\,, \qquad z_1(n) = \frac{1}{2 - 2^{\frac{1}{2n+1}}}\,, \tag{48}$$

is an integrator of order $2n+2$.

Lemma 5 (Higher–order integrator with approximated coefficients)**.** *There exist* $m \in \mathbb{N}$ *and a set of real coefficients* $\{w_j\}_{j=0}^{m}$ *such that the map*

$$S^{(m)}(\tau) = S_2(w_m\tau)S_2(w_{m-1}\tau)\cdots S_2(w_0\tau)\cdots S_2(w_{m-1}\tau)S_2(w_m\tau)\,, \tag{49}$$

is an integrator of order $2n$. *The coefficients* $w_1, \ldots, w_m$ *are obtained as approximated solutions to an algebraic equation derived from the Baker–Campbell–Hausdorff formula.*

We refer to [8] for the proofs of these lemmas and to [8, 22] for the analysis of the corresponding geometric integrators.

Consequently all we need to find in order to obtain contact integrators for contact Hamiltonian systems such as (10) is a splitting of the corresponding contact Hamiltonian (33) into a sum of contact Hamiltonians whose vector fields are exactly integrable. As an example, we provide the next result.

Proposition 4 (Second–order contact optimization algorithm)**.** *Splitting the contact Hamiltonian* (33) *into the terms*

$$\mathcal{H}_{\phi_1} = \frac{\mathrm{e}^{2\alpha}}{2}\,\|P\|^2 \tag{50}$$

$$\mathcal{H}_{\phi_2} = \mathrm{e}^{\beta} f(X) \tag{51}$$

$$\mathcal{H}_{\phi_3} = (\mathrm{e}^{\alpha} + \dot{\alpha})\, S\,, \tag{52}$$

gives the following second–order contact integrator, which in turn derives an explicit optimization algorithm,

$$S_2(\tau) = \mathrm{e}^{\frac{\tau}{2}\partial_t}\mathrm{e}^{\frac{\tau}{2}X_{\phi_1}}\mathrm{e}^{\frac{\tau}{2}X_{\phi_2}}\mathrm{e}^{\tau X_{\phi_3}}\mathrm{e}^{\frac{\tau}{2}X_{\phi_2}}\mathrm{e}^{\frac{\tau}{2}X_{\phi_1}}\mathrm{e}^{\frac{\tau}{2}\partial_t}\,, \tag{53}$$

where each map is given by

$$\mathrm{e}^{\frac{\tau}{2}\partial_t}\begin{bmatrix} X \\ P \\ S \\ t \end{bmatrix} = \begin{bmatrix} X \\ P \\ S \\ t + \frac{\tau}{2} \end{bmatrix} \tag{54}$$

$$\mathrm{e}^{\frac{\tau}{2}X_{\phi_1}}\begin{bmatrix} X \\ P \\ S \\ t \end{bmatrix} = \begin{bmatrix} X + \frac{\tau}{2}\mathrm{e}^{2\alpha}P \\ P \\ S + \frac{\tau}{4}\mathrm{e}^{2\alpha}\,\|P\|^2 \\ t \end{bmatrix} \tag{55}$$

$$\mathrm{e}^{\frac{\tau}{2}X_{\phi_2}}\begin{bmatrix} X \\ P \\ S \\ t \end{bmatrix} = \begin{bmatrix} X \\ P - \frac{\tau}{2}\mathrm{e}^{\beta}\nabla f(X) \\ S - \frac{\tau}{2}\mathrm{e}^{\beta} f(X) \\ t \end{bmatrix} \tag{56}$$

$$\mathrm{e}^{\tau X_{\phi_3}}\begin{bmatrix} X \\ P \\ S \\ t \end{bmatrix} = \begin{bmatrix} X \\ P\,\mathrm{e}^{-\tau(\mathrm{e}^{\alpha}+\dot{\alpha})} \\ S\,\mathrm{e}^{-\tau(\mathrm{e}^{\alpha}+\dot{\alpha})} \\ t \end{bmatrix}. \tag{57}$$

Corollary 1 (Higher–order contact optimization algorithms)**.** *By combining the second–order contact optimization algorithm of Proposition 4 as in Lemmas 4 and 5, we can obtain contact optimization algorithms of any even order.*

Remark 10 (Matching rates)**.** *Given that the thus–obtained optimization algorithms are based on contact integrators of any even order, in principle one can use backward error analysis and show that the convergence rates of the corresponding discrete maps match those of the continuous differential equation up to the order of the integrator. See, for example, the discussion in [13]. This goes beyond the scope of the present work and will be presented elsewhere.*

Remark 11 (Increased stability)**.** *Since contact integrators are very stable under the increase of the step size τ (see [8]), we can use larger steps than standard optimization algorithms and achieve an effective increase in the rate of convergence. In particular, the higher the order of the integrator, the larger we can choose τ. See, for example, Example 3 below.*

3.1 Contact relativistic gradient descent

As an example of how the contact geometry framework can guide the generalization of optimization algorithms, let us consider generalizing the Relativistic Gradient Descent proposed in [13].

Proposition 5 (Contact Relativistic Gradient Descent)**.** *Consider the contact version of the Relativistic Gradient Descent (RGD) introduced in [13]. We start with the contact Hamiltonian*

$$\mathcal{H}(X,P,S,t) = c\sqrt{||P||^2 + (mc)^2} + f(X) + h(t)S\,, \tag{58}$$

with a time–dependent dissipative term $h(t) = \left(\gamma + \frac{\alpha}{t}\right)$ instead of the constant factor γ considered in [13]. Splitting the contact Hamiltonian into the sum of

$$\mathcal{H}_{\phi_1} = h(t)S \tag{59}$$
$$\mathcal{H}_{\phi_2} = f(X) \tag{60}$$
$$\mathcal{H}_{\phi_3} = c\sqrt{||P||^2 + (mc)^2}\,, \tag{61}$$

yields the following second–order contact integrator,

$$S_2(\tau) = \mathrm{e}^{\frac{\tau}{2}\partial_t}\mathrm{e}^{\frac{\tau}{2}X_{\phi_1}}\mathrm{e}^{\frac{\tau}{2}X_{\phi_2}}\mathrm{e}^{\tau X_{\phi_3}}\mathrm{e}^{\frac{\tau}{2}X_{\phi_2}}\mathrm{e}^{\frac{\tau}{2}X_{\phi_1}}\mathrm{e}^{\frac{\tau}{2}\partial_t}\,, \tag{62}$$

where each map is given explicitly by

$$\mathrm{e}^{\frac{\tau}{2}\partial_t}\begin{bmatrix}X\\P\\S\\t\end{bmatrix} = \begin{bmatrix}X\\P\\S\\t+\frac{\tau}{2}\end{bmatrix} \tag{63}$$

$$\mathrm{e}^{\frac{\tau}{2}X_{\phi_1}}\begin{bmatrix}X\\P\\S\\t\end{bmatrix} = \begin{bmatrix}X\\P\mathrm{e}^{-h(t)\frac{\tau}{2}}\\S\mathrm{e}^{-h(t)\frac{\tau}{2}}\\t\end{bmatrix} \tag{64}$$

$$\mathrm{e}^{\frac{\tau}{2}X_{\phi_2}}\begin{bmatrix}X\\P\\S\\t\end{bmatrix} = \begin{bmatrix}X\\P-\nabla f(X)\tau\\S-f(X)\tau\\t\end{bmatrix} \tag{65}$$

$$\mathrm{e}^{\tau X_{\phi_3}}\begin{bmatrix}X\\P\\S\\t\end{bmatrix} = \begin{bmatrix}X+\frac{cP}{\sqrt{\|P\|^2+(mc)^2}}\tau\\P\\P+\frac{-c^3m^2}{\sqrt{\|P\|^2+(mc)^2}}\tau\\t\end{bmatrix}. \tag{66}$$

These discretized dynamics define a new accelerated optimization algorithm that we call Contact Relativistic Gradient Descent (CRGD).

Remark 12. *If we take $h(t)=\gamma$ and using a first–order discretization*

$$S_1(\tau) = \mathrm{e}^{\tau X_{\phi_3}}\mathrm{e}^{\tau X_{\phi_2}}\mathrm{e}^{\tau X_{\phi_1}}, \tag{67}$$

applied to only the variables (X,P) then we re–obtain the RGD algorithm originally proposed in [13]. RGD, as well as any other algorithm based on conformally symplectic systems, can be considered as a particular contact optimization algorithm.

Remark 13. *If we fix $\gamma=\alpha$ and $\mu=\mathrm{e}^{-\gamma\tau}$ then can rewrite the map for X_{ϕ_1} as*

$$\mathrm{e}^{\frac{\tau}{2}X_{\phi_1}}\begin{bmatrix}X\\P\\S\\t\end{bmatrix} = \begin{bmatrix}X\\P\mu^{\frac{1}{2}\left(1+\frac{1}{t}\right)}\\S\mu^{\frac{1}{2}\left(1+\frac{1}{t}\right)}\\t\end{bmatrix}. \tag{68}$$

The dissipation parameter $\mu\in(0,1)$ in RGD is constant and therefore it has to be carefully tuned; in regions where $f(X)$ has a high curvature one would prefer such a parameter to be small while in flat regions one would like it to be as large as possible. A dynamically tuned μ, however, has the potential to work well in both regimes.

If we assume convex functions that are relatively flat around the minimum then the dynamics are likely to start in a region of high curvature before converging to the low curvature near the optimum. In other words we would want the dissipation parameter to transition from a small value to a larger value as the system evolves. Indeed the choice of $h(t)=\gamma+\gamma/t$ in CRGD is equivalent to a time–dependent dissipation parameter

$$\mu^{\frac{1}{2}\left(1+\frac{1}{t}\right)}, \tag{69}$$

which interpolates between 0 *and* $\mu^{\frac{1}{2}} < 1$ *with increasing time. Consequently this choice should improve convergence for objective functions that exhibit nearly flat regions. We explore this possibility in the numerical experiments below.*

This is just one example of the type of reasoning that can guide the generalization from standard symplectic and conformally symplectic to contact optimization algorithms.

4 Numerical experiments

In this section we compare the classical momentum algorithm (CM), NAG, RGD and CRGD on the benchmark examples considered in [13].

Example 3 (Quadratic function)**.** Let us start with a simple quadratic function

$$f(X) = \frac{1}{2} X^T A X, \quad X \in \mathbb{R}^{500}, \quad \lambda(A) \sim \mathcal{U}(10^{-3}, 1), \tag{70}$$

where $A \in \mathbb{R}^{500\times 500}$ is a positive–definite random matrix with eigenvalues uniformly distributed over the range $[10^{-3}, 1]$.

In Figure 1 we show the convergence rate of each algorithm when minimizing a $f(X)$ in each of 50 Monte Carlo simulations. We see that CRGD converges the fastest for most sampled objective functions.

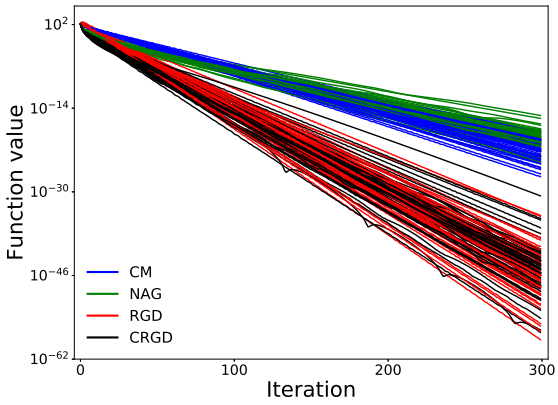

Figure 1: CRGD demonstrates the fastest convergence on random quadratic functions (70). The initial state for each run is always $X_0 = (1, \ldots, 1)^T$, $P_0 = 0$ (and $S_0 = 0$ for CRGD).

In Figure 2 we compare the performance of RGD and CRGD on this problem using dissipation parameter $\mu = 0.72$, step size $\tau = 0.43$, speed of light $v = 4403754.17$, and mass $m = 0.073$ all tuned to optimize the performance of RGD. For this particular tuning both RGD and CRGD exhibit similar rates of convergence, but if we increase the step size slightly to $\tau = 0.53$ then RGD becomes unstable and fails to converge entirely while the CRGD sustains the rapid convergence. This increased stability allows one to run CRGD with higher step sizes and faster convergence in practice.

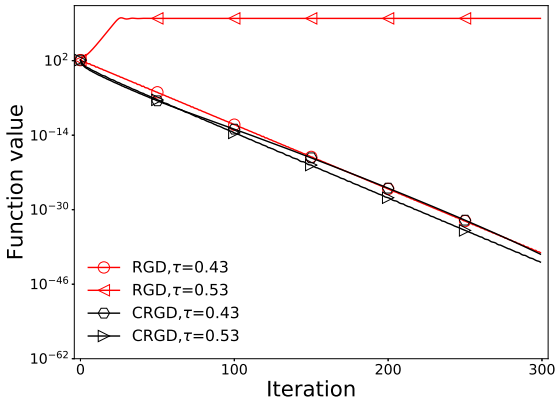

Figure 2: When well–tuned, RGD quickly converges to the optimum of a random quadratic objective function, but so too does CRGD with the same configuration. When we increase the step size, however, RGD becomes unstable and diverges while CRGD maintains its rapid convergence.

Example 4 (Correlated Quadratic function)**.** Next let us consider the correlated quadratic function

$$f(X) = \frac{1}{2} X^T A X, \quad A_{ij} = \frac{\sqrt{ij}}{2^{|i-j|}} \quad \text{for} \quad i,j = 1,\dots,50\,. \tag{71}$$

We perform 200 Monte Carlo simulations where the initial position in each is sampled uniformly at random in the range $-1 \leq X_{0,i} \leq 1$ to test the robustness of each method to the initialization. In this case we observe a similar behavior for the four methods (Fig. 3).

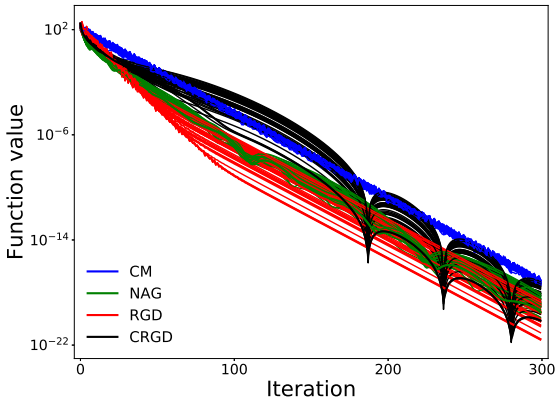

Figure 3: NAG, RGD and CRGD perform very similarly when targeting a strongly correlated quadratic objective function (71) with varying initial conditions.

Example 5 (Camelback Function)**.** To push the algorithms further we consider the nonconvex Camelback objective function,

$$f(X_1, X_2) = 2X_1^2 - 1.05X_1^4 + \frac{1}{6}X_1^6 + X_1X_2 + X_2^2. \tag{72}$$

The contour plot in Fig. 4(a) demonstrates the multimodality of this objective function, with three locally convex neighborhoods separated by nonconvex valleys. A unique global minimum can be found at $f(0) = 0$ while two local minima can be found at $(X_1, X_2) \simeq \pm(-1.75, 0.87)$ with $f \simeq 0.3$.

In Fig. 4(b) we set the initial state $X_0 = (5, 5)^T$, $P_0 = 0$, and $S_0 = 0$ for CRGD, and see that CRGD has the fastest convergence. For Fig. 4(c) we repeat the same experiment but initialize each algorithm close to one of the local minimizers. While CM and NAG are unable to escape the local minimum, both RGD and CRGD escape to the global minimum, with CRGD escaping earlier and converging to the global minimum faster after it has escaped.

For a quantitative comparison, we report the numerical estimation for the rate of convergence of RGD and CRGD in Table 1.

Initialization	$X_0 = (5, 5)^T$			$X_0 = (1.8, -0.9)^T$
Iterations	0-50	50-150	150-300	0-50
RGD	$t^{-3.17}$	$t^{-15.9}$	$t^{-27.52}$	$t^{-13.1}$
CRGD	$t^{-7.87}$	$t^{-23.13}$	$t^{-49.76}$	$t^{-17.7}$

Table 1: The superior convergence rate of CRGD seen in the examples of Fig. 4(b)–(c) can be quantified by numerically estimating the convergence rates of the competing algorithms.

In Fig. 4(d) we perform 500 experiments where the initial position is sampled uniformly in the range $-1 \leq X_{0,i} \leq 1$. Every algorithm was vulnerable to being trapped by the local minima, but CRGD found the global minimum more often than the other algorithms (Table 2).

Method	**Frequency of Finding Global Minimum**
CM	30.72 %
NAG	29.53 %
RGD	32.97 %
CRGD	33.10 %

Table 2: CRGD is able to find the global minimum of the Camelback objective function more frequently than CM, NAG, and RGD.

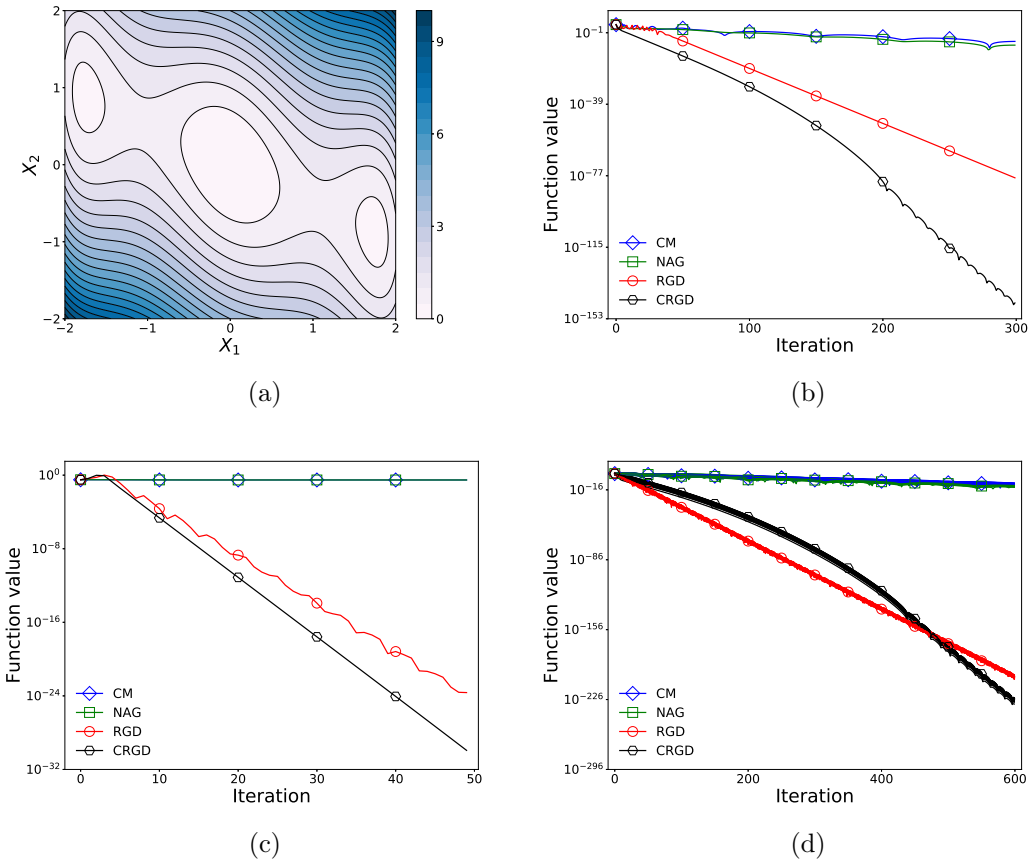

Figure 4: (a) The Camelback function (72) features three modes, one global minimum in the center surrounded by two local minima. (b) When initialized away from all of the minima, $X_0 = (5, 5)^T$, CRGD converges the fastest. (c) When initialized near one of the local minima, $X_0 = (1.8, -0.9)^T$, CRGD continues to dominate. (d) When the initialization is chosen at random, $-5 \leq X_{0,i} \leq 5$, CRGD displays the best asymptotic convergence.

Example 6 (Rosenbrock Function)**.** For a higher–dimensional challenge, let's consider the nonconvex Rosenbrock function,

$$f(X) = \sum_{i=1}^{n-1} \left(100(X_{i+1} - X_i^2)^2 + (1 - X_i)^2\right) , \tag{73}$$

with $n = 100$ dimensions. The Rosenbrock landscape is quite complex; for instance there are only two local minimizers, one global at $X^* = (1, 1, \ldots, 1)^T$ where $f(X^*) = 0$, and one local near $X \approx (-1, 1, \ldots, 1)^T$.

CRGD demonstrates the fastest convergence both when the algorithms are initialized close to the local minimum, $X_{0,2i} = 1$ and $X_{0,2i-1} = -1.2$, $i = 1, \ldots, 50$, (Fig. 5(b)) and when initialized in the tails of the Rosenbrock function, $X_{0,2i} = 5$

and $X_{0,2i-1} = -5$, $i = 1, \ldots, 50$ (Fig. 5(c)). Moreover if we sample the initialization uniformly at random in the range $-2.048 \leq X_{0,i} \leq 2.048$ then we see that the performance of CRGD is more robust to the specific initialization than CM, NAG, and RGD (Fig. 5(d)).

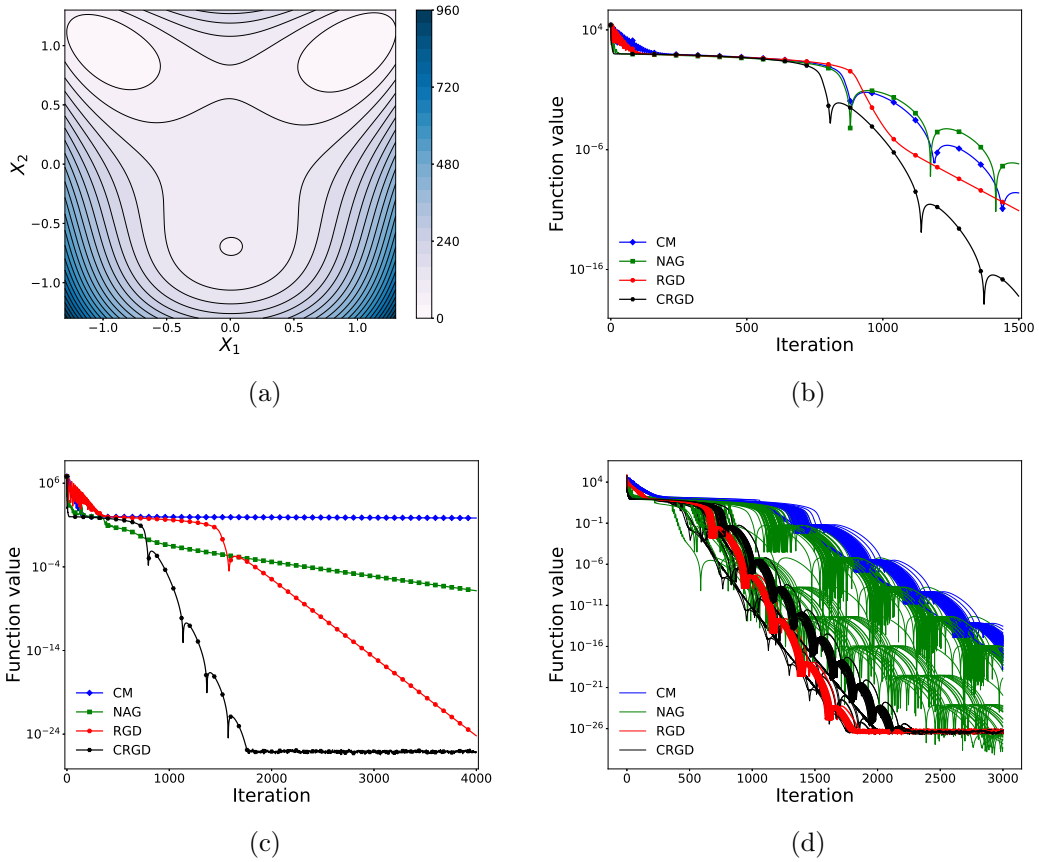

Figure 5: (a) A slice of the Rosenbrock function (73) where $X = (X_1, X_2, 1, \ldots, 1)$ demonstrates the complex landscape that can frustrate optimization algorithms. CRGD converges the fastest when the optimization algorithms are initialized (b) close to and (c) far from the local minimum. At the same time CRGD is less sensitive to the specific initialization, demonstrating much less variability as the initialization is randomly sampled.

5 Numerical Details

In this section we describe the tuning process for the parameters used in the examples of Section 4. In all the experiments we use an exhaustive random search on parameter space and the same number of Monte Carlo runs for each algorithm. Moreover we always set $P_0 = 0$ and $S_0 = 0$. All the numerical implementations have

been performed in Python, using the scipy, numpy and matplotlib packages. The codes are available in a Github repository (see [7]).

Quadratic function

In Fig. 1 we performed 50 samples of A and for each sample we ran each algorithm 150 times and for 200 iterations, and chose the parameters that give the lowest objective function value. Search ranges are shown in Table 3.

Algorithm	**Step Size**	**Momentum μ**	**Mass m**	**Speed of Light c**
CM	$[10^{-2}, 8 \cdot 10^{-1}]$	$[0.8, 0.999]$		
NAG	$[10^{-3}, 5 \cdot 10^{-1}]$	$[0.8, 0.999]$		
RGD	$[10^{-3}, 5 \cdot 10^{-1}]$	$[0.6, 0.95]$	$[10^{-4}, 10^{-2}]$	$[10^{3}, 10^{6}]$
CRGD	$[10^{-1}, 6 \cdot 10^{-1}]$	$[0.39, 0.9]$	$[10^{-3}, 10^{-1}]$	$[10^{4}, 10^{7}]$

Table 3: Ranges for parameters search in the experiment of Fig. 1.

Correlated quadratic function

For the experiment in Fig. 3, we performed 50 samples of the initial position, where each sample is chosen uniformly in the range $-1 \leq X_{0,i} \leq 1$. Then, for each initial position, we run each algorithm 100 times and for 100 iterations, and choose the parameters which give the lowest objective function value. Search ranges are shown in Table 4.

Algorithm	**Step Size**	**Momentum μ**	**Mass m**	**Speed of Light c**
CM	$[10^{-4}, 10^{-2}]$	$[0.6, 0.95]$		
NAG	$[10^{-4}, 10^{-2}]$	$[0.6, 0.95]$		
RGD	$[10^{-4}, 8 \cdot 10^{-3}]$	$[0.6, 0.8]$	$[10^{-6}, 10^{-4}]$	$[10^{3}, 10^{5}]$
CRGD	$[10^{-4}, 8 \cdot 10^{-3}]$	$[0.6, 0.95]$	$[10^{-4}, 10^{-2}]$	$[10^{3}, 10^{5}]$

Table 4: Ranges for parameters search in the experiment of Fig. 3 .

Camelback Function

For the experiment in Fig. 4(b)–(c), we run each algorithm 1500 times and for 300 iterations, and choose the parameters which give the lowest objective function value. For the experiment in Fig. 4(d) we performed 100 samples of the initial position, where each sample is chosen uniformly in the range $-5 \leq X_0 \leq 5$. Then, for each initial position, we run each algorithm 500 times and for 1200 iterations, and choose the parameters which give the lowest objective function value. Search ranges are show in Table 5.

Algorithm	Step Size	Momentum μ	Mass m	Speed of Light c
CM	$[10^{-5}, 10^{-3}]$	$[0.8, 0.999]$		
NAG	$[10^{-5}, 10^{-3}]$	$[0.8, 0.999]$		
RGD	$[10^{-5}, 8 \cdot 10^{-3}]$	$[0.3, 0.8]$	$[10^{-6}, 10^{-4}]$	$[10^{3}, 10^{5}]$
CRGD	$[10^{-5}, 8 \cdot 10^{-3}]$	$[0.15, 0.65]$	$[10^{-6}, 10^{-4}]$	$[10^{3}, 10^{5}]$

Table 5: Ranges for parameters search in the experiments of Fig. 4.

Rosenbrock Function

In the experiment in Fig. 5(b)–(c), we run each algorithm 500 times and for 1200 iterations, and choose the parameters which give the lowest objective function value. For CM, NAG and RGD, we use the same search range as in Table 5, except for the momentum factor, which is searched in the range $\mu \in [0.9, 0.98]$. For CRGD, we use the search range of Table 6. In the experiment in Fig. 5(d), we performed 20 samples of the initial position, where each sample is chosen uniformly in the range $-2.048 \leq X_0 \leq 2.048$. Then, for each initial position, we run each algorithm 500 times and for 1200 iterations, and choose the parameters which give the lowest objective function value. Search ranges are show in Table 6.

Algorithm	Step Size	Momentum μ	Mass m	Speed of Light c
CM	$[2 \cdot 10^{-4}, 4 \cdot 10^{-4}]$	$[0.94, 0.98]$		
NAG	$[2 \cdot 10^{-4}, 4 \cdot 10^{-4}]$	$[0.94, 0.98]$		
RGD	$[10^{-5}, 10^{-4}]$	$[0.93, 0.97]$	$[4 \cdot 10^{-7}, 10^{-6}]$	$[10^{4}, 9 \cdot 10^{4}]$
CRGD	$[10^{-5}, 8 \cdot 10^{-3}]$	$[0.9, 0.98]$	$[10^{-5}, 10^{-2}]$	$[10^{3}, 10^{5}]$

Table 6: Ranges for parameters search in the experiments of Fig. 5(d).

6 Conclusions

In this paper we have demonstrated that contact geometry, and contact Hamiltonian systems, naturally generate the dissipative dynamical systems whose discretizations give rise to accelerated gradients algorithms. Not only do these geometric systems subsume a wide range of dynamical systems previously considered in the literature, their geometric integration provides a principled means of constructing discretizations that preserve the important structure of the latent dynamics. Quite remarkably NAG itself can be decomposed as a contact transformation followed by a standard gradient descent step, demonstrating the fundamental nature of these systems.

We expect that unifying the development of optimization algorithms through this contact geometric lens will allow us to not only better understand these algorithms but also identify how their structure translates to ultimate performance and hence

derive improved algorithms more effectively. As a preliminary example we have shown that the RGD algorithm can be immediately generalized to the contact case, resulting in a more stable and generally faster algorithm.

The geometric foundation also brings with it a wealth of mathematical tools for the thorough analysis of the convergence of these algorithms, which we defer to subsequent work.

Acknowledgements

We would like to thank Diego Tapias, Mario Díaz, and Shin-itiro Goto for valuable comments, and Ana Pérez Arteaga and Ramiro Chávez Tovar for their help.

References

[1] Betancourt M. *A conceptual introduction to Hamiltonian Monte Carlo.* arXiv:1701.02434, 2017.

[2] Betancourt M., Byrne S., Livingstone S., Girolami M. et al. *The geometric foundations of Hamiltonian Monte Carlo.* Bernoulli, 23 : 2257–2298, 2017.

[3] Betancourt M., Jordan M. I. & Wilson A. C. *On symplectic optimization.* arXiv:1802.03653, 2018.

[4] Bravetti A. *Contact Hamiltonian dynamics: The concept and its use.* Entropy, 19, 2017.

[5] Bravetti A. *Contact geometry and thermodynamics.* International Journal of Geometric Methods in Modern Physics : 1940003, 2018.

[6] Bravetti A., Cruz H. & Tapias D. *Contact Hamiltonian mechanics.* Annals of Physics, 376 : 17–39, 2017.

[7] Bravetti A., Daza-Torres M. L., Flores-Arguedas H. & Betancourt M. *Contact optimization algorithms.* `https://github.com/mdazatorres/Contact-optimization-algorithms`, 2020.

[8] Bravetti A., Seri M., Vermeeren M. & Zadra F. *Numerical integration in celestial mechanics: a case for contact geometry.* Celestial Mechanics and Dynamical Astronomy, 132 : 1–29, 2020.

[9] Ciaglia F. M., Cruz H. & Marmo G. *Contact manifolds and dissipation, classical and quantum.* Annals of Physics, 398 : 159–179, 2018.

[10] de León M. & Lainz Valcázar M. *Contact Hamiltonian systems.* Journal of Mathematical Physics, 60 : 102902, 2019.

[11] de León M. & Sardón C. *Cosymplectic and contact structures for time-dependent and dissipative Hamiltonian systems.* Journal of Physics A: Mathematical and Theoretical, 50 : 255205, 2017.

[12] Diakonikolas J. & Jordan M. I. *Generalized momentum-based methods: A Hamiltonian perspective.* arXiv:1906.00436, 2019.

[13] França G., Sulam J., Robinson D. P. & Vidal R. *Conformal symplectic and relativistic optimization.* arXiv:1903.04100, 2019.

[14] Gaset J., Gràcia X., Muñoz-Lecanda M. C., Rivas X. & Román-Roy N. *New*

contributions to the Hamiltonian and Lagrangian contact formalisms for dissipative mechanical systems and their symmetries. arXiv:1907.02947, 2019.

[15] Geiges H. *An introduction to contact topology.* Volume 109. Cambridge University Press, 2008.

[16] Hairer E., Lubich C. & Wanner G. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations.* Volume 31 of *Springer Series in Computational Mathematics.* Springer-Verlag, Berlin Heidelberg, 2006.

[17] Livingstone S., Faulkner M. F. & Roberts G. O. *Kinetic energy choice in Hamiltonian/hybrid Monte Carlo.* Biometrika, 106 : 303–319, 2019.

[18] Maddison C. J., Paulin D., Teh Y. W., O'Donoghue B. & Doucet A. *Hamiltonian descent methods.* arXiv:1809.05042, 2018.

[19] Muehlebach M. & Jordan M. I. *A dynamical systems perspective on Nesterov acceleration.* arXiv:1905.07436, 2019.

[20] O'Donoghue B. & Maddison C. J. *Hamiltonian descent for composite objectives.* arXiv:1906.02608, 2019.

[21] Su W., Boyd S. & Candès E. J. *A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights.* Journal of Machine Learning Research, 17 : 1–43. `http://jmlr.org/papers/v17/15-084.html`, 2016.

[22] Vermeeren M., Bravetti A. & Seri M. *Contact variational integrators.* Journal of Physics A: Mathematical and Theoretical, 52 : 445206, 2019.

[23] Wibisono A., Wilson A. C. & Jordan M. I. *A variational perspective on accelerated methods in optimization.* Proceedings of the National Academy of Sciences, 113 : E7351–E7358, 2016.