A variational perspective on accelerated methods in optimization

Andre Wibisono[a,1], Ashia C. Wilson[b,1], and Michael I. Jordan[a,b,2]

[a]Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720; and [b]Department of Statistics, University of California, Berkeley, CA 94720

Contributed by Michael I. Jordan, September 15, 2016 (sent for review April 30, 2016; reviewed by Alexandre d'Aspremont and Stefano Soatto)

Accelerated gradient methods play a central role in optimization, achieving optimal rates in many settings. Although many generalizations and extensions of Nesterov's original acceleration method have been proposed, it is not yet clear what is the natural scope of the acceleration concept. In this paper, we study accelerated methods from a continuous-time perspective. We show that there is a Lagrangian functional that we call the Bregman Lagrangian, which generates a large class of accelerated methods in continuous time, including (but not limited to) accelerated gradient descent, its non-Euclidean extension, and accelerated higher-order gradient methods. We show that the continuous-time limit of all of these methods corresponds to traveling the same curve in spacetime at different speeds. From this perspective, Nesterov's technique and many of its generalizations can be viewed as a systematic way to go from the continuous-time curves generated by the Bregman Lagrangian to a family of discrete-time accelerated algorithms.

convex optimization | accelerated methods | Lagrangian framework | Bregman divergence | mirror descent

Optimization lies at the core of many fields concerned with data analysis. It provides a mathematical language in which both computational and statistical concepts can be expressed and it delivers practical data analysis algorithms that can scale to the enormous datasets that are increasingly the norm in science and technology. The recent literature on data analysis and optimization has focused on gradient-based optimization methods, given their low per-iteration cost and the relative ease with which they can be deployed on parallel and distributed processing architectures. Establishing that such methods do indeed address the scalability problems inherent in large-scale data analysis raises fundamental questions concerning the convergence rate of gradient-based methods, the extent to which those rates can be increased systematically, and whether there are upper bounds on achievable rates.

In the body of theory and practice built up to answer such questions, the phenomenon of acceleration plays a key role. In 1983, Nesterov introduced acceleration in the context of gradient descent for convex functions (1), showing that it achieves an improved convergence rate with respect to gradient descent and moreover that it achieves an optimal convergence rate under an oracle model of optimization complexity (2). The acceleration idea has since been extended to a wide range of other settings, including composite optimization (3–5), stochastic optimization (6, 7), nonconvex optimization (8, 9), and conic programming (10). There have been generalizations to non-Euclidean optimization (11, 12) and higher-order algorithms (13, 14), and there have been numerous applications that further extend the reach of the idea (15–18).

Despite this compelling evidence of the value of the idea of acceleration, it remains something of a conceptual mystery. Derivations of accelerated methods do not flow from a single underlying principle, but tend to rely on case-specific algebra (19). The basic Nesterov technique is often explained intuitively in terms of momentum, but this intuition does not easily carry over to non-Euclidean settings (20). In recent years, the number of explanations and interpretations of acceleration has increased (20–24), but these explanations have been focused on restrictive instances of acceleration, such as first-order algorithms, the Euclidean setting, or cases in which the objective function is strongly convex or quadratic. It is not yet clear what the natural scope of the acceleration concept is and indeed whether it is a single phenomenon.

In this paper we study acceleration from a continuous-time, variational point of view. We build on recent work by Su et al. (25), who show that the continuous-time limit of Nesterov's accelerated gradient descent is a second-order differential equation, and we take inspiration from the continuous-time analysis of mirror descent (2). In our approach, rather than starting from existing discrete-time accelerated gradient methods and deriving differential equations by taking limits, we take as our point of departure a variational formulation in which we define a functional on continuous-time curves that we refer to as a Bregman Lagrangian. Next, we calculate and discretize the Euler–Lagrange equation corresponding to the Bregman Lagrangian. It turns out that naive discretization (the Euler method) does not yield a stable discrete-time algorithm that retains the convergence rate of the underlying differential equation; rather, a more elaborate discretization involving an auxiliary sequence is necessary. This auxiliary sequence is essentially that used by Nesterov in his constructions of accelerated mirror descent (11) and accelerated cubic-regularized Newton's method (13) and later generalized by Baes (14). Thus, from our perspective, Nesterov's approach can be viewed as a methodology for the discretization of a certain class of differential equations. Given the complexities associated with the discretization of differential equations, it is perhaps not surprising

Significance

Optimization problems arise naturally in statistical machine learning and other fields concerned with data analysis. The rapid growth in the scale and complexity of modern datasets has led to a focus on gradient-based methods and also on the class of accelerated methods, first proposed by Nesterov in 1983. Accelerated methods achieve faster convergence rates than gradient methods and indeed, under certain conditions, they achieve optimal rates. However, accelerated methods are not descent methods and remain a conceptual mystery. We propose a variational, continuous-time framework for understanding accelerated methods. We provide a systematic methodology for converting accelerated higher-order methods from continuous time to discrete time. Our work illuminates a class of dynamics that may be useful for designing better algorithms for optimization.

Author contributions: A.W., A.C.W., and M.I.J. designed research, performed research, and wrote the paper.

Reviewers: A.dA., UMR CNRS 8548; and S.S., University of California, Los Angeles.

The authors declare no conflict of interest.

Freely available online through the PNAS open access option.

[1]A.W. and A.C.W. contributed equally to this work.

[2]To whom correspondence should be addressed. Email: jordan@cs.berkeley.edu.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1614734113/-/DCSupplemental.

that it has been difficult to perceive the generality and scope of the acceleration concept in a discrete-time framework.

Our Bregman Lagrangian framework permits a systematic understanding of the matching rates associated with higher-order gradient methods in discrete and continuous time. In the case of gradient descent, Su et al. (25) show that the discrete and continuous-time dynamics have convergence rates of $O(1/(\epsilon k))$ and $O(1/t)$, respectively, and that these match using the identification $t=\epsilon k$; for accelerated gradient descent, the convergence rates are $O(1/(\epsilon k^2))$ and $O(1/t^2)$, respectively, which match using the identification $t=\sqrt{\epsilon}k$. This result has been extended to the non-Euclidean case by Krichene et al. (12). Higher-order gradient descent is a descent method that minimizes a regularized $(p-1)$st-order Taylor approximation of the objective function f, generalizing gradient descent $(p=2)$ and Nesterov and Polyak's cubic-regularized Newton's method $(p=3)$ (26). For the pth-order gradient algorithm, we show that the discrete- and continuous-time dynamics have convergence rates of $O(1/(\epsilon k^{p-1}))$ and $O(1/t^{p-1})$, respectively, and that these match using the identification $t=\epsilon^{1/(p-1)}k$. The pth-order gradient algorithm with a constant step size ϵ has convergence rate $O(1/(\epsilon k^{p-1}))$ when $\nabla^{p-1}f$ is $(1/\epsilon)$ Lipschitz and, in continuous time, as $\epsilon\to 0$, this algorithm corresponds to the pth rescaled gradient flow, which is a first-order differential equation with a matching convergence rate $O(1/t^{p-1})$. Thus, the pth-order gradient algorithm can be seen as a discretization $t=\delta k$ of the rescaled gradient flow with time step $\delta=\epsilon^{1/(p-1)}$. Similarly, we show that the accelerated higher-order gradient algorithm achieves an improved convergence rate $O(1/(\epsilon k^p))$ under the same assumption [i.e., $\nabla^{p-1}f$ is $(1/\epsilon)$ Lipschitz]. In continuous time, as $\epsilon\to 0$, this corresponds to the second-order Euler–Lagrange curve of the Bregman Lagrangian with a matching convergence rate $O(1/t^p)$. Thus, the pth-order accelerated algorithm can be seen as a discretization $t=\delta k$ of the Euler–Lagrange equation of the Bregman Lagrangian with time step $\delta=\epsilon^{1/p}$.

In addition to its value in relating continuous-time and discrete-time acceleration, the study of the Bregman Lagrangian can provide further insights into the nature of acceleration. For instance, it is noteworthy that the Bregman Lagrangian is closed under time dilation. This means that if we take a Euler–Lagrange curve of a Bregman Lagrangian and reparameterize time so we travel the curve at a different speed, then the resulting curve is also the Euler–Lagrange curve of another Bregman Lagrangian, with appropriately modified parameters. Thus, the entire family of accelerated methods corresponds to a single curve in spacetime and can be obtained by speeding up (or slowing down) any single curve. Another insight is obtained by noting that from the discrete-time point of view, an interpretation of acceleration starts with a base algorithm, which we can accelerate by coupling with a suitably weighted mirror descent step. From the continuous-time point of view, however, it is the weighted mirror descent step that is important because the base gradient algorithm operates on a smaller time scale. Thus, Nesterov's accelerated gradient methods are but one possible implementation of second-order Bregman Lagrangian curves as a discrete-time algorithm.

The remainder of this paper is organized as follows. In *1. The Bregman Lagrangian*, we introduce the general family of Bregman Lagrangians and study its properties. In *2. Polynomial Convergence Rates and Accelerated Methods*, we demonstrate how to discretize the Euler–Lagrange equations corresponding to the polynomial subfamily of Bregman Lagrangians to obtain discrete-time accelerated algorithms. In particular, we introduce the family of higher-order gradient methods that can be used to complete the discretization. In *3. Further Explorations of the Bregman Lagrangian*, we discuss additional properties of the Bregman Lagrangian, including gauge-invariance properties, connection to classical gradient flows, and the correspondence with a functional that we refer to as a Bregman Hamiltonian. Finally, we end with a brief discussion in *4. Discussion*.

Problem Setting

We consider the optimization problem

$$\min_{x\in\mathcal{X}} f(x),$$

where $\mathcal{X}\subseteq\mathbb{R}^d$ is a convex set and $f:\mathcal{X}\to\mathbb{R}$ is a continuously differentiable convex function. To simplify the presentation in this paper we focus on the case $\mathcal{X}=\mathbb{R}^d$. We also assume f has a unique minimizer, $x^*\in\mathcal{X}$, satisfying the optimality condition $\nabla f(x^*)=0$. We use the inner product norm $\|x\|=\langle x,x\rangle^{1/2}$.

We consider the general non-Euclidean setting in which the space $\mathcal{X}$ is endowed with a distance-generating function $h:\mathcal{X}\to\mathbb{R}$ that is convex and essentially smooth (i.e., h is continuously differentiable in $\mathcal{X}$, and $\|\nabla h(x)\|_*\to\infty$ as $\|x\|\to\infty$). The function h can also be used to define an alternative measure of distance in $\mathcal{X}$ via its Bregman divergence,

$$D_h(y,x)=h(y)-h(x)-\langle\nabla h(x),y-x\rangle,$$

which is nonnegative because h is convex. When x is close to y, the Bregman divergence is an approximation to the Hessian metric,

$$D_h(y,x)\approx\frac{1}{2}\langle y-x,\nabla^2 h(x)(y-x)\rangle:=\frac{1}{2}\|y-x\|^2_{\nabla^2 h(x)}.$$

The Euclidean setting is obtained when $h(x)=\frac{1}{2}\|x\|^2$, in which case the Bregman divergence and Hessian metric coincide because $\nabla^2 h(x)$ is the identity matrix.

In continuous time, the Hessian metric is generally studied rather than the more general Bregman divergence; for instance, this is the case for natural gradient flow, which is the continuous-time limit of mirror descent (27, 28). By way of contrast, we shall see that our continuous-time, Lagrangian framework crucially employs the Bregman divergence.

In this paper we denote a discrete-time sequence in lowercase, e.g., x_k with $k\geq 0$ an integer. We denote a continuous-time curve in uppercase, e.g., X_t with $t\in\mathbb{R}$. An overdot means derivative with respect to time, i.e., $\dot{X}_t=\frac{d}{dt}X_t$.

1. The Bregman Lagrangian

We define the Bregman Lagrangian

$$\mathcal{L}(X,V,t)=e^{\alpha_t+\gamma_t}\left(D_h(X+e^{-\alpha_t}V,X)-e^{\beta_t}f(X)\right), \qquad [1]$$

which is a function of position $X\in\mathcal{X}$, velocity $V\in\mathbb{R}^d$, and time $t\in\mathbb{T}$, where $\mathbb{T}\subseteq\mathbb{R}$ is an interval of time. The functions $\alpha,\beta,\gamma:\mathbb{T}\to\mathbb{R}$ are arbitrary smooth (continuously differentiable) functions of time that determine the weighting of the velocity, the potential function, and the overall damping of the Lagrangian. We also define the ideal scaling conditions

$$\dot{\beta}_t\leq e^{\alpha_t} \qquad [2a]$$

$$\dot{\gamma}_t=e^{\alpha_t}; \qquad [2b]$$

these conditions are justified in the following section.

Convergence Rates of the Euler–Lagrange Equation. In this section we show that—under the ideal scaling assumption [2]—the Bregman Lagrangian [1] defines a variational problem, the solutions to which minimize the objective function f at an exponential rate.

Given a general Lagrangian $\mathcal{L}(X,V,t)$, we define a functional on curves $\{X_t:t\in\mathbb{T}\}$ via integration of the Lagrangian: $J(X)=\int_{\mathbb{T}}\mathcal{L}(X_t,\dot{X}_t,t)dt$. From the calculus of variations, a necessary

condition for a curve to minimize this functional is that it solve the Euler–Lagrange equation:

$$\frac{d}{dt}\left\{\frac{\partial \mathcal{L}}{\partial V}(X_t,\dot{X}_t,t)\right\}=\frac{\partial \mathcal{L}}{\partial X}(X_t,\dot{X}_t,t). \quad [3]$$

Specifically, for the Bregman Lagrangian [1], the partial derivatives are

$$\frac{\partial \mathcal{L}}{\partial X}(X,V,t)=e^{\gamma_t+\alpha_t}\left(\nabla h(X+e^{-\alpha_t}V)-\nabla h(X)-e^{-\alpha_t}\nabla^2 h(X)V-e^{\beta_t}\nabla f(X)\right) \quad [4a]$$

$$\frac{\partial \mathcal{L}}{\partial V}(X,V,t)=e^{\gamma_t}(\nabla h(X+e^{-\alpha_t}V)-\nabla h(X)). \quad [4b]$$

Thus, for general functions $\alpha_t,\beta_t,\gamma_t$, the Euler–Lagrange equation [3] for the Bregman Lagrangian [1] is a second-order differential equation given by

$$\ddot{X}_t+(e^{\alpha_t}-\dot{\alpha}_t)\dot{X}_t+e^{2\alpha_t+\beta_t}\left[\nabla^2 h\left(X_t+e^{-\alpha_t}\dot{X}_t\right)\right]^{-1}\nabla f(X_t)+e^{\alpha_t}(\dot{\gamma}_t-e^{\alpha_t})\left[\nabla^2 h\left(X_t+e^{-\alpha_t}\dot{X}_t\right)\right]^{-1}\left(\nabla h\left(X_t+e^{-\alpha_t}\dot{X}_t\right)-\nabla h(X_t)\right)=0. \quad [5]$$

We now impose the ideal scaling condition [2b]. In this case the last term in [5] vanishes, so the Euler–Lagrange equation simplifies to

$$\ddot{X}_t+(e^{\alpha_t}-\dot{\alpha}_t)\dot{X}_t+e^{2\alpha_t+\beta_t}\left[\nabla^2 h\left(X_t+e^{-\alpha_t}\dot{X}_t\right)\right]^{-1}\nabla f(X_t)=0. \quad [6]$$

In [6], we have assumed the Hessian matrix $\nabla^2 h(X_t+e^{-\alpha_t}\dot{X}_t)$ is invertible. But we can also write equation [6] in the following way, which requires only that ∇h be differentiable,

$$\frac{d}{dt}\nabla h\left(X_t+e^{-\alpha_t}\dot{X}_t\right)=-e^{\alpha_t+\beta_t}\nabla f(X_t). \quad [7]$$

To establish a convergence rate associated with solutions to the Euler–Lagrange equation—under the ideal scaling conditions—we take a Lyapunov function approach. Defining the energy functional

$$\mathcal{E}_t=D_h\left(x^*,X_t+e^{-\alpha_t}\dot{X}_t\right)+e^{\beta_t}(f(X_t)-f(x^*)), \quad [8]$$

we immediately obtain a convergence rate, as shown in *Theorem 1.1*. The derivation of the energy functional [8] is given in *SI Appendix*, *C. Deriving the Energy Functional*.

Theorem 1.1. *If the ideal scaling* [2] *holds, then solutions to the Euler–Lagrange equation* [7] *satisfy*

$$f(X_t)-f(x^*)\le O\left(e^{-\beta_t}\right).$$

Proof: The time derivative of the energy functional is

$$\dot{\mathcal{E}}_t=-\left\langle\frac{d}{dt}\nabla h\left(X_t+e^{-\alpha_t}\dot{X}_t\right),x^*-X_t-e^{-\alpha_t}\dot{X}_t\right\rangle+\dot{\beta}_t e^{\beta_t}(f(X_t)-f(x^*))+e^{\beta_t}\left\langle\nabla f(X_t),\dot{X}_t\right\rangle.$$

If X_t satisfies the Euler–Lagrange equation [7], then the time derivative simplifies to

$$\dot{\mathcal{E}}_t=-e^{\alpha_t+\beta_t}D_f\left(x^*,X_t\right)+\left(\dot{\beta}_t-e^{\alpha_t}\right)e^{\beta_t}(f(X_t)-f(x^*)),$$

where $D_f(x^*,X_t)=f(x^*)-f(X_t)-\langle\nabla f(X_t),x^*-X_t\rangle$ is the Bregman divergence of f. Note that $D_f(x^*,X_t)\ge 0$ because f is convex, so the first term in $\dot{\mathcal{E}}_t$ is nonpositive. Furthermore, if the ideal scaling condition [2a] holds, then the second term is also nonpositive, so $\dot{\mathcal{E}}_t\le 0$. Because $D_h(x^*,X_t+e^{-\alpha_t}\dot{X}_t)\ge 0$, this implies that for any $t\ge t_0\in\mathbb{T}$, $e^{\beta_t}(f(X_t)-f(x^*))\le\mathcal{E}_t\le\mathcal{E}_{t_0}$. Thus, $f(X_t)-f(x^*)\le\mathcal{E}_{t_0}e^{-\beta_t}=O(e^{-\beta_t})$, as desired. ■

For a given α_t, which determines γ_t by [2a], the optimal convergence rate is achieved by setting $\beta_t=e^{\alpha_t}$, resulting in convergence rate $O(e^{-\beta_t})=O(\exp(-\int_{t_0}^t e^{\alpha_s}\,ds))$. In *2. Polynomial Convergence Rates and Accelerated Methods* we study a subfamily of Bregman Lagrangians that have a polynomial convergence rate, and we show how we can discretize the resulting Euler–Lagrange equations to obtain discrete-time methods that have a matching, accelerated convergence rate. In *3. Further Explorations of the Bregman Lagrangian* we study another subfamily of Bregman Lagrangians that have an exponential convergence rate and discuss its connection to a generalization of Nesterov's restart scheme. In the Euclidean setting, our derivations simplify. We present these derivations in *SI Appendix*, *H. Further Properties* and comment on the insight that they provide into the question posed by Su et al. (25) on the significance of the value 3 in the damping coefficient for Nesterov's accelerated gradient descent.

Time Dilation. A notable property of the Bregman Lagrangian family is that it is closed under time dilation. This means if we take the Euler–Lagrange equation [5] of the Bregman Lagrangian [1] and reparameterize time to travel the curve at a different speed, the resulting curve is also the Euler–Lagrange equation of a Bregman Lagrangian with a suitably modified set of parameters.

Concretely, let $\tau:\mathbb{T}\to\mathbb{T}'$ be a smooth (twice-continuously differentiable) increasing function, where $\mathbb{T}'=\tau(\mathbb{T})\subseteq\mathbb{R}$ is the image of $\mathbb{T}$. Given a curve $X:\mathbb{T}'\to\mathcal{X}$, we consider the reparameterized curve $Y:\mathbb{T}\to\mathcal{X}$ defined by

$$Y_t=X_{\tau(t)}. \quad [9]$$

That is, the new curve Y is obtained by traversing the original curve X at a new speed of time determined by τ. If $\tau(t)>t$, then we say that Y is the sped-up version of X, because the curve Y at time t has the same value as the original curve X at the future time $\tau(t)$.

For clarity, we let $\mathcal{L}_{\alpha,\beta,\gamma}$ denote the Bregman Lagrangian [1] parameterized by α,β,γ. Then we have the following result whose proof is provided in *SI Appendix*, *A. Proof of Theorem 1.2*.

Theorem 1.2. *If X_t satisfies the Euler–Lagrange equation* [5] *for the Bregman Lagrangian $\mathcal{L}_{\alpha,\beta,\gamma}$, then the reparameterized curve $Y_t=X_{\tau(t)}$ satisfies the Euler–Lagrange equation for the Bregman Lagrangian $\mathcal{L}_{\tilde{\alpha},\tilde{\beta},\tilde{\gamma}}$, with modified parameters*

$$\tilde{\alpha}_t=\alpha_{\tau(t)}+\log\dot{\tau}(t) \quad [10a]$$

$$\tilde{\beta}_t=\beta_{\tau(t)} \quad [10b]$$

$$\tilde{\gamma}_t=\gamma_{\tau(t)}. \quad [10c]$$

Furthermore, α,β,γ satisfy the ideal scaling [2] *if and only if $\tilde{\alpha},\tilde{\beta},\tilde{\gamma}$ do.*

We note that in general, when we reparameterize time by a time-dilation function $\tau(t)$, the Lagrangian functional transforms to $\tilde{\mathcal{L}}(X,V,t)=\dot{\tau}(t)\,\mathcal{L}(X,\frac{1}{\dot{\tau}(t)}V,\tau(t))$. Thus, another way of stating the result in *Theorem 1.2* is to claim that

$$\mathcal{L}_{\tilde{\alpha},\tilde{\beta},\tilde{\gamma}}(X,V,t)=\dot{\tau}(t)\mathcal{L}_{\alpha,\beta,\gamma}\left(X,\frac{1}{\dot{\tau}(t)}V,\tau(t)\right), \quad [11]$$

which we can easily verify by directly substituting the definition of the Lagrangian [**1**] and the modified parameters $\tilde{\alpha},\tilde{\beta},\tilde{\gamma}$ [**10a–10c**].

In *2. Polynomial Convergence Rates and Accelerated Methods*, we show that the Bregman Lagrangian generates the family of higher-order accelerated methods in discrete time. Thus, the time-dilation property means that the entire family of curves for accelerated methods in continuous time corresponds to a single curve in spacetime, which is traveled at different speeds. This result suggests that the underlying solution curve has a more fundamental structure that is worth exploring further.

2. Polynomial Convergence Rates and Accelerated Methods

In this section, we study a subfamily of Bregman Lagrangians [**1**] with the following choice of parameters, indexed by a parameter $p>0$,

$$\alpha_t=\log p-\log t \quad [12a]$$

$$\beta_t=p\log t+\log C \quad [12b]$$

$$\gamma_t=p\log t, \quad [12c]$$

where $C>0$ is a constant. The parameters α,β,γ satisfy the ideal scaling condition [**2**] (with an equality on the first condition [**2a**]). The Euler–Lagrange equation [**6**] is given by

$$\ddot{X}_t+\frac{p+1}{t}\dot{X}_t+Cp^2t^{p-2}\left[\nabla^2 h\left(X_t+\frac{t}{p}\dot{X}_t\right)\right]^{-1}\nabla f(X_t)=0 \quad [13]$$

and, by *Theorem 1.1*, it has an $O(1/t^p)$ rate of convergence. As a direct result of the time-dilation property (*Theorem 1.2*), the entire family of curves [**13**] can be obtained by speeding up the curve in the case $p=2$ by the time-dilation function $\tau(t)=t^{p/2}$. In *SI Appendix*, *B. Existence and Uniqueness of Solution to the Polynomial Family* we discuss the issue of the existence and uniqueness of the solution to the differential equation [**13**].

The case $p=2$ of equation [**13**] is the continuous-time limit of Nesterov's accelerated mirror descent (11), and the case $p=3$ is the continuous-time limit of Nesterov's accelerated cubic-regularized Newton's method (13). The case $p=2$ has also been derived independently in the recent work of Krichene et al. (12); in the Euclidean case, when the Hessian $\nabla^2 h$ is the identity matrix, we recover the differential equation of Su et al. (25).

Naive Discretization. We now turn to the challenge of discretizing the differential equation in [**13**], with the goal of obtaining a discrete-time algorithm whose convergence rate matches that of the underlying differential equation. As we show in this section, a naive Euler method is not able to match the underlying rate. To match the rate a more sophisticated approach is needed, and it is at this juncture that Nesterov's three-sequence idea makes its appearance.

We first write the second-order equation [**13**] as the following system of first-order equations:

$$Z_t=X_t+\frac{t}{p}\dot{X}_t \quad [14a]$$

$$\frac{d}{dt}\nabla h(Z_t)=-Cpt^{p-1}\nabla f(X_t). \quad [14b]$$

Now we discretize X_t and Z_t into sequences x_k and z_k with time step $\delta>0$. That is, we make the identification $t=\delta k$ and set $x_k=X_t$, $x_{k+1}=X_{t+\delta}\approx X_t+\delta\dot{X}_t$ and $z_k=Z_t$, $z_{k+1}=Z_{t+\delta}\approx Z_t+\delta\dot{Z}_t$. Applying the forward-Euler method to [**14a**] gives the equation $z_k=x_k+\frac{\delta k}{p}\frac{1}{\delta}(x_{k+1}-x_k)$ or, equivalently,

$$x_{k+1}=\frac{p}{k}z_k+\frac{k-p}{k}x_k. \quad [15]$$

Similarly, applying the backward-Euler method to equation [**14b**] gives $\frac{1}{\delta}(\nabla h(z_k)-\nabla h(z_{k-1}))=-Cp(\delta k)^{p-1}\nabla f(x_k)$, which we can write as the optimality condition of the following weighted mirror descent step,

$$z_k=\arg\min_z\left\{Cpk^{p-1}\langle\nabla f(x_k),z\rangle+\frac{1}{\epsilon}D_h(z,z_{k-1})\right\}, \quad [16]$$

with step size $\epsilon=\delta^p$. In principle, the two updates [**15**] and [**16**] define an algorithm that implements the dynamics [**14a**] and [**14b**] in discrete time. However, we cannot establish a convergence rate for the algorithm in [**15**] and [**16**]; indeed, empirically, we find that the algorithm is unstable. Even for the simple case in which f is a quadratic function in two dimensions, the iterates of the algorithm initially approach and oscillate near the minimizer, but eventually the oscillation increases and the iterates shoot off to infinity.

A Rate-Matching Discretization. We now discuss how to modify the naive discretization scheme in [**15**] and [**16**] into an algorithm whose convergence rate matches that of the underlying differential equation. Our approach is inspired by Nesterov's constructions of accelerated mirror descent (11) and accelerated cubic-regularized Newton's method (13), which maintain three sequences in the algorithms and use the estimate sequence technique to prove convergence. Indeed, from our point of view, Nesterov's methodology can be viewed as a rate-matching discretization methodology.

Specifically, we consider the following scheme, in which we introduce a third sequence y_k to replace x_k in the updates,

$$x_{k+1}=\frac{p}{k+p}z_k+\frac{k}{k+p}y_k \quad [17a]$$

$$z_k=\arg\min_z\left\{Cpk^{(p-1)}\langle\nabla f(y_k),z\rangle+\frac{1}{\epsilon}D_h(z,z_{k-1})\right\}, \quad [17b]$$

where $k^{(p-1)}:=k(k+1)\cdots(k+p-2)$ is the rising factorial. A sufficient condition for the algorithm [**17**] to have an $O(1/(\epsilon k^p))$ convergence rate is that the new sequence y_k satisfies the inequality

$$\langle\nabla f(y_k),x_k-y_k\rangle\geq M\epsilon^{1/(p-1)}\|\nabla f(y_k)\|_*^{p/(p-1)}, \quad [18]$$

for some constant $M>0$. Note that in going from [**15**] to [**17a**] we have replaced the weight $\frac{p}{k}$ by $\frac{p}{k+p}$; this is only for convenience in the proof given below and does not change the asymptotics because $\frac{p}{k}=\Theta\left(\frac{p}{k+p}\right)$ as $k\to\infty$. Similarly, we replace k^{p-1} in [**16**] by the rising factorial $k^{(p-1)}$ in [**17b**] to make the algebra easier, but we still have $k^{(p-1)}=\Theta(k^{p-1})$.

The following result also requires a uniform convexity assumption on the distance-generating function h. Recall that h is σ-uniformly convex of order $p\geq 2$ if its Bregman divergence is lower bounded by the pth power of the norm,

$$D_h(y,x)\geq\frac{\sigma}{p}\|y-x\|^p. \quad [19]$$

The case $p=2$ is the usual definition of strong convexity. An example of a uniformly convex function is the pth power of the norm, $h(x)=\frac{1}{p}\|x-w\|^p$ for any $w\in\mathcal{X}$, which is σ-uniformly convex of order p with $\sigma=2^{-p+2}$ (ref. 13, lemma 4).

Theorem 2.1. *Assume h is 1-uniformly convex of order $p \geq 2$, and the sequence y_k satisfies the inequality* [18] *for all $k \geq 0$. Then the algorithm* [17] *with the constant $C \leq M^{p-1}/p^p$ and initial condition $z_0 = x_0 \in \mathcal{X}$ has the convergence rate*

$$f(y_k) - f(x^*) \leq \frac{D_h(x^*, x_0)}{C\epsilon k^{(p)}} = O\left(\frac{1}{\epsilon k^p}\right). \quad [20]$$

The proof of *Theorem 2.1* uses a generalization of Nesterov's estimate sequence technique and can be found in *SI Appendix*, *D. Proof of Theorem 2.1*. We note that with the scaling $\epsilon = \delta^p$ as in the previous section, the convergence rate $O(1/(\epsilon k^p))$ matches the $O(1/t^p)$ rate in continuous time for the differential equation [13]. We also note that the result in *Theorem 2.1* does not require any assumptions on f beyond the ability to construct a sequence y_k satisfying [18]. In the next section, we will see that we can satisfy [18] using the higher-order gradient method, which requires a higher-order smoothness assumption on f; the resulting algorithm is then the accelerated higher-order gradient method.

Higher-Order Gradient Method. We study the higher-order gradient update, which minimizes a regularized higher-order Taylor approximation of the objective function f.

Recall that for an integer $p \geq 2$, the $(p-1)$ st-order Taylor approximation of f centered at $x \in \mathcal{X}$ is the $(p-1)$ st degree polynomial

$$f_{p-1}(y;x) = \sum_{i=0}^{p-1} \frac{1}{i!} \nabla^i f(x)(y-x)^i = f(x) + \langle \nabla f(x), y - x \rangle + \cdots + \frac{1}{(p-1)!} \nabla^{p-1} f(x)(y-x)^{p-1}.$$

We say that f is L smooth of order $p-1$ if f is p-times continuously differentiable and $\nabla^{p-1} f$ is L Lipschitz, which means for all $x, y \in \mathcal{X}$,

$$\left\| \nabla^{p-1} f(y) - \nabla^{p-1} f(x) \right\|_* \leq L \|y - x\|. \quad [21]$$

For a constant $N > 0$ and step size $\epsilon > 0$, we define the update operator $G_{p,\epsilon,N} : \mathcal{X} \to \mathcal{X}$ by

$$G_{p,\epsilon,N}(x) = \arg\min_y \left\{ f_{p-1}(y;x) + \frac{N}{\epsilon p} \|y - x\|^p \right\}. \quad [22]$$

When f is smooth of order $p-1$, the operator $G_{p,\epsilon,N}$ has the following property, which generalizes (ref. 13, lemma 6). We provide the proof in *SI Appendix*, *E. Proof of Lemma 2.2*.

Lemma 2.2. *Let $x \in \mathcal{X}$, $y = G_{p,\epsilon,N}(x)$, and $N > 1$. If f is $L = \frac{(p-1)!}{\epsilon}$ smooth of order $p-1$, then*

$$\langle \nabla f(y), x - y \rangle \geq \frac{(N^2-1)^{(p-2)/(2p-2)}}{2N} \epsilon^{1/(p-1)} \|\nabla f(y)\|_*^{p/(p-1)}. \quad [23]$$

Furthermore,

$$\begin{aligned} &\frac{(N^2-1)^{(p-2)/(2p-2)}}{2N} \epsilon^{1/(p-1)} \|\nabla f(y)\|_*^{1/(p-1)} \\ &\leq \|x - y\| \\ &\leq \frac{1}{(N-1)^{1/(p-1)}} \epsilon^{1/(p-1)} \|\nabla f(y)\|_*^{1/(p-1)}. \end{aligned} \quad [24]$$

The inequality [23] means that we can use the update operator $G_{p,\epsilon,N}$ to produce a sequence y_k satisfying the requirement [18] under a higher-order smoothness condition on f. We state the resulting algorithm in the next section.

Higher-Order Gradient Method. In this section, we study the following higher-order gradient algorithm defined by the update operator $G_{p,\epsilon,N}$:

$$x_{k+1} = G_{p,\epsilon,N}(x_k). \quad [25]$$

The case $p = 2$ is the usual gradient descent algorithm, and the case $p = 3$ is Nesterov and Polyak's cubic-regularized Newton's method (26).

If f is smooth of order $p-1$, then the algorithm [25] is a descent method. Furthermore, we can prove the following rate of convergence, which generalizes the results for gradient descent and the cubic-regularized Newton's method. We provide the proof in *SI Appendix*, *F. Proof of Theorem 2.3*.

Theorem 2.3. *If f is $\frac{(p-1)!}{\epsilon}$ smooth of order $p-1$, then the algorithm* [25] *with constant $N > 0$ and initial condition $x_0 \in \mathcal{X}$ has the convergence rate*

$$f(x_k) - f(x^*) \leq \frac{p^{p-1}(N+1)R^p}{\epsilon k^{p-1}} = O\left(\frac{1}{\epsilon k^{p-1}}\right), \quad [26]$$

where $R = \sup_{x : f(x) \leq f(x_0)} \|x - x^*\|$ is the radius of the level set of f from the initial point x_0.

Rescaled Gradient Flow. We can take the continuous-time limit of the higher-order gradient algorithm as the step size $\epsilon \to 0$. The resulting curve is a first-order differential equation that is a rescaled version of gradient flow. We show that it minimizes f with a matching convergence rate. In the following, we take $N = 1$ in [25] for simplicity (the general N simply scales the vector field by a constant). We provide the proof of Theorem 2.4 in *SI Appendix*, *G. Proof of Theorem 2.4*.

Theorem 2.4. *The continuous-time limit of the algorithm* [25] *is the rescaled gradient flow*

$$\dot{X}_t = -\frac{\nabla f(X_t)}{\|\nabla f(X_t)\|_*^{(p-2)/(p-1)}}, \quad [27]$$

where we define the right-hand side to be the zero if $\nabla f(X_t) = 0$. Furthermore, the rescaled gradient flow has convergence rate

$$f(X_t) - f(x^*) \leq \frac{(p-1)^{p-1} R^p}{t^{p-1}} = O\left(\frac{1}{t^{p-1}}\right), \quad [28]$$

where $R = \sup_{x : f(x) \leq f(X_0)} \|x - x^*\|$ is the radius of the level set of f from the initial point X_0.

Equivalently, we can interpret the higher-order gradient algorithm [25] as a discretization of the rescaled gradient flow [27] with time step $\delta = \epsilon^{1/(p-1)}$, so $t = \delta k = \epsilon^{1/(p-1)} k$. With this identification, the convergence rates in discrete time, $O(1/(\epsilon k^{p-1}))$, and in continuous time, $O(1/t^{p-1})$, match. The convergence rate for the continuous-time dynamics does not require any assumption beyond the convexity and differentiability of f (as in the case of the Lagrangian flow [6]), whereas the convergence rate for the discrete-time algorithm requires the higher-order smoothness assumption on f. We note that the limiting case $p \to \infty$ of [27] is the normalized gradient flow, which has been shown to converge to the minimizer of f in finite time (29). We also note that unlike the Lagrangian flow, the family of rescaled gradient flows is not closed under time dilation.

Accelerated Higher-Order Gradient Method. By the result of *Lemma 2.2*, we can use the higher-order gradient update $G_{p,\epsilon,N}$ to produce a sequence y_k satisfying the inequality [18], to complete the algorithm [25] that implements the polynomial family of the

Bregman Lagrangian flow [**13**]. Explicitly, the resulting algorithm is as follows:

$$x_{k+1} = \frac{p}{k+p} z_k + \frac{k}{k+p} y_k \qquad \textbf{[29a]}$$

$$y_k = \arg\min_y \left\{ f_{p-1}(y; x_k) + \frac{N}{\epsilon p} \|y - x_k\|^p \right\} \qquad \textbf{[29b]}$$

$$z_k = \arg\min_z \left\{ Cpk^{(p-1)} \langle \nabla f(y_k), z \rangle + \frac{1}{\epsilon} D_h(z, z_{k-1}) \right\}. \qquad \textbf{[29c]}$$

By *Theorem 2.1* and *Lemma 2.2*, we have the following guarantee for this algorithm.

Corollary 2.5. *Assume f is $\frac{(p-1)!}{\epsilon}$ smooth of order $p-1$, and h is 1-uniformly convex of order p. Then the algorithm* [**29**] *with constants $N > 1$ and $C \le (N^2-1)^{(p-2)/2}/((2N)^{p-1}p^p)$ and initial conditions $z_0 = x_0 \in \mathcal{X}$ has an $O(1/(\epsilon k^p))$ convergence rate.*

The resulting algorithm [**29**] and its convergence rate recover the results of Baes (14), who studied a generalization of Nesterov's estimate sequence technique to higher-order algorithms. We note that the convergence rate $O(1/(\epsilon k^p))$ of algorithm [**29**] is better than the $O(1/(\epsilon k^{p-1}))$ rate of the higher-order gradient algorithm [**25**], under the same assumption of the $(p-1)$ st-order smoothness of f. This gives the interpretation of the algorithm [**29**] as "accelerating" the higher-order gradient method. Indeed, in this view the "base algorithm" that we start with is the higher-order gradient algorithm in the y-sequence [**29b**], and the acceleration is obtained by coupling it with a suitably weighted mirror descent step in [**29a**] and [**29c**].

However, from the continuous-time point of view, where our starting point is the polynomial Lagrangian flow [**13**], the algorithm [**29**] is only one possible implementation of the flow as a discrete-time algorithm. As pointed out in *2. Polynomial Convergence Rates and Accelerated Methods*, it is only the x- and z-sequences [**29a**] and [**29c**] that play a role in the correspondence between the continuous-time dynamics and their discrete-time implementation, and the requirement [**18**] in the y update is needed only to complete the convergence proof. Indeed, the higher-order gradient update [**29b**] does not change the continuous-time limit, because from [**24**] in *Lemma 2.2* we have that $\|x_k - y_k\| = \Theta(\epsilon^{1/(p-1)})$, which is smaller than the $\delta = \epsilon^{1/p}$ time step in the discretization of [**13**]. Therefore, the x and y sequences in [**29**] coincide in continuous time as $\epsilon \to 0$. Thus, from this point of view, Nesterov's accelerated methods (for the cases $p = 2$ and $p = 3$) are one of possibly many discretizations of the polynomial Lagrangian flow [**13**]. For instance, in the case $p = 2$, Krichene et al. (ref. 12, section 4.1) show that we can use a general regularizer in the gradient step [**29b**] under some additional smoothness assumptions. If there are other implementations, it would be interesting to see whether the higher-gradient methods have some distinguishing property, such as computational efficiency.

3. Further Explorations of the Bregman Lagrangian

In addition to providing a unifying framework for the generation of accelerated gradient-based algorithms, the Bregman Lagrangian has mathematical structure that can be investigated directly. In this section we briefly discuss some of the additional perspective that can be obtained from the Bregman Lagrangian. See *SI Appendix*, *H. Further Properties* for technical details of the results discussed here.

Hessian vs. Bregman Lagrangian. The presence of the Bregman divergence in the Bregman Lagrangian [**1**] is particularly striking. In the non-Euclidean setting, intuition might suggest using the Hessian metric $\nabla^2 h$ to measure a "kinetic energy" and thereby obtain a Hessian Lagrangian. This approach turns out to be unsatisfying, however, because the resulting differential equation does not yield a convergence rate and the Euler–Lagrange equation involves the third-order derivative $\nabla^3 h$, posing serious difficulties for discretization. As we have seen, the Bregman Lagrangian, on the other hand, readily provides a rate of convergence via a Lyapunov function; moreover, the resulting discrete-time algorithm in [**29**] involves only the gradient ∇h via the weighted mirror descent update.

Gradient vs. Lagrangian Flows. In the Euclidean case, it is known classically that we can view gradient flow as the strong-friction limit of a damped Lagrangian flow (ref. 30, p. 646). We show that the same interpretation holds for natural gradient flow and rescaled gradient flow. In particular, we show in *SI Appendix*, *H. Further Properties* that we can recover natural gradient flow as the strong-friction limit of a Bregman Lagrangian flow with an appropriate choice of parameters. Similarly, we can recover the rescaled gradient flow [**27**] as the strong-friction limit of a Lagrangian flow that uses the pth power of the norm as the kinetic energy. Therefore, the general family of second-order Lagrangian flows is more general and includes first-order gradient flows in its closure. From this point of view, a particle with gradient-flow dynamics is operating in the regime of high friction. The particle simply rolls downhill and stops at the equilibrium point as soon as the force $-\nabla f$ vanishes; there is no oscillation because it is damped by the infinitely strong friction. Thus, the effect of moving from a first-order gradient flow to a second-order Lagrangian flow is to reduce the friction from infinity to a finite amount; this permits oscillation (*cf.* refs. 12, 25, and 31), but also allows faster convergence.

Bregman Hamiltonian. One way to understand a Lagrangian is to study its Hamiltonian, which is the Legendre conjugate (dual function) of the Lagrangian. Typically, when the Lagrangian takes the form of the difference between kinetic and potential energy, the Hamiltonian is the sum of the kinetic and potential energy. The Hamiltonian is often easier to study than the Lagrangian, because its second-order Euler Lagrangian equation is transformed into a pair of first-order equations. In our case, the Hamiltonian corresponding to the Bregman Lagrangian [**1**] is the following Bregman Hamiltonian,

$$\mathcal{H}(X, P, t) = e^{\alpha_t + \gamma_t} \left(D_{h^*}(\nabla h(X) + e^{-\gamma_t} P, \nabla h(X)) + e^{\beta_t} f(X) \right),$$

which indeed has the form of the sum of the kinetic and potential energy. Here the kinetic energy is measured using the Bregman divergence of h^*, which is the convex dual function of h. See *SI Appendix*, *H. Further Properties* for further discussion.

Gauge Invariance. The Euler–Lagrange equation of a Lagrangian is gauge invariant, which means it does not change when we add a total time derivative to the Lagrangian. For the Bregman Lagrangian with the ideal scaling condition [**2b**], this property implies that we can replace the Bregman divergence $D_h(X + e^{-\alpha_t} V, X)$ in [**1**] by its first term $h(X + e^{-\alpha_t} V)$. This might suggest a different interpretation of the role of h in the Lagrangian.

Natural Motion. The natural motion of the Bregman Lagrangian (i.e., the motion when there is no force, $-\nabla f \equiv 0$) is given by $X_t = ae^{-\gamma_t} + b$, for some constants $a, b \in \mathcal{X}$. Note that even though the Bregman Lagrangian still involves the distance-generating function h, its natural motion is actually independent of h. Thus, the effect of h is felt only via its interaction with f—this can also be seen in [**6**] where h and f appear together only in the final term. Furthermore, assuming $e^{\gamma_t} \to \infty$, the natural motion always converges to a limit point, which a priori can be anything. However, as we see from *Theorem 1.1*, as soon

as we introduce a convex potential function f, all motions converge to the minimizer x^* of f.

Exponential Convergence Rate via Uniform Convexity. In addition to the polynomial family in *2. Polynomial Convergence Rates and Accelerated Methods*, we can also study the subfamily of Bregman Lagrangians that have exponential convergence rates $O(e^{-ct})$, $c>0$. As we discuss in *SI Appendix*, *H. Further Properties*, in this case the link to discrete-time algorithms is not as clear. Using the same discretization technique as in *2. Polynomial Convergence Rates and Accelerated Methods* suggests that to get a matching convergence rate, constant progress is needed at each iteration.

From the discrete-time perspective, we show that the higher-order gradient algorithm [**25**] achieves an exponential convergence rate when the objective function f is uniformly convex. Furthermore, we show that a restart scheme applied to the accelerated method [**29**] achieves a better dependence on the condition number; this generalizes Nesterov's restart scheme for the case $p=3$ (ref. 13, section 5).

It is an open question to understand whether there is a better connection between the discrete-time restart algorithms and the continuous-time exponential Lagrangian flows. In particular, it is of interest to consider whether a restart scheme is necessary to achieve exponential convergence in discrete time; we know it is not needed for the special case $p=2$, because a variant of Nesterov's accelerated gradient descent (32) that incorporates the condition number also achieves the optimal convergence rate.

4. Discussion

In this paper, we have presented a variational framework for understanding accelerated methods from a continuous-time perspective. We presented the general family of Bregman Lagrangians, which generates a family of second-order Lagrangian dynamics that minimize the objective function at an accelerated rate compared with gradient flows. These dynamics are related to each other by the operation of speeding up time, because the Bregman Lagrangian family is closed under time dilation. In the polynomial case, we showed how to discretize the second-order Lagrangian dynamics to obtain an accelerated algorithm with a matching convergence rate. The resulting algorithm accelerates a base algorithm by coupling it with a weighted mirror descent step. An example of a base algorithm is a higher-order gradient method, which in continuous time corresponds to a first-order rescaled gradient flow with a matching convergence rate. Our continuous-time perspective makes clear that it is the mirror descent coupling that is more important for the acceleration phenomenon rather than the base algorithm. Indeed, the higher-order gradient algorithm operates on a smaller time scale than the enveloping mirror descent coupling step, so it makes no contribution in the continuous-time limit, and in principle we can use other base algorithms.

Our work raises many questions for further research. First, the case $p=2$ is worthy of further investigation. In particular, the assumptions needed to show convergence of the discrete-time algorithm ($\nabla^{p-1}f$ is Lipschitz) are different from those required to show existence and uniqueness of solutions of the continuous-time dynamics (∇f is Lipschitz). In the case $p=2$, however, these assumptions match. This result suggests a strong link between the discrete- and continuous-time dynamics that might help us understand why several results seem to be unique to the special case $p=2$. Second, in discrete time, Nesterov's accelerated methods have been extended to various settings, for example to the stochastic setting. An immediate question is whether we can extend our Lagrangian framework to these settings. Third, we want to understand better the transition from continuous-time dynamics to discrete-time algorithms and whether we can establish general assumptions that preserve desirable properties (e.g., convergence rate). In *2. Polynomial Convergence Rates and Accelerated Methods* we saw that the polynomial convergence rate requires a higher-order smoothness assumption in discrete time, and in *3. Further Explorations of the Bregman Langrangian* we discussed whether the exponential case requires a uniform convexity assumption. Finally, our work to date focuses on the convergence rates of the function values rather than the iterates. Recently there has been some work extending ref. 25 to study the convergence of the iterates (33) and some perturbative aspects (34); it would be interesting to extend these results to the general Bregman Lagrangian.

At an abstract level, the general family of Bregman Lagrangians has a rich mathematical structure that deserves further study; we discussed some of these properties in *3. Further Explorations of the Bregman Langrangian*. We hope that doing so will give us new insights into the nature of the optimization problem in continuous time and help us design better dynamics with matching discrete-time algorithms. For example, we can study how to use some of the appealing properties of the Hamiltonian formalism (e.g., volume preservation in phase space) to help us discretize the dynamics. We also wish to understand where the Bregman Lagrangian itself comes from, why it works so well, and whether there are other Lagrangian families with similarly favorable properties.

ACKNOWLEDGMENTS. This work was supported in part by the Mathematical Data Science program of the Office of Naval Research.

1. Nesterov Y (1983) A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Mathematics Doklady* 27(2):372–376.
2. Nemirovskii A, Yudin D (1983) *Problem Complexity and Method Efficiency in Optimization* (Wiley, New York).
3. Nesterov Y (2007) *Gradient Methods for Minimizing Composite Objective Function*, CORE Discussion Papers 2007076 (Université Catholique de Louvain, Louvain-la-Neuve, Belgium).
4. Tseng P (2008) On accelerated proximal gradient methods for convex-concave optimization. Available at www.mit.edu/~dimitrib/PTseng/papers/apgm.pdf. Accessed October 27, 2016.
5. Beck A, Teboulle M (2009) A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J Imaging Sci* 2(1):183–202.
6. Hu C, Kwok JT, Pan W (2009) Accelerated gradient methods for stochastic optimization and online learning. *Advances in Neural Information Processing Systems (NIPS) 22*, eds Bengio Y, Schuurmans D, Lafferty JD, Williams CKI, Culotta A (Curran Associates, Inc., Red Hook, NY), pp 781–789.
7. Lan G (2012) An optimal method for stochastic composite optimization. *Math Program* 133(1–2):365–397.
8. Ghadimi S, Lan G (2015) Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Math Program* 156(1):59–99.
9. Li H, Lin Z (2015) Accelerated proximal gradient methods for nonconvex programming. *Advances in Neural Information Processing Systems (NIPS) 28*, eds Cortes C, Lawrence ND, Lee DD, Sugiyama M, Garnett R (Curran Associates, Inc., Red Hook, NY), pp 379–387.
10. Lan G, Lu Z, Monteiro R (2011) Primal-dual first-order methods with $O(1/\epsilon)$ iteration-complexity for cone programming. *Math Program* 126(1):1–29.
11. Nesterov Y (2005) Smooth minimization of non-smooth functions. *Math Program* 103(1):127–152.
12. Krichene W, Bayen A, Bartlett P (2015) Accelerated mirror descent in continuous and discrete time. *Advances in Neural Information Processing Systems (NIPS) 29*, eds Cortes C, Lawrence ND, Lee DD, Sugiyama M, Garnett R (Curran Associates, Inc., Red Hook, NY), pp 2845–2853.
13. Nesterov Y (2008) Accelerating the cubic regularization of Newton's method on convex problems. *Math Program* 112(1):159–181.
14. Baes M (2009) Estimate sequence methods: Extensions and approximations. Available at www.optimization-online.org/DB_FILE/2009/08/2372.pdf. Accessed June 30, 2015.
15. Ji S, Ye J (2009) An accelerated gradient method for trace norm minimization. *Proceedings of the 26th International Conference on Machine Learning (ICML)*, eds Bottou L, Littman M (ACM Press, New York), pp 457–464.
16. Ji S, Sun L, Jin R, Ye J (2009) Multi-label multiple kernel learning. *Advances in Neural Information Processing Systems (NIPS) 21*, eds Koller D, Schuurmans D, Bengio Y, Bottou L (Curran Associates, Inc., Red Hook, NY), pp 777–784.
17. Jojic V, Gould S, Koller D (2010) Accelerated dual decomposition for MAP inference. *Proceedings of the 27th International Conference on Machine Learning (ICML)*, eds Fuernkranz J, Joachims T (Omnipress, Madison, WI), pp 503–510.
18. Mukherjee I, Canini K, Frongillo R, Singer Y (2013) Parallel boosting with momentum. *Machine Learning and Knowledge Discovery in Databases*, eds Blockeel H, Kersting K, Nijssen S, Zelezny F (Springer, Berlin), pp 17–32.

19. Juditsky A, Nemirovsky A (2011) First-order methods for nonsmooth convex large-scale optimization, II: Utilizing problem's structure. *Optimization for Machine Learning*, eds Sra S, Nowozin S, Wright SJ (MIT Press, Cambridge, MA).
20. Allen-Zhu Z, Orecchia L (2014) Linear coupling: An ultimate unification of gradient and mirror descent. *arXiv*:1407.1537.
21. Arjevani Y, Shalev-Shwartz S, Shamir O (2015) On lower and upper bounds for smooth and strongly convex optimization problems. *arXiv*:1503.06833.
22. Flammarion N, Bach FR (2015) From averaging to acceleration, there is only a step-size. *Proceedings of the 28th Conference on Learning Theory (COLT)*, eds Grunwald P, Hazan E, Kale S (Association for Computational Learning, Barcelona), pp 658–695.
23. Lessard L, Recht B, Packard A (2016) Analysis and design of optimization algorithms via integral quadratic constraints. *SIAM J Optim* 26(1):57–95.
24. Bubeck S, Lee YT, Singh M (2015) A geometric alternative to Nesterov's accelerated gradient descent. *arXiv*:1506.08187.
25. Su W, Boyd S, Candès EJ (2014) A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights. *Advances in Neural Information Processing Systems (NIPS) 27*, eds Ghahramani Z, Welling M, Cortes C, Lawrence N, Weinberger KQ (Curran Associates, Inc., Red Hook, NY), pp 2510–2518.
26. Nesterov Y, Polyak BT (2006) Cubic regularization of Newton's method and its global performance. *Math Program* 108(1):177–205.
27. Alvarez F, Bolte J, Brahic O (2004) Hessian Riemannian gradient flows in convex programming. *SIAM J Contr Optim* 43(2):477–501.
28. Raskutti G, Mukherjee S (2015) The information geometry of mirror descent. *IEEE Trans Inf Theory* 61(3):1451–1457.
29. Cortés J (2006) Finite-time convergent gradient flows with applications to network consensus. *Automatica* 42(11):1993–2000.
30. Villani C (2008) *Optimal Transport, Old and New* (Springer, Berlin).
31. O'Donoghue B, Candès E (2015) Adaptive restart for accelerated gradient schemes. *Found Comput Math* 15(3):715–732.
32. Nesterov Y (2004) *Introductory Lectures on Convex Optimization: A Basic Course, Applied Optimization* (Kluwer, Boston).
33. Attouch H, Peypouquet J, Redont P (2015) On the fast convergence of an inertial gradient-like system with vanishing viscosity. *arXiv*:1507.04782.
34. Attouch H, Chbani Z (2015) Fast inertial dynamics and FISTA algorithms in convex optimization: Perturbation aspects. *arXiv*:1507.01367.