EURO Journal on Computational Optimization 9 (2021) 100015

Contents lists available at ScienceDirect

EURO Journal on Computational Optimization

journal homepage: www.elsevier.com/locate/ejco

First-Order Methods for Convex Optimization

Pavel Dvurechensky[a,b,c], Shimrit Shtern[d], Mathias Staudigl[e,*]

[a] Weierstrass Institute for Applied Analysis and Stochastics, Mohrenstr. 39, Berlin 10117, Germany
[b] Institute for Information Transmission Problems RAS, Bolshoy Karetny per. 19, build.1, Moscow, 127051, Russia
[c] Moscow Institute of Physics and Technology, 9 Institutskiy per., Dolgoprudny, Moscow Region, 141701, Russia
[d] Faculty of Industrial Engineering and Management, Technion - Israel Institute of Technology, Haifa, Israel
[e] Maastricht University, Department of Data Science and Knowledge Engineering (DKE) and Mathematics Centre Maastricht (MCM), Paul-Henri Spaaklaan 1, Maastricht 6229 EN, The Netherlands

ARTICLE INFO

2010 MSC:
90C25
90C30
90C06
68Q25
65Y20
68W40

Keywords:
Convex Optimization
Composite Optimization
First-Order Methods
Numerical Algorithms
Convergence Rate
Proximal Mapping
Proximity Operator
Bregman Divergence

ABSTRACT

First-order methods for solving convex optimization problems have been at the forefront of mathematical optimization in the last 20 years. The rapid development of this important class of algorithms is motivated by the success stories reported in various applications, including most importantly machine learning, signal processing, imaging and control theory. First-order methods have the potential to provide low accuracy solutions at low computational complexity which makes them an attractive set of tools in large-scale optimization problems. In this survey, we cover a number of key developments in gradient-based optimization methods. This includes non-Euclidean extensions of the classical proximal gradient method, and its accelerated versions. Additionally we survey recent developments within the class of projection-free methods, and proximal versions of primal-dual schemes. We give complete proofs for various key results, and highlight the unifying aspects of several optimization algorithms.

1. Introduction

The traditional standard in convex optimization was to translate a problem into a conic program and solve it using a primal-dual interior point method (IPM). The monograph Nesterov and Nemirovski (1994) was instrumental in setting this standard. The primal-dual formulation is a mathematically elegant and powerful approach as these conic problems can then be solved to high accuracy when the dimension of the problem is of moderate size. This philosophy culminated into the development of a robust technology for solving convex optimization problems which is nowadays the computational backbone of many specialized solution packages like MOSEK (Andersen and Andersen, 2000), or SeDuMi (Sturm, 1999). However, in general, the iteration costs of interior point methods grow non-linearly with the problem's dimension. As a result, as the dimension n of optimization problems grows, off-the shelve interior point methods eventually become impractical. As an illustration, the computational complexity of a single step of many standardized IPMs scales like n^3, corresponding roughly to the complexity of inverting an $n \times n$ matrix. This means that for already quite small problems of size like $n = 10^2$, we would need roughly 10^6 arithmetic operations just to compute a single iterate. From a practical viewpoint, such a scaling is not acceptable. An alternative solution approach, particularly attractive for such "large-scale" problems, are *first-order methods* (FOMs). These are iterative schemes with computationally cheap iterations usually known to yield low-precision solutions within reasonable computation time. The success-story of FOMs went hand-in-hand with the fast progresses made in data science, analytics and machine learning. In such data-driven optimization problems, the trade-off between fast iterations and low accuracy is particularly pronounced, as these problems usually feature high-dimensional decision variables. In these application domains precision is usually considered to be a subordinate goal because of the inherent randomness of the problem data, which makes it unreasonable to minimize with accuracy below the statistical error.

The development of first-order methods for convex optimization problems is still a very vibrant field, with a lot of stimulus from the

* Corresponding author.
E-mail addresses: pavel.dvurechensky@wias-berlin.de (P. Dvurechensky), shimrits@technion.ac.il (S. Shtern), m.staudigl@maastrichtuniversity.nl (M. Staudigl).

https://doi.org/10.1016/j.ejco.2021.100015
Received 30 December 2020; Received in revised form 19 August 2021; Accepted 11 October 2021

already mentioned applications in machine learning, statistics, optimal control, signal processing, imaging, and many more, see e.g. the recent review papers on optimization for machine learning (Bottou et al., 2018; Jain and Kar, 2017; Wright, 2018). Naturally, any attempt to try to survey this lively scientific field is already doomed from the beginning to be a failure, if one is not willing to make restrictions on the topics covered. Hence, in this survey we tried to give a largely self-contained and concise summary of some important families of FOMs, which we believe have had an ever-lasting impact on the modern perspective of continuous optimization. Before we give an outline of what is covered in this survey, it is therefore fair to mention explicitly, what is NOT covered in the pages to come. One major restriction we imposed on ourselves is the concentration on *deterministic* optimization algorithms. This is indeed a significant cut in terms of topics, since the field of stochastic optimization and randomized algorithms has particularly been at the forefront of recent progresses made. Nonetheless, we intentionally made this cut, since most of the developments within stochastic optimization algorithms are based on deterministic counterparts, and actually in many cases one can think of deterministic algorithms as the mean-field equivalent of a stochastic optimization technique. As a well-known example, we can mention the celebrated stochastic approximation theory initiated by Robbins and Monro (1951), with its close connection to deterministic gradient descent. See Benveniste et al. (1990); Kushner (1984); Ljung et al. (2012), for classical references from the point of view of systems theory and optimization, and Benaïm, 1998 for its deep connection with deterministic dynamical systems. This link has gained significant relevance in various stochastic optimization models recently (Davis et al., 2020; Duchi and Ruan, 2018; Mertikopoulos and Staudigl, 2018a; 2018b). Some excellent references on stochastic optimization are Shapiro et al., 2009 and Lan, 2020. Furthermore, we excluded some important classes of alternating minimization methods, such as block-coordinate descent, and variations thereof. Section 14 in the beautiful book (Beck, 2017) gives a thorough account of these methods, and we urge the interested reader to start reading there.

So, what is it that we actually do in this survey? Four seemingly different optimization algorithms are surveyed, all of which belong now to the standard toolkit of mathematical programmers. After introducing the (standard) notation that will be used in this survey, we give a precise formulation of the model problem for which modern convex optimization algorithms are developed. In particular, we focus on the general composite convex optimization model, including smooth and non-smooth terms. This model is rich enough to capture a significant class of convex optimization problems. Non-smoothness is an important feature of the model, as it allows us to incorporate constraints via penalty and barrier functions. Non-smooth optimization methods also gained a lot of attention in statistical and machine learning where regularization functions are usually included in the estimation part in order to promote sparsity or other a-priori relevant information about the estimator to be obtained. An efficient way to deal with non-smoothness is provided by the use of *proximal operators*, a key methodological contribution born within convex analysis (see Rockafellar and Wets (1998) for an historical overview). Section 3 introduces the general non-Euclidean proximal setup, which describes the mathematical framework within which the celebrated *Mirror Descent* and *Bregman proximal gradient methods* are analyzed nowadays. These tools achieved extreme popularity in online learning and convex optimization (Bubeck, 2015; Juditsky and Nemirovski, 2011a; 2011b). The main idea behind this technology is to exploit favorable structure in the problem's geometry to boost the practical performance of gradient-based methods. The proximal revolution has also influenced the further development of primal-dual optimization methods based on augmented Lagrangians. We review proximal variants of the celebrated Alternating Direction Method of Multipliers (ADMM) in Section 4. We then move on to give an in-depth presentation of projection-free optimization methods based on linear minimization oracles, the classical *Conditional Gradient* (CG) (a.k.a Frank-Wolfe) method and its recent variants. CG gained extreme popularity in large-scale optimization, mainly because of its good scalability properties and small iteration costs. Conceptually, it is an interesting optimization method, as it allows us to solve convex programming problems with complicated geometry on which proximal operators are not easy to evaluate. This, in fact, applies to many important domains, like the Spectrahedron, or domains defined via intersections of several half spaces. CG is also relevant when the iterates should preserve structural features of the desired solution, like sparsity. Section 5 gives a comprehensive account of this versatile method.

All the methods we discussed so far generally provide sublinear convergence guarantees in terms of function values with iteration complexity of $O(1/\varepsilon)$. In his influential paper (Nesterov, 1983), Nesterov published an optimal method with iteration complexity of $O(1/\sqrt{\varepsilon})$ to reach an ε-optimal solution. This was the starting point for the development of acceleration techniques for given FOMs. Section 6 summarizes the recent developments in this field.

With writing this survey, we tried to give a holistic presentation of the main methods in use. At various stages in the survey, we establish connections, if not equivalences, between various methods. For many of the key results we provide self-contained proofs.

Notation We use standard notation and concepts from convex and variational analysis, which, unless otherwise specified, can all be found in the monographs (Bauschke and Combettes, 2016; Hiriart-Urrut and Lemaréchal, 2001; Rockafellar and Wets, 1998). Throughout this article, we let V represent a finite-dimensional vector space of dimension n with norm $\|\cdot\|$. We will write V^* for the (algebraic) dual space of V with duality pairing $\langle y, x\rangle$ between $y \in \mathsf{V}^*$ and $x \in \mathsf{V}$. The dual norm of $y \in \mathsf{V}^*$ is $\|y\|_* = \sup\{\langle y, x\rangle | \quad \|x\| \leq 1\}$. The set of proper lower semi-continuous functions $f : \mathsf{V} \to (-\infty, \infty]$ is denoted as $\Gamma_0(\mathsf{V})$. The (effective) domain of a function $f \in \Gamma_0(\mathsf{V})$ is defined as $\operatorname{dom} f = \{x \in \mathsf{V} | f(x) < \infty\}$. For a given continuously differentiable function $f : \mathsf{X} \subseteq \mathsf{V} \to \mathbb{R}$ we denote its gradient vector

$$\nabla f(x_1, \dots, x_n) = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n}\right)^\top.$$

The subdifferential at a point $x \in \mathsf{X} \subseteq \mathsf{V}$ of a convex function $f : \mathsf{V} \to \mathbb{R} \cup \{+\infty\}$ is denoted as

$$\partial f(x) = \{p \in \mathsf{V}^* | f(y) \geq f(x) + \langle p, y - x\rangle \quad \forall y \in \mathsf{V}\}. \tag{1.1}$$

The elements of $\partial f(x)$ are called subgradients.

As a notational convention, we write matrices in bold capital fonts. Given some set $\mathsf{X} \subseteq \mathsf{V}$, denote its relative interior as $\operatorname{relint}(\mathsf{X})$. Recall that, if the dimension of the set X agrees with the dimension of the ground space V, then the relative interior coincides with the topological interior, which we denote as $\operatorname{int}(\mathsf{X})$. Hence, the two notions differ only in situations where X is contained in a lower-dimensional submanifold. We denote the closure as $\operatorname{cl}(\mathsf{X})$. The boundary of X is defined in the usual way $\operatorname{bd}(\mathsf{X}) = \operatorname{cl}(\mathsf{X}) \setminus \operatorname{int}(\mathsf{X})$.

2. Composite convex optimization

In this survey we focus on the generic optimization problem

$$\min_{x \in \mathsf{X}} \{\Psi(x) := f(x) + r(x)\}. \tag{P}$$

At many stages of this survey, the following properties are imposed on the data of the minimization problem (P):

Assumption 1.

(a) $\mathsf{X} \subseteq \mathsf{V}$ is a nonempty closed convex set embedded in a finite-dimensional real vector space V;

(b) $f : \mathsf{V} \to \mathbb{R}$ is convex and continuously differentiable on a neighborhood of X. Furthermore, it possesses a L_f-Lipschitz continuous gradient on X:

$$(\forall x, x' \in \mathsf{X}) : \quad \|\nabla f(x) - \nabla f(x')\|_* \leq L_f \|x - x'\| \tag{2.1}$$

(c) $r \in \Gamma_0(\mathsf{V})$ and μ-strongly convex on V for some $\mu \geq 0$ with respect to a norm $\|\cdot\|$ on V. This means that for all $x, y \in \operatorname{dom} r$, and any selection $r'(x) \in \partial r(x)$, we have

$$r(y) \geq r(x) + \langle r'(x), y - x\rangle + \frac{\mu}{2}\|x - y\|^2.$$

If $\mu = 0$ then the function r is called convex.

We are interested in problems which are feasible.

Assumption 2. $\operatorname{dom} r \cap \mathsf{X} \neq \emptyset$.

In many recent applications, the smooth function f represents a data fidelity term and the non-smooth part r takes the role of a penalty function or regularizer. The most important examples of function r are as follows:

- r is an indicator function of a closed convex set $\mathsf{C} \subset \mathsf{V}$ with $\mathsf{C} \cap \mathsf{X} \neq \emptyset$:

$$r(x) = \delta_{\mathsf{C}}(x) := \begin{cases} 0 & \text{if } x \in \mathsf{C}, \\ +\infty & \text{if } x \notin \mathsf{C}. \end{cases} \tag{2.2}$$

- r is a self-concordant barrier (Nesterov, 2018b; Nesterov and Nemirovski, 1994) for a closed convex set $\mathsf{C} \subset \mathsf{V}$ with $\mathsf{C} \cap \mathsf{X} \neq \emptyset$.
- r is a nonsmooth convex function with relatively simple structure. For example, it could be a norm regularization like the celebrated ℓ_1-regularizer $r(x) = \|x\|_1$. This regularizer plays a fundamental role in high-dimensional statistics (Bühlmann and van de Geer, 2011) and signal processing (Bruckstein et al., 2009; Daubechies et al., 2004).

For characterizing solutions to our problem (P), define the *tangent cone* associated with the closed convex set $\mathsf{X} \subseteq \mathsf{V}$ as

$$\mathsf{TC}_{\mathsf{X}}(x) := \begin{cases} \{v = t(x' - x) | x' \in \mathsf{X}, t \geq 0\} \subseteq \mathsf{V} & \text{if } x \in \mathsf{X}, \\ \emptyset & \text{else} \end{cases}$$

and the *normal cone*

$$\mathsf{NC}_{\mathsf{X}}(x) := \begin{cases} \{p \in \mathsf{V}^* | \sup_{v \in \mathsf{TC}_{\mathsf{X}}(x)} \langle p, v\rangle \leq 0\} & \text{if } x \in \mathsf{X} \\ \emptyset & \text{else.} \end{cases}$$

We remark that $\partial \delta_{\mathsf{X}}(x) = \mathsf{NC}_{\mathsf{X}}(x)$ for all $x \in \mathsf{X}$.

Given the feasible set $\mathsf{X} \subseteq \mathsf{V}$, we denote the value function

$$\Psi_{\min}(\mathsf{X}) := \inf_{x \in \mathsf{X}} \Psi(x). \tag{2.3}$$

We are focusing in this survey on problems which are solvable.

Assumption 3. $\mathsf{X}^* := \{x \in \mathsf{X} | \Psi(x) = \Psi_{\min}(\mathsf{X})\} \neq \emptyset$.

Given the standing hypothesis on the functions f and r, it is easy to see that X^* is always a closed convex set. Moreover, if $\mu > 0$, then problem (P) is *strongly convex*, and so X^* is a singleton.

Optimality conditions for problem (P) can be formulated using differential calculus tools from convex analysis. (P) can be treated as an unconstrained problem by augmenting the objective function $\Psi = r + f$ by the non-smooth penalty δ_{X}. Our main problem becomes then

$$\min_{x \in \mathsf{V}} (f(x) + r(x) + \delta_{\mathsf{X}}(x)). \tag{2.4}$$

Fermat's rule says that $x^* \in \mathsf{V}$ is a solution to (2.4) if and only if $0 \in \partial\Psi(x^*) + \mathsf{NC}_{\mathsf{X}}(x^*)$ (Rockafellar and Wets (1998, Theorem 8.15)). In general, the subdifferential operator is not linear, and we only have a "fuzzy sum rule" $\partial\Psi(x) \subseteq \nabla f(x) + \partial r(x)$. However, we know that r is finite at x^* and f is smooth on a neighborhood containing X (Assumptions 1(b)). Hence, by Rockafellar and Wets (1998, Exercise 8.8c), we have $\partial\Psi(x^*) = \nabla f(x^*) + \partial r(x^*)$. Therefore, Fermat's optimality condition becomes

$$0 \in \nabla f(x^*) + \partial r(x^*) + \mathsf{NC}_{\mathsf{X}}(x^*). \tag{2.5}$$

This means that there exists $\xi \in \partial r(x^*)$ such that

$$\langle \nabla f(x^*) + \xi, v\rangle \geq 0 \qquad \forall v \in \mathsf{TC}_{\mathsf{X}}(x^*). \tag{2.6}$$

The structured composite optimization problem (P) has attracted a lot of interest in convex programming over the last 20 years motivated by a number of important applications, see, e.g. Combettes and Wajs (2005). This led to a rich interplay between convex programming on the one hand and machine learning and signal/image processing on the other hand. Indeed, several work-horse models in these application domains are of the composite type

$$\Psi(x) = g(\mathbf{A}x) + r(x) \tag{2.7}$$

where $g : \mathsf{E} \to \mathbb{R}$ is a smooth function defined on a finite-dimensional set E (usually of lower dimension than V), and $\mathbf{A} \in \mathsf{BL}(\mathsf{V}, \mathsf{E})$ is bounded linear operator mapping points $x \in \mathsf{V}$ to elements $\mathbf{A}x \in \mathsf{E}$. Convexity allows us to switch between primal and dual formulations freely, so that the above problem can be equivalently considered as a convex-concave minimax problem

$$\min_{x \in \mathsf{X}} \max_{y \in \mathsf{E}} \{r(x) + \langle \mathbf{A}x, y\rangle - g^*(y)\} \tag{2.8}$$

Such minimax formulations have been of key importance in signal processing and machine learning (Juditsky et al., 2013; Juditsky and Nemirovski, 2011b), game theory (Sorin, 2000), decomposition methods (Tseng, 1991) and its very recent innovation around generative adversarial networks (Goodfellow et al., 2014).

Another canonical class of optimization problems in machine learning is the finite-sum model

$$\Psi(x) = \frac{1}{N}\sum_{i=1}^{N} f_i(x) + r(x), \tag{2.9}$$

which comes from supervised learning, where $f_i(x)$ corresponds to the loss incurred on the i-th data sample using a hypothesis parameterized by the decision variable x. Typically, N is an extremely large number as it corresponds to the size of the data set. The recent literature on variance reduction techniques and distributed optimization is very active in making such large scale optimization problems tractable. Surveys on the latest developments in these fields can be found in Gower et al. (2020) and the recent comprehensive textbook Lan (2020).

3. The Proximal Gradient Method

3.1. Motivation

In the context of the composite optimization problem (P), a classical and very powerful idea is to construct numerical optimization methods by exploiting problem structure. Following this philosophy, we determine the position of the next iterate by minimizing the sum of the linearization of the smooth part, the non-smooth part $r \in \Gamma_0(\mathsf{V})$, and a quadratic regularization term with weight $\gamma > 0$:

$$x^+(\gamma) = \operatorname*{argmin}_{u \in \mathsf{X}} \{f(x) + \langle \nabla f(x), u - x\rangle + r(u) + \frac{1}{2\gamma}\|u - x\|_2^2\}. \tag{3.1}$$

Disregarding terms which do not influence the computation of the solution of this strongly convex minimization problem, and absorbing the set constraint into the non-smooth part by defining $\phi(x) = r(x) + \delta_{\mathsf{X}}(x)$, we see that (3.1) can be equivalently written as

$$x^+(\gamma) = \operatorname*{argmin}_{u \in \mathsf{V}} \left\{\gamma\phi(u) + \frac{1}{2}\|u - (x - \gamma\nabla f(x))\|_2^2\right\}. \tag{3.2}$$

This way of writing the updating scheme immediately reveals some interesting geometric principles acting here. Indeed, if r would be a finite constant on X (say 0 for concreteness), then the rule (3.2) is nothing else than the Euclidean projection of the directional vector $x - \gamma\nabla f(x)$ onto the set X. In this case, the minimization routine returns the classical projected gradient step $x^+(\gamma) = P_{\mathsf{X}}(x - \gamma\nabla f(x))$, where $P_{\mathsf{X}}(x) = \operatorname{argmin}_{y \in \mathsf{X}} \frac{1}{2}\|y - x\|_2^2$. Iterating the map $x \mapsto P_{\mathsf{X}} \circ (\mathrm{Id} - \gamma\nabla f)$ generates the classical *gradient projection method*. A new obstacle arises in cases where the non-smooth function r is non-trivial over the domain X. A fundamental idea, going back to Moreau (1965), is to define the *proximity operator* $\operatorname{Prox}_{\phi} : \mathsf{V} \to \mathsf{V}$ associated with a function $\phi \in \Gamma_0(\mathsf{V})$

as[1]

$$\mathrm{Prox}_{\phi}(x) := \operatorname*{argmin}_{u\in \mathsf{V}} \left\{ \phi(u) + \frac{1}{2}\|u-x\|_2^2 \right\}. \tag{3.3}$$

Remark 3.1. The classical Moreau proximity operator of ϕ is, in general, explicitly computable when ϕ is norm like, or when ϕ is the indicator function of sets whose geometry is favorable to Euclidean projections. Although quite frequent in applications (orthant, second-order cone, ℓ_1 norm), these prox-friendly functions are very scarce, see, e.g., Combettes and Wajs (2005, Section 2.6). A significant improvement will be made in Section 3.2, where a general Bregman proximal framework will be introduced.

The value function

$$\phi_\gamma(x) = \inf_u \{\phi(u) + \frac{1}{2\gamma}\|u-x\|^2\}$$

is called the *Moreau envelope* of the function ϕ, and is an important smoothing and regularization tool, frequently employed in numerical analysis. Indeed, for a function $\phi \in \Gamma_0(\mathsf{V})$ and $\gamma > 0$, its Moreau envelope is finite everywhere, convex and has γ^{-1}-Lipschitz continuous gradient on V given by $\nabla\phi_\gamma(x) = \frac{1}{\gamma}(x - \mathrm{Prox}_{\gamma\phi}(x))$ Bauschke and Combettes (2016, Prop. 12.30).

In the context of minimization the composite model $\Psi = f + r$, the proximity operator is the key actor in generating a large family of gradient-based methods. Choosing $\phi = r + \delta_{\mathsf{X}}$ in (3.3), and replacing the generic input with the specific input $x - \gamma\nabla f(x)$, we are in the framework of the *Proximal Gradient Method* (PGM). PGM is a very powerful method which received enormous interest in optimization and its applications. For a survey in the context of signal processing we refer the reader to Combettes and Pesquet (2011). A general survey on proximal operators can be found in Beck (2017); Parikh and Boyd (2014).

The Proximal Gradient Method (PGM)
Input: $x^0 \in \mathsf{X}$.
General step: For $k = 0, 1, \ldots$ do:
Choose $\gamma_k > 0$.
set $x^{k+1} = \mathrm{Prox}_{\gamma_k\phi}\left(x^k - \gamma_k\nabla f(x^k)\right)$.

Remark 3.2. In the fully non-smooth case, i.e. when $f = 0$ in our model problem, PGM reduces to a classical recursion known as the *proximal point* method:

$$x^{k+1} = \mathrm{Prox}_{\gamma_k\phi}(x^k) = \operatorname*{argmin}_{u\in\mathsf{V}}\{\phi(u) + \frac{1}{2\gamma_k}\|u-x^k\|^2\}.$$

This scheme has been first proposed by Martinet (1970) and Rockafellar (1976b).

3.2. Bregman Proximal Setup

The basic idea behind non-Euclidean extensions of PGM is to replace the ℓ_2-norm $\frac{1}{2}\|u-x\|^2$ by a different distance-like function which is tailored to the geometry of the feasible set $\mathsf{X} \subseteq \mathsf{V}$. These non-Euclidean distance-like functions that will be used are *Bregman divergences*. The transition from Euclidean to non-Euclidean distance measures is motivated by the usefulness and flexibility of the latter in computational perspectives and potentials for improving convergence properties for specific application domains. In particular, the move from Euclidean to non-Euclidean distance measures allows to adapt the algorithm to the underlying geometry, typically explicitly embodied in the description of the domain X, see e.g. Auslender and Teboulle (2009). This can not only positively affect the per-iteration complexity, but also will have a footprint on the overall iteration complexity of the method, as we will demonstrate in this section. The point of departure of Bregman Proximal algorithms is to introduce a *distance generating function* $h : \mathsf{V} \to (-\infty, \infty]$, which is a barrier-type mapping suitably chosen to capture geometric features of the set X.

Definition 3.1. Let X be a closed convex subset of V. We say that $h \in \Gamma_0(\mathsf{V})$ is a *distance generating function (DGF)* with modulus $\alpha > 0$ with respect to $\|\cdot\|$ on X if

(a) h is closed, convex and proper;
(b) $\mathsf{X} \subseteq \mathrm{dom}\, h$;
(c) the set $\mathsf{X}^\circ = \{x \in \mathsf{X} | \partial h(x) \neq \emptyset\}$ is nonempty and convex;
(d) h restricted to X° is continuously differentiable and strongly convex under the norm $\|\cdot\|$ with parameter α:

$$(\forall x, x' \in \mathsf{X}^\circ): \quad \langle \nabla h(x) - \nabla h(x'), x - x'\rangle \geq \alpha\|x-x'\|^2.$$

We denote by $\mathcal{H}_\alpha(\mathsf{X})$ the set of DGFs on X.

From classical differential theory of convex functions Rockafellar (1970, Section 23-25), we know that $\mathrm{dom}(\partial h) \subset \mathrm{dom}\, h$. Hence, $\mathsf{X}^\circ = \mathrm{dom}(\partial h) \cap \mathsf{X}$ is contained in the set $\mathrm{dom}\, h \cap \mathsf{X}$, which in turn agrees with X thanks to property (b). Restricted to X° the DGF h is continuously differentiable.

In many proximal settings we are interested in DGFs which act as barriers on the feasible set X. Naturally, the barrier properties of the function h are captured by its scaling near $\mathrm{bd}(\mathsf{X})$, usually encoded in terms of the notion of *essential smoothness* (Rockafellar (1970, Section 26).)

Definition 3.2 (Essential smoothness)**.** $h \in \mathcal{H}_\alpha(\mathsf{X})$ is *essentially smooth* if it satisfies the following three conditions:

(a) $\mathrm{int}(\mathrm{dom}\, h) \neq \emptyset$;
(b) h is differentiable throughout $\mathrm{int}(\mathrm{dom}\, h)$;
(c) $\lim_{i\to\infty} \|\nabla h(x_i)\| = +\infty$ whenever $x_1, x_2, \ldots$ is a sequence in $\mathrm{int}(\mathrm{dom}\, h)$ converging to a boundary point x of $\mathrm{int}(\mathrm{dom}\, h)$.

Given $h \in \mathcal{H}_\alpha(\mathsf{X})$, its *Bregman divergence* $D_h : \mathrm{dom}\, h \times \mathrm{dom}(\partial h) \to \mathbb{R}$ is defined as

$$D_h(u,x) := h(u) - h(x) - \langle \nabla h(x), u - x\rangle. \tag{3.4}$$

The α-strong convexity of the DGF ensures that

$$(\forall x \in \mathrm{dom}(\partial h), \forall u \in \mathrm{dom}\, h): \quad D_h(u,x) \geq \frac{\alpha}{2}\|u-x\|^2. \tag{3.5}$$

Hence, $D_h(x,x) = 0$ for $x \in \mathrm{dom}(\partial h)$, but in general it is not a symmetric function and it does not satisfy a triangle inequality. This disqualifies D_h from carrying the label of a metric, but it can still be interpreted as a distance measure.

The convex conjugate $h^*(y) = \sup_{x\in\mathsf{V}}\{\langle x, y\rangle - h(x)\}$ for a function $h \in \mathcal{H}_\alpha(\mathsf{X})$ is known to be differentiable on V^* with a $\frac{1}{\alpha}$-Lipschitz continuous gradient (Rockafellar and Wets (1998, Proposition 12.60)):

$$h^*(y_2) \leq h^*(y_1) + \langle \nabla h^*(y_1), y_2 - y_1\rangle + \frac{1}{2\alpha}\|y_2 - y_1\|_*^2. \tag{3.6}$$

for all $y_1, y_2 \in \mathsf{V}^*$.

It will be instructive to go over some standard examples of distance generating functions. See also Combettes and Wajs (2005), Bauschke and Combettes (2016), and Ben-Tal and Nemirovski (2020).

Example 3.1 (Euclidean Projection)**.** We begin by revisiting the ℓ_2-projection on some closed convex subset $\mathsf{X} \subset \mathsf{V} = \mathbb{R}^n$. Letting $h(x) = \frac{1}{2}\|x\|_2^2$ for $x \in \mathsf{V}$, we readily see that $\mathsf{X}^\circ = \mathsf{X}$. Moreover, for $x \in \mathsf{X}^\circ$, the DGF is 1-strongly convex and continuously differentiable with $\nabla h(x) = x$. The associated Bregman divergence is the Euclidean distance $D_h(u,x) = \frac{1}{2}\|u-x\|_2^2$ for all $u, x \in \mathsf{X}$.

[1] The repository http://proximity-operator.net/index.html provides codes and explicit expressions for proximity operators of many standard functions. A useful MATLAB implementation of proximal methods is described in Beck and Guttmann-Beck (2019).

Example 3.2 (Entropic Regularization). Let $\mathsf{X} = \{x \in \mathbb{R}^n_+ | \sum_{i=1}^n x_i = 1\} = \mathbb{R}^n_+ \cap \{x \in \mathbb{R}^n | \sum_{i=1}^n x_i = 1\}$ denote the unit simplex in $\mathsf{V} = \mathbb{R}^n$. Define the function $\psi : \mathbb{R} \to [0, \infty]$ as

$$\psi(t) = \begin{cases} t\ln(t) - t & \text{if } t > 0, \\ 0 & \text{if } t = 0, \\ +\infty & \text{else.} \end{cases}$$

As DGF consider the *Boltzmann-Shannon entropy* $h(x) := \sum_{i=1}^n \psi(x_i)$. Endowing the ground space V with the ℓ_1 norm $\|\cdot\| = \|\cdot\|_1$, it can be shown that $h \in \mathcal{H}_1(\mathsf{X})$ with $\operatorname{dom} h = \mathbb{R}^n_+$ and $\mathsf{X}^\circ = \{x \in \mathbb{R}^n_{++} | \sum_{i=1}^n x_i = 1\}$. Indeed, on X° the function h is continuously differentiable with $\nabla h(x) = [\ln(x_1), \dots, \ln(x_n)]^\top$. Furthermore, $\partial h(0) = \varnothing$, so that $\operatorname{dom}(\partial h) = \mathbb{R}^n_{++}$. The resulting Bregman divergence is the *Kullback-Leibler divergence*

$$D_h(u, x) = \sum_{i=1}^n u_i \ln\left(\frac{u_i}{x_i}\right) + \sum_{i=1}^n (x_i - u_i).$$

Example 3.3 (Box Constraints). Let $\mathsf{V} = \mathbb{R}^n$ and $\mathsf{X} = \prod_{i=1}^n [a_i, b_i]$, where $0 \le a_i \le b_i$. Given parameters $0 \le a \le b$, define the *Fermi-Dirac entropy*

$$\psi_{a,b}(t) := \begin{cases} (t-a)\ln(t-a) + (b-t)\ln(b-t) & \text{if } t \in (a, b), \\ 0 & \text{if } t \in \{a, b\}, \\ +\infty & \text{else} \end{cases}$$

Note that for $t \in (a, b)$, we have $\psi'_{a,b}(t) = \ln\left(\frac{t-a}{b-t}\right)$, and $\partial\psi_{a,b}(t) = \varnothing$ for $t \in \mathbb{R} \setminus (a, b)$. Accordingly, the function $h(x) = \sum_{i=1}^n \psi_{a_i,b_i}(x_i)$ is a DGF on $\mathsf{X} = \operatorname{dom} h$ with $\mathsf{X}^\circ = \prod_{i=1}^n (a_i, b_i)$. On X°, the gradient mapping is $\nabla h(x) = [\psi'_{a_1,b_1}(x_1), \dots, \psi'_{a_n,b_n}(x_n)]^\top$.

Example 3.4 (Semidefinite Constraints). Let $\mathsf{V} = \mathsf{S}^n$ be the set of real symmetric matrices and $\mathsf{X} = \mathsf{S}^n_+$ be the cone of real symmetric positive semi-definite matrices equipped with the inner product $\langle \mathbf{A}, \mathbf{B}\rangle = \operatorname{tr}(\mathbf{AB})$. Define the negative von Neumann entropy $h(\mathbf{X}) = \operatorname{tr}[\mathbf{X}\log(\mathbf{X})]$, which can be seen as the matrix-equivalent of the negative Boltzmann-Shannon entropy. It can be verified that $\operatorname{dom} h = \mathsf{X}$ and $\nabla h(\mathbf{X}) = \log(\mathbf{X}) + \mathbf{I}$ for $\mathbf{X} \in \mathsf{S}^n_{++}$. Hence, $\operatorname{dom} h = \mathsf{X}$, and $\mathsf{X}^\circ = \mathsf{S}^n_{++}$, the cone of positive definite matrices. For $\mathbf{X}' \in \mathsf{S}^n_{++}$, the corresponding Bregman divergence is given by

$$D_h(\mathbf{X}', \mathbf{X}) = \operatorname{tr}[\mathbf{X}'\log(\mathbf{X}') - \mathbf{X}'\log(\mathbf{X}) + \mathbf{X}' - \mathbf{X}]$$

See Doljansky and Teboulle (1998) for further examples on matrix domains. Pinsker's inequality (Cesa-Bianchi and Lugosi (2006)) says that h is $\frac{1}{2}$-strongly convex with respect to the nuclear norm $\|\mathbf{X}\|_1 = \sum_i |\lambda_i(\mathbf{X})|$ for $\mathbf{X} \in \mathsf{S}^n$, i.e.

$$D_h(\mathbf{X}', \mathbf{X}) \ge \frac{1}{2}\|\mathbf{X}' - \mathbf{X}\|_1.$$

Example 3.5 (2nd order cone constraints). Let $\mathsf{V} = \mathbb{R}^n$ and $L^n_{++} := \{x \in \mathsf{V} | x_n > (x_1^2 + \dots + x_{n-1}^2)^{1/2}\}$ the interior of the second-order cone. Let $\mathsf{X} = \operatorname{cl}(L^n_{++})$. Denote by $\mathbf{J}_n$ be the $n \times n$ diagonal matrix with -1 in its first $n-1$ diagonal entries and 1 in the last one. Define $h(x) = -\ln(\langle \mathbf{J}_n x, x\rangle) + \frac{\alpha}{2}\|x\|_2^2$. Then $h \in \mathcal{H}_\alpha(\mathsf{X})$ with $\operatorname{dom} h = \mathsf{X}^\circ = L^n_{++} \subset \mathsf{X}$. The associated Bregman divergence is

$$D_h(x, u) = -\ln\left(\frac{\langle \mathbf{J}_n x, x\rangle}{\langle \mathbf{J}_n u, u\rangle}\right) + 2\frac{\langle \mathbf{J}_n x, u\rangle}{\langle \mathbf{J}_n u, u\rangle} - 2 + \frac{\alpha}{2}\|x - u\|_2^2.$$

The proximal framework for general conic constraints has been developed in Auslender and Teboulle (2006b).

Once we endow our set X with a DGF, the technology generating a gradient method in this non-Euclidean setting is the *Bregman proximal operator* (Teboulle, 1992) applied to the function $\phi \in \Gamma_0(\mathsf{V})$:

$$\operatorname{Prox}^h_\phi(x) := \operatorname*{argmin}_{u \in \mathsf{V}} \{\phi(u) + D_h(u, x)\}. \tag{3.7}$$

If $\phi(x) = \delta_{\mathsf{X}}(x)$ is the indicator function of the closed convex set $\mathsf{X} \subset \mathsf{V}$, then the Bregman proximal operator defines the *Bregman projection* (Bauschke et al., 2003; Censor and Zenios, 1992) onto the set X. It should be pointed out that under the standard Euclidean setup described in Example 3.1 the Bregman proximal operator boils down to the Moreau proximal operator (3.3). As we already alluded to, the main rationale for the introduction of Bregman proximal operators is that it allows us to define a projection framework which can be adapted to the geometry of X. Below, we give examples for which $\operatorname{Prox}^h_\phi(x)$ is easy to compute in closed form, whereas the standard Moreau proximal map is not explicitly known (and would thus require a numerical procedure, implying a nested scheme if used in an algorithm).

Example 3.6. In the following examples we assume that $\mathsf{V} = \mathbb{R}$ for simplicity.

1. Let $\phi(x) = \gamma|x - \xi|$ where $\gamma, \xi > 0$. Take $h(x) = x\ln(x)$, $\operatorname{dom} h = [0, \infty)$. Then
$$\operatorname{Prox}^h_\phi(x) = \begin{cases} \exp(\gamma)x & \text{if } x < \exp(-\gamma)\xi, \\ \xi & \text{if } x \in [\exp(-\gamma)\xi, \exp(\gamma)\xi], \\ \exp(-\gamma)x & \text{if } x > \exp(\gamma)\xi. \end{cases}$$
2. Let $\phi(x) = \frac{\gamma}{2}x^2$ for $\gamma > 0$, and $h(x) = -\ln(x)$, $\operatorname{dom} h = (0, \infty)$. Then $\operatorname{Prox}_\phi(x) = \frac{1}{2\gamma x}\left(\sqrt{1 + 4\gamma x^2} - 1\right)$.

3.3. Bregman proximal gradient method

For solving our main problem (P), a special selection of the function $\phi \in \Gamma_0(\mathsf{V})$ in (3.7) is $\phi(u) = \gamma(r + \delta_{\mathsf{X}})(u) + \langle \gamma\nabla f(x), u - x\rangle$. Replacing the gradient $\gamma\nabla f(x)$ with a general dual vector $y \in \mathsf{V}^*$, we obtain the *prox-mapping*

$$\mathcal{P}^h_{\gamma r}(x, y) := \operatorname*{argmin}_{u \in \mathsf{X}} \{\gamma r(u) + \langle y, u - x\rangle + D_h(u, x)\}. \tag{3.8}$$

The prox-mapping takes as inputs a "primal-dual" pair $(x, y) \in \mathsf{X}^\circ \times \mathsf{V}^*$ where x is the current iterate, and y is a dual variable representing a "gradient signal" we obtain on the smooth part of the minimization problem (P) (usually obtained after consulting a black-box oracle). Various conditions on the well-posedness of the prox-mapping have been stated in the literature. We will not repeat them here, but rather refer to the recent survey (Teboulle, 2018). Below we give some examples.

Example 3.7 (Moreau Proximal Operator). Let $\mathsf{V} = \mathbb{R}^n$ and X a nonempty, closed and convex set in V. Let $\|\cdot\| = \|\cdot\|_2$, and $h(x) = \frac{1}{2}\|x\|_2^2$. The prox-mapping (3.8) reduces in this case to

$$\mathcal{P}^h_{\gamma r}(x, y) = \operatorname{Prox}_{\gamma(r + \delta_{\mathsf{X}})}(x - y).$$

Example 3.8 (Simplex Constraints). Let $\mathsf{V} = \mathbb{R}^n$ with ℓ_1-norm $\|\cdot\| = \|\cdot\|_1$. Consider the set $\mathsf{X} = \{x \in \mathbb{R}^n_+ | \sum_{i=1}^n x_i = 1\}$ and endow this set with the Boltzmann-Shannon entropy $h(x) = \sum_{i=1}^n x_i \ln(x_i)$. For $r(x) = 0$, a standard calculation gives rise to the prox-mapping

$$[\mathcal{P}^h_{\delta_{\mathsf{X}}}(x, y)]_i = \frac{x_i e^{y_i}}{\sum_{j=1}^n x_j e^{y_j}} \qquad 1 \le i \le n, x \in \mathsf{X}, y \in \mathsf{V}^*. \tag{3.9}$$

This mapping plays a key role in optimization, where it is known as exponentiated gradient descent (Beck and Teboulle, 2003; Juditsky et al., 2005).

Example 3.9 (Box Constraints). Consider the setting introduced in Example 3.3. For $r(x) = 0$, one can compute

$$[\mathcal{P}^h_{\delta_{\mathsf{X}}}(x, y)]_i = a_i + \frac{b_i - a_i}{1 + \frac{b_i - x_i}{x_i - a_i}\exp(y_i)} \qquad 1 \le i \le n, x \in \mathsf{X}, y \in \mathsf{V}^*.$$

If $\gamma > 0$ is a step-size parameter and $y = \gamma\nabla f(x)$, then we obtain the *Bregman proximal map* $T^h_\gamma(x) := \mathcal{P}^h_{\gamma r}(x, \gamma\nabla f(x))$ for all $x \in \mathsf{X}$. Iterating this map generates a discrete-time dynamical system known as the *Bregman proximal gradient method* (BPGM).

Remark 3.3. For simple implementation, BPGM relies on the structural assumption that the prox-mapping $\mathcal{P}^h_r(x, y)$ can be evaluated efficiently

The Bregman Proximal Gradient Method (BPGM)
Input: $h \in \mathcal{H}_\alpha(\mathsf{X})$. Pick $x^0 \in \mathrm{dom}(r) \cap \mathsf{X}^\circ$.
General step: For $k = 0, 1, \ldots$ do:
choose $\gamma_k > 0$.
update $x^{k+1} = \mathcal{P}^h_{\gamma_k r}(x^k, \gamma_k \nabla f(x^k))$.

on the trajectory $\{(x^k, \gamma_k \nabla f(x^k)) | 0 \le k \le K \in \mathbb{N}^*\}$. This, often somewhat hidden, assumption is known in the literature as the "prox-friendliness" assumption, a terminology apparently coined by Cox et al. (2014)).

3.3.1. Basic Complexity Properties

To assess the iteration complexity of BPGM, let us start with some preparatory estimates. The first-order optimality condition for the point $x^+ = \mathcal{P}^h_{\gamma r}(x, \gamma \nabla f(x))$ is given by

$$0 \in \gamma \partial r(x^+) + \gamma \nabla f(x) + \nabla h(x^+) - \nabla h(x) + \mathsf{NC}_{\mathsf{X}}(x^+)$$
$$= \gamma(\partial r + \mathsf{NC}_{\mathsf{X}})(x^+) + \gamma \nabla f(x) + \nabla h(x^+) - \nabla h(x).$$

Whence, there exists $\xi \in \partial r(x^+)$ such that, for all $u \in \mathsf{X}$,

$$\langle \gamma \xi + \gamma \nabla f(x) + \nabla h(x^+) - \nabla h(x), x^+ - u \rangle \le 0. \tag{3.10}$$

Via the subgradient inequality for the convex function $x \mapsto \phi(x)$, we obtain for all $u \in \mathsf{X}$:

$$r(x^+) - r(u) \le \langle \nabla f(x), u - x^+ \rangle + \frac{1}{\gamma} \langle \nabla h(x^+) - \nabla h(x), u - x^+ \rangle. \tag{3.11}$$

For further analysis, we need the celebrated three-point identity, due to Chen and Teboulle (1993), whose simple proof we omit.

Lemma 3.3 (3-point lemma). *For all $x, y \in \mathsf{X}^\circ$ and $z \in \mathrm{dom}\, h$ we have*

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

□

Thanks to Lemma 3.3, relation (3.11) reads as

$$\gamma(r(x^+) - r(u)) \le \gamma \langle \nabla f(x), u - x^+ \rangle + D_h(u, x) - D_h(u, x^+) - D_h(x^+, x) \tag{3.12}$$

for all $u \in \mathrm{dom}\, h \cap \mathsf{X}$.

Remark 3.4. Note that if x^+ is calculated inexactly in the sense that instead of (3.10), for some $\xi \in \partial r(x^+)$, it holds that

$$\langle \gamma \xi + \gamma \nabla f(x) + \nabla h(x^+) - \nabla h(x), x^+ - u \rangle \le \Delta \tag{3.13}$$

for some $\Delta \ge 0$, then instead of (3.12) we have

$$\gamma(r(x^+) - r(u)) \le \gamma \langle \nabla f(x), u - x^+ \rangle + D_h(u, x) - D_h(u, x^+) - D_h(x^+, x) + \Delta. \tag{3.14}$$

See Auslender and Teboulle (2006b) for an explicit analysis of the error-prone implementation.

Assumption 1(a) gives rise to the the classical "descent Lemma" (Nesterov, 2018b):

$$f(x^+) \le f(x) + \langle \nabla f(x), x^+ - x \rangle + \frac{L_f}{2} \|x^+ - x\|^2. \tag{3.15}$$

Additionally, for all $u \in \mathsf{X}$, differential convexity of f on V implies (cf. (1.1))

$$f(u) \ge f(x) + \langle \nabla f(x), u - x \rangle. \tag{3.16}$$

This allows us to bound

$$\langle \nabla f(x), u - x^+ \rangle = \langle \nabla f(x), x - x^+ \rangle + \langle \nabla f(x), u - x \rangle$$
$$\overset{(3.15)}{\le} f(x) - f(x^+) + \frac{L_f}{2} \|x^+ - x\|^2 + \langle \nabla f(x), u - x \rangle$$
$$\overset{(3.16)}{\le} f(u) - f(x^+) + \frac{L_f}{2} \|x^+ - x\|^2$$
$$\overset{(3.5)}{\le} f(u) - f(x^+) + \frac{L_f}{\alpha} D_h(x^+, x).$$

Using this estimate in relation (3.12), we obtain, for all $u \in \mathrm{dom}\, h \cap \mathsf{X}$,

$$\gamma(\Psi(x^+) - \Psi(u)) \le D_h(u, x) - D_h(u, x^+) - \left(1 - \frac{\gamma L_f}{\alpha}\right) D_h(x^+, x). \tag{3.17}$$

If $\gamma \in (0, \frac{\alpha}{L_f}]$, then the above yields

$$\gamma(\Psi(x^+) - \Psi(u)) \le D_h(u, x) - D_h(u, x^+), \quad u \in \mathrm{dom}\, h \cap \mathsf{X}.$$

Setting $x = x^k, x^+ = x^{k+1}$ and $\gamma = \gamma_k$, one can reformulate the previous display as

$$\gamma_k \left(\Psi(x^{k+1}) - \Psi(u)\right) \le D_h(u, x^k) - D_h(u, x^{k+1}), \quad u \in \mathrm{dom}\, h \cap \mathsf{X}.$$

Choosing $u = x^k$, we readily see $\gamma_k(\Psi(x^{k+1}) - \Psi(x^k)) \le -D_h(x^k, x^{k+1}) \le 0$, i.e. the sequence of function values $\{\Psi(x^k)\}_{k \in \mathbb{N}}$ is non-increasing. On the other hand, for a general reference point $u \in \mathrm{dom}\, h \cap \mathsf{X}$, we also see that

$$\sum_{k=0}^{N-1} \left(\Psi(x^{k+1}) - \Psi(u)\right) \le \sum_{k=0}^{N-1} \frac{1}{\gamma_k} \left[D_h(u, x^k) - D_h(u, x^{k+1})\right]$$
$$= \frac{1}{\gamma_0} D_h(u, x^0) - \frac{1}{\gamma_{N-1}} D_h(u, x^N) + \sum_{k=0}^{N-2} \left(\frac{1}{\gamma_{k+1}} - \frac{1}{\gamma_k}\right) D_h(u, x^{k+1}).$$

Assuming a constant step size policy $\gamma_k = \gamma$, this gives us

$$\sum_{k=0}^{N-1} \left(\Psi(x^{k+1}) - \Psi(u)\right) \le \frac{1}{\gamma} D_h(u, x^0).$$

Define the function gap $s^k := \Psi(x^k) - \Psi(u)$, then $s^{k+1} - s^k = \Psi(x^{k+1}) - \Psi(x^k) \le 0$, and therefore

$$s^N \le \frac{1}{N} \sum_{k=0}^{N-1} s^{k+1} = \frac{1}{N} \sum_{k=0}^{N-1} [\Psi(x^{k+1}) - \Psi(u)] \le \frac{1}{N\gamma} D_h(u, x^0)$$

for all $u \in \mathrm{dom}\, h \cap \mathsf{X}$. As an attractive step size choice, we may take the greedy choice $\gamma = \frac{\alpha}{L_f}$. However, we need to know the Lipschitz constant of the gradient map of the smooth part f of the minimization problem (P) to make this an implementable solution strategy.

Proposition 3.4. *Consider problem (P) with Assumptions 1-3 in place. Let $h \in \mathcal{H}_\alpha(\mathsf{X})$. If BPGM is run with the constant step size $\gamma_k = \frac{\alpha}{L_f}$, then for any $x^* \in \mathsf{X}^*$, we have*

$$\Psi(x^k) - \Psi_{\min}(\mathsf{X}) \le \frac{L_f}{\alpha k} D_h(x^*, x^0). \tag{3.18}$$

This global sublinear rate of convergence for the Euclidean setting is due to Beck and Teboulle (2009b); Nesterov (2013).

Remark 3.5. Under additional assumption that the objective Ψ is μ-strongly-convex with $\mu > 0$ it is possible to obtain linear convergence rate of BPGM, i.e. $\Psi(x^k) - \Psi_{\min}(\mathsf{X}) \le 2L_f \exp(-k\mu/L_f) D_h(x^*, x^0)$.

3.3.2. Subgradient and Mirror Descent

In the previous subsections we focused on the setting of problem (P) with smooth part f and obtained for BPGM a convergence rate $O(1/k)$. The same method actually works for non-smooth convex optimization problems when f has bounded subgradients In this setting BPGM with a different choice of the step-size $(\gamma_k)_k$ is known as the *Mirror Descent* (MD) method (Nemirovski and Yudin, 1983). A version of this method for convex composite non-smooth optimization was proposed in Duchi et al. (2010), and an overview of Subgradient/Mirror Descent type of methods for non-smooth problems can be found in Beck (2017); Dvurechensky et al. (2020b); Lan (2020). The main difference between BPGM and MD is that one replaces the assumption that ∇f is Lipschitz continuous with the assumption that f is subdifferentiable with bounded subgradients, i.e. $\|f'(x)\|_* \le M_f$ for all $x \in \mathsf{X}$ and $f'(x) \in \partial f(x)$. For a given sequence of step-sizes $(\gamma_k)_k$ one defines the next test point as

$$x^{k+1} = \operatorname*{argmin}_u \left\{\langle \gamma_k f'(x^k), u - x^k \rangle + \gamma_k r(u) + D_h(u, x^k)\right\} = \mathcal{P}^h_{\gamma_k r}(x^k, \gamma_k f'(x^k)).$$

A typical choice for the step size sequence is a monotonically decreasing policy like $\gamma_k \sim k^{-1/2}$. Under such a specification, the MD sequence $(x^k)_k$ can be shown to converge with rate $O(1/\sqrt{k})$ to the solution, which is optimal in this setting. A proof of this result can be patterned via a suitable adaption of the arguments employed in our analysis of the Dual Averaging Method in Section 3.4.

3.3.3. Potential Improvements due to relative smoothness

A key pillar of the complexity analysis of BPGM was the descent inequality (3.15), which is available thanks to the assumed Lipschitz continuity of the gradient ∇f. The influential work Bauschke et al. (2016) introduced a very clever construction which allows one to relax this restrictive assumption.[2] The elegant observation made in Bauschke et al. (2016) is that the Lipschitz-gradient-based descent lemma has the equivalent, but insightful, expression

$$\left(\frac{L_f}{2}\|x\|^2 - f(x)\right) - \left(\frac{L_f}{2}\|u\|^2 - f(u)\right) \geq \langle L_f u - \nabla f(u), x - u\rangle \qquad \forall x, u \in \mathsf{V}.$$

This is just the gradient inequality for the convex function $x \mapsto \frac{L_f}{2}\|x\|^2 - f(x)$.

Definition 3.5. The family of functions $\mathcal{E}(\mathsf{X})$ is the class of DGFs $h \in \mathcal{H}_0(\mathsf{X})$ which are of *Legendre type*: h essentially smooth and strictly convex on $\operatorname{int}\operatorname{dom} h$ with $\operatorname{cl}(\operatorname{dom} h) = \mathsf{X}$.

In this section we work in a Bregman proximal setting with Legendre type distance generating function.

Assumption 4. X has nonempty interior and $h \in \mathcal{E}(\mathsf{X})$.

Remark 3.6. For $h \in \mathcal{E}(\mathsf{X})$, it is true that $\mathsf{X}^\circ \subseteq \operatorname{int}(\operatorname{dom} h) \cap \mathsf{X}$.

Based on the general intuition we have gained while working with a general proximal setup, a very tempting and natural generalization is the following.

Definition 3.6 (Relative Smoothness, (Bauschke et al., 2016; Van Nguyen, 2017))**.** The function f is smooth relative to $h \in \mathcal{E}(\mathsf{X})$, if there exists $L_f^h > 0$ such that for any $x, u \in \mathsf{X}^\circ$

$$f(u) \leq f(x) + \langle \nabla f(x), u - x\rangle + L_f^h D_h(u, x). \tag{3.19}$$

Rearranging terms, a very concise and elegant way of writing the relative smoothness condition is $D_{L_f^h h - f}(u, x) \geq 0$ on X°. This amounts to saying that $L_f^h h - f$ is convex on X°. Clearly, if f and h are twice continuously differentiable on X°, the relative smoothness condition can be stated in terms of a positive semi-definitness condition on the set X° as

$$L_f^h \nabla^2 h(x) - \nabla^2 f(x) \succeq 0 \qquad \forall x \in \mathsf{X}^\circ. \tag{3.20}$$

Beside providing a non-Euclidean version of the descent lemma, the notion of relative smoothness allows us to rigorously apply gradient methods to problems whose smooth part admits no global Lipschitz continuous gradient. This gains relevance in solving various classes of inverse problems under Poisson noise (see Section 5.2 in Bauschke et al. (2016)), and optimal experimental design (Lu et al., 2018), a class of problems structurally equivalent to finding the minimum volume ellipsoid containing a list of vectors (Boyd and Vandenberghe, 2004; Todd, 2016).

Assumption 5. There exists a DGF $h \in \mathcal{E}(\mathsf{X})$ for which (f, h) is a relatively smooth pair.

The complexity analysis of BPGM under a relative smoothness assumption on the pair (f, h) (the so called NoLips algorithm of Bauschke et al. (2016)), proceeds analogous to the previous analysis. The first important observation is an extended version of the fundamental inequality (3.17), which reads as

$$\gamma(\Psi(x^+) - \Psi(u)) \leq D_h(u, x) - D_h(u, x^+) - (1 - \gamma L_f^h) D_h(x^+, x) \quad \forall u \in \operatorname{dom} h \cap \mathsf{X}. \tag{3.21}$$

The derivation of this inequality is analogous to inequality (3.17), replacing the Lipschitz-gradient-based descent inequality (3.15) by the relative smoothness inequality (3.19) with parameter L_f^h. The continuation of the proof differs then in one important aspect. It relies on the introduction of the *symmetry coefficient* of the DGF h as

$$\nu(h) := \inf\left\{\frac{D_h(x,u)}{D_h(u,x)} \middle| (x,u) \in \mathsf{X}^\circ \times \mathsf{X}^\circ, x \neq u\right\}. \tag{3.22}$$

The symmetry coefficient $\nu(h)$ is confined to the interval [0,1], and $\nu(h) = 1$ applies essentially only to the energy function $h(x) = \frac{1}{2}\|x\|^2$. Choosing $\gamma = \frac{1+\nu(h)}{2L_f^h}, x^+ = x^{k+1}, x = x^k$ gives

$$\gamma(\Psi(x^{k+1}) - \Psi(u)) \leq D_h(u, x^k) - D_h(u, x^{k+1}) - \frac{1 - \nu(h)}{2} D_h(x^{k+1}, x^k)$$

Setting $u = x^k$ gives descent of the function value sequence $(\Psi(x^k))_{k\geq 0}$. Moreover, it immediately follows that

$$\Psi(x^k) - \Psi(u) \leq \frac{2L_f^h}{1 + \nu(h)}\left(D_h(u, x^{k-1}) - D_h(u, x^k)\right)$$

Summing from $k = 1, 2, \ldots, N$, the same argument as for the BPGM give sublinear convergence of NoLips

$$\Psi(x^N) - \Psi(u) \leq \frac{2L_f^h}{N(1 + \nu(h))} D_h(u, x^0). \tag{3.23}$$

Comparing the constants in the complexity estimates of NoLips and BPGM we see that the relative efficiency of the two methods depends on the relative condition number $\frac{2L_f^h/(1+\nu(h))}{L_f/\alpha}$. Hence, even if the objective function is globally Lipschitz smooth (i.e. admits a Lipschitz continuous gradient), exploiting the idea of relative smoothness might lead to superior performance of NoLips.

To establish global convergence of the trajectory $(x^k)_{k\in\mathbb{N}}$, additional "reciprocity" conditions on the Bregman divergence must be imposed.

Assumption 6. The DGF $h \in \mathcal{E}(\mathsf{X})$ satisfies the Bregman reciprocity condition: The level sets $\{u \in \mathsf{X}^\circ | D_h(u, x) \leq \beta\}$ are bounded for all $\beta \in \mathbb{R}$, and $x^k \to x \in \mathsf{X}^\circ$ if and only if $\lim_{k\to\infty} D_h(x, x^k) = 0$.

This assumption is necessary, as in some settings Bregman reciprocity is violated. See Example 4.1 in Doljansky and Teboulle (1998) as a simple illustration. Under Bregman reciprocity, one can prove global convergence in the spirit of Opial's lemma (Opial, 1967):

Theorem 3.7 (Bauschke et al. (2016), Theorem 2)**.** *Suppose Assumptions 1-6 hold. Let $(x^k)_{k\in\mathbb{N}}$ be the sequence generated by BPGM with the relatively smooth pair (f, h) with $\gamma \in (0, \frac{1+\nu(h)}{L_f^h})$ and $h \in \mathcal{E}(\mathsf{X})$. Then, the sequence $(x^k)_{k\in\mathbb{N}}$ converges to some solution $x^* \in \mathsf{X}^*$.*

Under additional assumption that f is μ_f^h-relatively strongly convex (Lu et al., 2018) with $\mu_f^h > 0$, i.e. (cf. (3.20))

$$\nabla^2 f(x) - \mu_f^h \nabla^2 h(x) \succeq 0 \qquad \forall x \in \mathsf{X}^\circ, \tag{3.24}$$

it is possible to obtain linear convergence rate of BPGM, i.e. $\Psi(x^k) - \Psi_{\min}(\mathsf{X}) \leq 2L_f^h \exp(-k\mu_f^h / L_f^h) D_h(x^*, x^0)$. Stonyakin et al. (2020, 2019) show how to adapt the method to cope with inexact oracles and inexact Bregman proximal steps.

3.4. Dual Averaging

An alternative method called Dual Averaging (DA) was proposed in Nesterov (2009) and, on the contrary, is a primal-dual method making alternating updates in the space of gradients and in the space of iterates.

[2] Variations on the same theme can be found in Lu et al. (2018) and further developments can be found in Bùi and Combettes (2021); Stonyakin et al. (2020).

Below we give a self-contained complexity analysis of this scheme for non-smooth optimization. The following assumptions shall be in place:

Assumption 7. X is a nonempty convex compact set.

Assumption 8. $r = 0$ and $f \in \Gamma_0(\mathsf{V})$ with $\mathsf{X} \subset \mathrm{dom}(f)$. Furthermore, $\sup_{x\in\mathsf{X}} \|f'(x)\|_* \leq M_f$ for all $f'(x) \in \partial f(x)$ and some $M_f \in (0, +\infty)$.

Let $h \in \mathcal{H}_\alpha(\mathsf{X})$ be a given DGF for the feasible set $\mathsf{X} \subseteq \mathsf{V}$. Define the h-center $x^0 = \mathrm{argmin}_{x\in\mathsf{X}} h(x)$. Without loss of generality with assume $h(x^0) = 0$. Denote by

$$\Theta_h(\mathsf{X}) := \max_{p\in\mathsf{X}} h(p). \tag{3.25}$$

We emphasize that Assumption 7 implies that $\Theta_h(\mathsf{X}) < \infty$. Define the *gap function*

$$(\forall y \in \mathsf{V}^*) : g(y) = \max_{x\in\mathsf{X}} \langle y, x - x^0\rangle, \tag{3.26}$$

and

$$H_\beta(y) = \max_{x\in\mathsf{X}} \{\langle y, x - x^0\rangle - \beta h(x)\}. \tag{3.27}$$

Note that $g(y) \geq 0$ for all $y \in \mathsf{V}^*$, finite thanks to compactness of X. Also, observe that for $\beta_2 \geq \beta_1 > 0$, it holds $H_{\beta_2}(y) \leq H_{\beta_1}(y)$ for all $y \in \mathsf{V}^*$. Moreover, using the definition of the convex conjugate of the DGF $h \in \mathcal{H}_\alpha(\mathsf{X})$, one sees that

$$\begin{aligned} H_\beta(y) &= \max_{x\in\mathsf{V}} \{\langle y, x\rangle - (h + \delta_{\mathsf{X}})(x)\} - \langle y, x^0\rangle \\ &= \beta(h + \delta_{\mathsf{X}})^*(y/\beta) - \langle y, x^0\rangle \end{aligned}$$

for every $y \in \mathsf{V}^*$ and $\beta > 0$. Define the *mirror map*

$$Q_\beta(y) := \operatorname*{argmax}_{x\in\mathsf{X}} \{\langle y, x\rangle - \beta h(x)\}. \tag{3.28}$$

Then, from (3.6), one obtains the important identity

$$(\forall y \in \mathsf{V}^*) : \nabla H_\beta(y) = Q_\beta(y) - x^0.$$

In particular, in view of (3.6), the function $H_\beta(y)$ is seen to have a $\frac{1}{\alpha\beta}$-Lipschitz continuous gradient. Given the current primal-dual pair (x, y), DA performs a gradient step in the dual space V^* to produce a new gradient feedback point $y^+ = y - \lambda f'(x)$, where $\lambda > 0$ is a step size parameter. Taking this as a new signal, we update the primal state by applying the mirror map $x^+ = Q_\beta(y^+)$.

The Dual Averaging method (DA)
Input: pick $y^0 = 0, x^0 = Q_{\beta_0}(0)$, nondecreasing learning sequence $(\beta_k)_{k\in\mathbb{N}_0}$ and non-increasing step-size sequence $(\lambda_k)_{k\in\mathbb{N}_0}$
General step: For $k = 0, 1, \ldots$ do:
dual update $y^{k+1} = y^k - \lambda_k f\prime(x^k)$,
set $x^{k+1} = Q_{\beta_{k+1}}(y^{k+1})$.

We now assess the iteration complexity of DA, showing that it features the same order convergence rate $O(1/\sqrt{k})$, just as BPGM and MD.

The function $y \mapsto H_\beta(y)$ is convex and continuously differentiable with

$$H_\beta(y + w) \leq H_\beta(y) + \langle \nabla H_\beta(y), w\rangle + \frac{1}{2\alpha\beta}\|w\|_*^2 \quad \forall y, w \in \mathsf{V}^*. \tag{3.29}$$

Thanks to the monotonicity in the parameters, we get through some elementary manipulations the relation

$$\begin{aligned} H_{\beta_{k+1}}(y^{k+1}) &\leq H_{\beta_{k+1}}(y^k - \lambda_k f'(x^k)) \\ &\leq H_{\beta_k}(y^k) + \langle \nabla H_{\beta_k}(y^k), -\lambda_k f'(x^k)\rangle + \frac{\lambda_k^2}{2\alpha\beta_k}\|f'(x^k)\|_*^2 \\ &= H_{\beta_k}(y^k) - \lambda_k\langle x^k - x^0, f'(x^k)\rangle + \frac{\lambda_k^2}{2\alpha\beta_k}\|f'(x^k)\|_*^2. \end{aligned}$$

Rearranging and summing over $k = 1, 2, \ldots, N$, we get

$$\sum_{k=1}^N \lambda_k\langle f'(x^k), x^k - x^0\rangle \leq H_{\beta_1}(y^1) - H_{\beta_{N+1}}(y^{k+1}) + \sum_{k=1}^N \frac{\lambda_k^2}{2\alpha\beta_k}\|f'(x^k)\|_*^2.$$

Observe

$$\begin{aligned} H_{\beta_1}(y^1) = H_{\beta_1}(-\lambda_0 f'(x^0)) &\leq H_{\beta_1}(0) + \langle \nabla H_{\beta_1}(0), -\lambda_0 f'(x^0)\rangle + \frac{\lambda_0^2}{2\alpha\beta_1}\|f'(x^0)\|_*^2 \\ &= \frac{\lambda_0^2}{2\alpha\beta_1}\|f'(x^0)\|_*^2 \leq \frac{\lambda_0^2}{2\alpha\beta_0}\|f'(x^0)\|_*^2. \end{aligned}$$

Therefore,

$$\sum_{k=0}^N \lambda_k\langle f'(x^k), x^k - x^0\rangle \leq \frac{1}{2\alpha}\sum_{k=0}^N \frac{\lambda_k^2}{\beta_k}\|f'(x^k)\|_*^2 - H_{\beta_{N+1}}(y^{N+1}). \tag{3.30}$$

Note that

$$\begin{aligned} \sum_{k=0}^N \lambda_k\langle f'(x^k), x^k - x^0\rangle &= \sum_{k=0}^N \lambda_k\langle f'(x^k), x^k - x\rangle + \sum_{k=0}^N \lambda_k\langle f'(x^k), x - x^0\rangle \\ &= \sum_{k=0}^N \lambda_k\langle f'(x^k), x^k - x\rangle - \langle y^{N+1}, x - x^0\rangle, \end{aligned}$$

for all $x \in \mathsf{X}$. This shows

$$\sum_{k=0}^N \lambda_k\langle f'(x^k), x^k - x\rangle \leq \frac{1}{2\alpha}\sum_{k=0}^N \frac{\lambda_k^2}{\beta_k}\|f'(x^k)\|_*^2 - H_{\beta_{N+1}}(y^{N+1}) + \langle y^{N+1}, x - x^0\rangle \tag{3.31}$$

for all $x \in \mathsf{X}$. Defining $\theta_k := \max_{x\in\mathsf{X}} \sum_{i=0}^k \lambda_i\langle f'(x^i), x^i - x\rangle$, (3.31) gives the estimate

$$\theta_N \leq \frac{1}{2\alpha}\sum_{k=0}^N \frac{\lambda_k^2}{\beta_k}\|f'(x^k)\|_*^2 - H_{\beta_{N+1}}(y^{N+1}) + g(y^{N+1}).$$

A simple application of the min-max inequality shows

$$\begin{aligned} g(y) &= \max_{x\in\mathsf{X}} \langle y, x - x^0\rangle \\ &= \max_{x\in\mathsf{X}} \min_{\beta\geq 0} \{\langle y, x - x^0\rangle + \beta(\Theta_h(\mathsf{X}) - h(x))\} \\ &\leq \min_{\beta\geq 0} \left[\left(\max_{x\in\mathsf{X}} \{\langle y, x - x^0\rangle - \beta h(x)\right) + \beta\Theta_h(\mathsf{X})\right] \\ &\leq \beta\Theta_h(\mathsf{X}) + H_\beta(y) \end{aligned}$$

for all $y \in \mathsf{V}^*$ and $\beta > 0$. Hence, (3.31) gives

$$\theta_N \leq \beta_{N+1}\Theta_h(\mathsf{X}) + \frac{1}{2\alpha}\sum_{k=0}^N \frac{\lambda_i^2}{\beta_k}\|f'(x^k)\|_*^2.$$

Define $\Lambda_N := \sum_{k=0}^N \lambda_k$, and the ergodic average $\bar{x}_N := \frac{1}{\Lambda_N}\sum_{k=0}^N \lambda_k x^k$. The subgradient inequality (1.1) applied to f gives

$$\sum_{k=0}^N \lambda_k\langle f'(x^k), x^k - x\rangle \geq \sum_{k=0}^N \lambda_k f(x^k) - \Lambda_N f(x) \geq \Lambda_N(f(\bar{x}_N) - f(x)).$$

Therefore, $\theta_N \geq \Lambda_N(f(\bar{x}_N) - \Psi_{\min}(\mathsf{X}))$, and we conclude

$$\Psi(\bar{x}_N) - \Psi_{\min}(\mathsf{X}) \leq \frac{\beta_{N+1}}{\Lambda_N}\Theta_h(\mathsf{X}) + \frac{1}{2\alpha\Lambda_N}\sum_{k=0}^N \frac{\lambda_k^2}{\beta_k}\|f'(x^k)\|_*^2.$$

Let us now make the concrete choice of parameters $\beta_k = \beta > 0$ and $\lambda_k = \frac{1}{\sqrt{k+1}}$ for all $k \geq 0$. Then,

$$\Lambda_N = \sum_{k=0}^N \frac{1}{\sqrt{k+1}} \geq \int_0^{N+1} \frac{1}{\sqrt{x+1}}\,dx \geq \sqrt{N+1},$$

as well as

$$\sum_{k=0}^N \frac{\lambda_k^2}{\beta_k} = \frac{1}{\beta}\sum_{k=0}^N \frac{1}{k+1} \leq \frac{1 + \ln(N+1)}{\beta}.$$

Theorem 3.8. *Suppose Assumptions 3, 7 and Assumption 8 hold true. Let $(x^k)_k$ be generated by DA with parameters $\beta_k = \beta > 0$ and $\lambda_k = \frac{1}{\sqrt{k+1}}$. Then,*

$$\Psi(\bar{x}_N) - \Psi_{\min}(\mathsf{X}) \leq \frac{\beta\Theta_h(\mathsf{X})}{\sqrt{N+1}} + \frac{M_f^2(1 + \ln(N+1))}{2\alpha\beta\sqrt{N+1}} \tag{3.32}$$

A slightly better bound can be obtained if we decide to run DA over a fixed time window $k \in \{0, 1, \dots, N\}$. Committing over this time interval on the constant parameter sequences $\beta_k = \beta > 0$ and $\lambda_k = \lambda > 0$ yields

$$\Psi(\bar{x}_N) - \Psi_{\min}(\mathsf{X}) \leq \frac{\beta \Theta_h(\mathsf{X})}{(N+1)\lambda} + \frac{1}{2\alpha}\frac{\lambda^2}{\beta} M_f^2.$$

Optimizing with respect to $\lambda > 0$ gives the choice $\lambda = \frac{\beta}{M_f}\sqrt{\frac{2\alpha\Theta_h(\mathsf{X})}{N+1}}$, and the complexity upper bound

$$\Psi(\bar{x}_N) - \Psi_{\min}(\mathsf{X}) \leq \sqrt{\frac{2\Theta_h(\mathsf{X})}{\alpha(N+1)}} M_f.$$

The effectiveness of non-Euclidean setups With the help of the explicit rate estimate (3.32) we are now in the position to evaluate the potential efficiency gains we can make by adopting the non-Euclidean framework. We will do so by focusing on the geometry $\mathsf{X} = \{x \in \mathbb{R}^n_+ | \sum_{i=1}^n x_i = 1\}$. There are two natural projection frameworks for the unit simplex:

- Consider the ℓ_2 setup in which $\|\cdot\| = \|\cdot\|_2$ and $h(x) = \frac{1}{2}\|x\|_2^2 - \frac{1}{2n}$. Then $x^0 = (1/n, \dots, 1/n)$ and $h(x^0) = 0$. The h-diameter of X is $\Theta_h(\mathsf{X}) = \frac{n-1}{2n}$. We denote by $M_{f,\|\cdot\|_2}$ the bound on the subgradients of the function f under the ℓ_2-norm. The corresponding complexity estimate is
$$\text{Complexity}(\mathsf{X}, \|\cdot\|_2) = \sqrt{\frac{(n-1)/n}{\alpha(N+1)}} M_{f,\|\cdot\|_2}.$$
- A different sensible projection framework is obtained by consider the ℓ_1-norm $\|\cdot\| = \|\cdot\|_1$ with DGF $h(x) = \sum_{i=1}^n x_i \ln(x_i) - \ln(1/n)$. The h-diameter is $\Theta_h(\mathsf{X}) = \ln(n)$. We let $M_{f,\|\cdot\|_\infty}$ denote the bound on the subgradients of the function f under the dual norm ℓ_∞. The corresponding complexity estimate is
$$\text{Complexity}(\mathsf{X}, \|\cdot\|_1) = \sqrt{\frac{2\ln(n)}{\alpha(N+1)}} M_{f,\|\cdot\|_\infty}.$$

To compare the complexity estimates implied by the two different Bregman setups, we compute the efficiency ratio

$$R := \frac{\text{Complexity}(\mathsf{X}, \|\cdot\|_2)}{\text{Complexity}(\mathsf{X}, \|\cdot\|_1)} = \sqrt{\frac{n-1}{2n\ln(n)}} \frac{M_{f,\|\cdot\|_2}}{M_{f,\|\cdot\|_\infty}}.$$

If $R < 1$ then the ℓ_2 setup is more efficient than the ℓ_1 setup. Since $\|a\|_\infty \leq \|a\|_2 \leq \sqrt{n}\|a\|_\infty$ for all $a \in \mathbb{R}^n$, we see that $1 \leq \frac{M_{f,\|\cdot\|_2}}{M_{f,\|\cdot\|_\infty}} \leq \sqrt{n}$. This implies

$$\sqrt{\frac{n-1}{2n\ln(n)}} \leq R \leq \sqrt{\frac{n-1}{2\ln(n)}}.$$

The closer the efficiency ratio R gets to the upper bound, the more favorable the ℓ_1-setup would be compared to the standard ℓ_2-setup. This is not an unrealistic situation in practice. Ben-Tal et al. (2001) experimentally verify the significant advantages of the ℓ_1 setup for large-dimensional simplex domains.

3.4.1. On the connection between Dual Averaging and Mirror Descent

A deep and important connection between the Dual Averaging and Mirror Descent algorithms for convex non-smooth optimization has been observed in Beck and Teboulle (2003). To relate these iterates to BPGM, we assume that $h \in \mathcal{E}(\mathsf{X})$, in the sense of Definition 3.5. Let us recall that h is essentially smooth if and only if its Fenchel conjugate h^* is essentially smooth. Moreover, $\nabla h : \text{int}(\text{dom}\, h) \to \text{int}(\text{dom}\, h^*)$ is a bijection with

$$(\nabla h)^{-1} = \nabla h^* \text{ and } \nabla h^*(\nabla h(x)) = \langle x, \nabla h(x)\rangle - h(x) \tag{3.33}$$

Since $\mathsf{X} = \text{cl}(\text{dom}\, h)$, it follows

$$\text{dom}\, \partial h = \text{int}(\text{dom}\, h) = \text{int}(\mathsf{X}) \text{ with } \partial h(x) = \{\nabla h(x)\} \quad \forall x \in \text{int}(\mathsf{X}).$$

Assuming that the penalty function h is of Legendre type, the primal projection step is seen to be the regularized maximization step

$$x^k = \underset{u\in\mathsf{X}}{\text{argmax}}\{\langle y^k, u\rangle - \beta_k h(u)\} \Leftrightarrow y^k = \beta_k \nabla h(x^k).$$

Using the definition of the dual trajectory, we see that for all $k \geq 0$ the primal-dual relation obeys:

$$0 = \lambda_k \nabla f(x^k) + \beta_{k+1}\nabla h(x^{k+1}) - \beta_k \nabla h(x^k).$$

Assuming that $\beta_k \equiv 1$, this implies

$$x^{k+1} \in \underset{u\in\mathsf{X}}{\text{argmin}}\{\langle \lambda_k \nabla f(x^k), u - x^k\rangle + D_h(u, x^k)\} = \mathcal{P}_{\delta_\mathsf{X}}(x^k, \lambda_k \nabla f(x^k)).$$

We have thus shown that DA and BPGM/MD agree if all parameters and initial conditions are chosen in the same way.

3.4.2. Links to continuous-time dynamical systems

The connection between numerical algorithms and continuous-time dynamical systems for optimization is classical and well-documented in the literature (see e.g. Helmke and Moore (1996) for a textbook reference). Here we describe an interesting link between dual averaging and a class or Riemannian gradient flows originally introduced in Alvarez et al. (2004); Attouch et al. (2004); Attouch and Teboulle (2004) and further studied in Bolte and Teboulle (2003). A complexity analysis of discretized versions of these gradient flows has recently been obtained in Bomze et al. (2019). Our point of departure is the following continuous-time dynamical system based on dual averaging, which has been introduced in Mertikopoulos and Staudigl (2018a) in the context of convex programming and in Mertikopoulos and Staudigl (2018b) for general monotone variational inequality problems. The main ingredient of this dynamical system is a pair of primal-dual trajectories $(x(t), y(t))_{t\geq 0}$ evolving in continuous time according to the differential-projection system

$$\begin{cases} y'(t) := \frac{dy(t)}{dt} = -\lambda(t)\nabla f(x(t)), \\ x(t) = Q_1(\eta(t)y(t)) =: Q(\eta(t)y(t)). \end{cases} \tag{3.34}$$

In this formulation, Assumption 1(a) is in place, in order to ensure that the dynamical system is well-posed, thanks to the Picard-Lindelöf theorem. To relate this scheme formally to its discrete-time counterpart DA, let us perform an Euler discretization of the dual trajectory by $y^k - y^{k-1} = -\lambda_k \nabla f(x^k)$, and project the resulting point to the primal space by applying the mirror map $Q(\frac{1}{\beta_{k+1}} y^{k+1})$, where β_{k+1}^{-1} is the discrete-time learning rate appropriately sampled from the function $\eta(t)$. As in Section 3.4.1, let us assume that the mirror map is generated by a Legendre function $h \in \mathcal{E}(\mathsf{X})$, so that

$$x(t) = \nabla h^*(\eta(t)y(t)).$$

Let us further assume that h is twice continuously differentiable and $\eta(t) \equiv 1$. Differentiating the previous equation with respect to time t gives

$$x'(t) = \nabla^2 h^*(y(t))y'(t) = -\lambda(t)\nabla^2 h^*(y(t))\nabla f(x(t)).$$

To make headway, recall the basic properties of Legendre function saying $\nabla h^*(\nabla h(x)) = x$ for all $x \in \text{int dom}\, h$ (cf. (3.33)). Differentiating implicitly this identity, we obtain $\nabla^2 h^*(\nabla h(x)) \equiv \text{Id}$, or

$$\nabla^2 h^*(\nabla h(x)) = [\nabla^2 h(x)]^{-1} =: H(x)^{-1}. \tag{3.35}$$

As in Section 3.4.1, it holds true that $y(t) = \nabla h(x(t))$ for all $t \geq 0$, we therefore obtain the interesting characterization of the primal trajectory as

$$x'(t) = -\lambda(t)H(x(t))^{-1}\nabla f(x(t)).$$

If X is a smooth manifold, we can define a Riemannian metric

$$g_x(u, v) := \langle H(x)u, v\rangle \qquad \forall (x, u, v) \in \mathsf{X}^\circ \times \mathsf{V} \times \mathsf{V}.$$

The gradient of a smooth function ϕ with respect to the metric g is then given by $\nabla_g \phi(x) = H(x)^{-1}\nabla\phi(x)$. Hence, the continuous-time version of

the dual averaging method gives rise the class of primal *Riemannian-Hessian gradient flows*

$$x'(t) + \lambda(t)\nabla_g f(x(t)) = 0, \quad x(0) \in \mathsf{X}^\circ. \tag{3.36}$$

This class of continuous-time dynamical systems gave rise to a vigorous literature in connection with Nesterov's optimal method, which we will thoroughly discuss in Section 6. As an appetizer, consider the system of differential equations

$$y'(t) = -\lambda(t)\nabla f(x(t)), \quad x'(t) = \gamma(t)[Q(\eta(t)y(t)) - x(t)]. \tag{3.37}$$

Suppose that in (3.37) we take $Q(y) = y, \eta(t) = 1$. This corresponds to the Legendre function $h(x) = \frac{1}{2}\|x\|_2^2 + \delta_{\mathsf{X}}(x)$ for a given closed convex set X. Under this specification, the dynamical system (3.37) becomes

$$y'(t) = -\lambda(t)\nabla f(x(t)), \quad x'(t) = \gamma(t)[y(t) - x(t)].$$

Combining the primal and the dual trajectory, we easily derive a purely primal second-order in time dynamical system given by

$$x''(t) - x'(t)\left(\frac{\gamma(t)^2 - \gamma'(t)}{\gamma(t)}\right) + \lambda(t)\nabla f(x(t)) = 0.$$

Setting $\gamma(t) = \beta/t$ and $\lambda(t) = 1/\gamma(t)$ and rearranging gives

$$x''(t) + \frac{\beta+1}{t}x'(t) + \nabla f(x(t)) = 0,$$

which corresponds to the continuous-time version of the Heavy-ball method of Polyak Polyak (1964). For $\beta = 2$ this gives the continuous-time formulation of Nesterov's accelerated scheme, as shown by Su et al. (2016).

More generally, suppose that h is a twice continuously differentiable Legendre function and $\eta(t) \equiv 1$. Then a direct calculation shows that

$$x''(t) + \left(\gamma(t) - \frac{\gamma'(t)}{\gamma(t)}\right)x'(t) + \gamma(t)\lambda(t)(\nabla^2 h)^{-1}(Q(y(t)))\nabla f(x(t)) = 0.$$

Using the identity (3.35), as well as $\frac{x'(t)}{\gamma(t)} + x(t) = \nabla h^*(y(t))$, it follows that

$$\nabla^2 h\left(x(t) + \frac{x'(t)}{\gamma(t)}\right)\left(\frac{x''(t)}{\gamma(t)} + \left(1 - \frac{\gamma'(t)}{\gamma(t)^2}\right)x'(t)\right) = -\lambda(t)\nabla f(x(t)) \Leftrightarrow \frac{d}{dt}\nabla h\left(x(t) + \frac{x'(t)}{\gamma(t)}\right) = -\lambda(t)\nabla f(x(t)).$$

This shows that for $\eta \equiv 1$, the dynamic coincides with the Lagrangian family of second-order systems constructed in Wibisono et al. (2016). These ideas are now investigated heavily when combined with numerical discretization schemes for dynamical system with the hope to get insights how to construct new and more efficient algorithmic formulation of gradient-methods. This literature grew quite fastly over the last years, and we mention (Attouch et al., 2020; 2018; Bah et al., 2019; Shi et al., 2019).

4. The Proximal Method of Multipliers and ADMM

In this section we turn our attention to a classical method for solving linearly constrained optimization problems building on the classical idea of the celebrated *method of multipliers*. An extremely powerful proponent of this class of algorithms is the *Alternating Direction Method of Multipliers* (ADMM), which has received enormous interest from different directions, including PDEs (Attouch et al., 2011; 2007), mixed-integer programming (Feizollahi et al., 2017), optimal control (Lin et al., 2012) and signal processing (Yang and Zhang, 2011; Yuan, 2012). The very influential monograph (Boyd et al., 2011) contains over 180 references, reflecting the deep impact of alternating methods on optimization theory and its applications. Following the general spirit of this survey, we introduce alternating direction methods in a proximal framework, as pioneered by Rockafellar Rockafellar (1976a,b), and due to Shefi and Teboulle (2014). See also Banert et al. (2021) for some further important elaborations.

To set the stage, consider the composite convex optimization problem (P), in its special form (2.7):

$$\Psi(x) = g(\mathbf{A}x) + r(x),$$

for a given bounded linear operator $\mathbf{A}$. To streamline the presentation, we directly assume in this section that $\mathsf{V} = \mathsf{V}^* = \mathbb{R}^n$, and the underlying metric structure is generated by the Euclidean norm $\|a\| \equiv \|a\|_2 = \langle a, a\rangle^{1/2} = \left(\sum_{i=1}^n a_i\right)^{1/2}$. Introducing the auxiliary variable $z = \mathbf{A}x$, this problem can be equivalently written as

$$\inf\{\Phi(x,z) = g(z) + r(x) | \mathbf{A}x - z = 0, x \in \mathsf{X}, z \in \mathsf{Z}\}, \tag{4.1}$$

where $\mathsf{X} = \mathbb{R}^n$ and $\mathsf{Z} = \mathbb{R}^m$. We will call this the *primal* problem. The Lagrangian associated to (4.1) is

$$L(x,z,y) = g(z) + r(x) + \langle y, \mathbf{A}x - z\rangle,$$

where $y \in \mathbb{R}^m$ is the Lagrange multiplier associated with the linear constraint. The dual function is accordingly defined as

$$\begin{aligned} q(y) &= \inf_{x,z} L(x,z,y) = \inf_z\{g(z) - \langle y, z\rangle\} + \inf_x\{r(x) + \langle \mathbf{A}x, y\rangle\} \\ &= -\sup_z\{\langle y, z\rangle - g(z)\} - \sup_x\{\langle -\mathbf{A}x, y\rangle - r(x)\} \\ &= -g^*(y) - r^*(-\mathbf{A}^\top y). \end{aligned}$$

Hence, we can represent the dual problem as the minimization problem

$$\min_y\{\tilde{\Psi}(y) = g^*(y) + r^*(-\mathbf{A}^\top y)\} \tag{4.2}$$

A classical implicit method for solving this minimization problem is the *proximal point method*:

$$y^{k+1} \in \operatorname*{argmin}_y\{\tilde{\Psi}(y) + \frac{1}{2c}\|y - y^k\|^2\} \tag{4.3}$$

where $c > 0$ is a regularization parameter controlling the effects of the quadratic penalty term. By Fermat's optimality condition, the point y^{k+1} satisfied the monotone inclusion

$$0 \in \partial g^*(y^{k+1}) - \mathbf{A}\partial r^*(-\mathbf{A}^\top y^{k+1}) + \frac{1}{c}(y^{k+1} - y^k).$$

This means that there exists $x^{k+1} \in \partial r^*(-\mathbf{A}^\top y^{k+1})$ and $z^{k+1} \in \partial g^*(y^{k+1})$ such that

$$0 = z^{k+1} - \mathbf{A}x^{k+1} + \frac{1}{c}(y^{k+1} - y^k).$$

This means that the proximal point method can be implemented for the given instance as the implicit method

$$\begin{aligned} x^{k+1} &\in \operatorname*{argmin}_x\{r(x) + \langle \mathbf{A}x, y^{k+1}\rangle\}, \\ z^{k+1} &\in \operatorname*{argmin}_z\{g(z) + \langle -z, y^{k+1}\rangle\} \\ y^{k+1} &= y^k + c(\mathbf{A}x^{k+1} - z^{k+1}). \end{aligned}$$

This defines a fully implicit iteration, which requires to compute the iterates $x^{k+1}, z^{k+1}, y^{k+1}$ simultaneously. Of course, this does not give rise to a practical algorithm. The main idea behind alternating methods is to organize the computations in a Gauss-Seidel kind of iterations in which the sequences are updated sequentially using the most recent information available. To set the stage, observe that (x^{k+1}, z^{k+1}) defined above is the coordinate-wise minimum of the function

$$F(x,z,y^k) = g(z) + r(x) + \frac{c}{2}\|\mathbf{A}x - z + \frac{1}{c}y^k\|^2.$$

In that sense, the proximal point method applied to the dual can be represented more compactly as

$$(x^{k+1}, z^{k+1}) \in \operatorname*{argmin}_{(x,z)} F(x,z,y^k), \; y^{k+1} = y^k + c(\mathbf{A}x^{k+1} - z^{k+1}).$$

This scheme is known as the *augmented Lagrangian method.*

Observe that minimizers of the function $F(x,z,y^k)$ with respect to (x,z) agree with the minimizers of the function

$$L_c(x,z,y^k) = g(z) + r(x) + \frac{c}{2}\|\mathbf{A}x - z + \frac{1}{c}y^k\|^2,$$

which is known as the *augmented Lagrangian* of problem (4.1). Using the augmented Lagrangian, an alternating minimization procedure building on the proximal point idea gives rise to the celebrated *Alternating Direction of Method of Multipliers* (ADMM).

The Alternating Direction of Method of Multipliers (ADMM)
Input: pick $(z^0, y^0) \in \mathsf{Z} \times \mathbb{R}^m$ and penalty parameter $c > 0$;
General step: For $k = 0, 1, \ldots$ do:

$$x^{k+1} = \operatorname{argmin}_{x\in\mathsf{X}}\{r(x) + \frac{c}{2}\|\mathbf{A}x - z^k + \frac{1}{c}y^k\|_2^2\} \quad (4.4)$$

$$z^{k+1} = \operatorname{argmin}_{z\in\mathsf{Z}}\{g(z) + \frac{c}{2}\|\mathbf{A}x^{k+1} - z + \frac{1}{c}y^k\|_2^2\} \quad (4.5)$$

$$y^{k+1} = y^k + c(\mathbf{A}x^{k+1} - z^{k+1}). \quad (4.6)$$

Remark 4.1. ADMM updates the decision variables in a sequential manner, and thus is not capable of featuring parallel updates unless the matrix $\mathbf{A}$ is of special structure. In the context of the AC optimal power flow problem in electric power grid optimization (Sun et al., 2013) provide such a modification of ADMM. Furthermore, the ADMM can be extended to consider formulations with general linear constraints of the form $\mathbf{A}_1 x + \mathbf{A}_2 z = b$. For ease of exposition we stick to the simplified problem formulation above.

4.1. The Douglas-Rachford algorithm and ADMM

The Douglas-Rachford (DR) algorithm is a fundamental method to solve general monotone inclusion problems where the task is to find zeros of the sum of two maximally monotone operators (see Bauschke and Combettes (2016) and Auslender and Teboulle (2006a)). To keep the focus on convex programming, we introduce this method for solving the dual problem (4.2). To that end, let us define the matrix $\mathbf{K} = -\mathbf{A}^\top$, so that our aim is to solve the convex programming problem

$$\min_z g^*(z) + r^*(\mathbf{K}z). \quad (4.7)$$

Any solution $\bar{z} \in \operatorname{dom}(r^*)$ satisfies the monotone inclusion

$$0 \in \mathbf{K}^\top \partial r^*(\mathbf{K}\bar{z}) + \partial g^*(\bar{z}). \quad (4.8)$$

The DR algorithm aims to determine such a point $\bar{z}$ by iteratively constructing a sequence $\{(u^k, v^k, y^k), k \geq 0\}$ determined by

$$v^{k+1} = (\mathrm{Id} + c\mathbf{K}^\top \circ \partial r^* \circ \mathbf{K})^{-1}(2y^k - u^k),$$
$$u^{k+1} = v^{k+1} + u^k - y^k,$$
$$y^{k+1} = (\mathrm{Id} + c\partial g^*)^{-1}(u^{k+1}).$$

To bring this into an equivalent form, let us focus on the definition of the y^{k+1} update, which reads as the inclusion

$$0 \in \frac{1}{c}(y^{k+1} - u^{k+1}) + \partial g^*(y^{k+1}).$$

This is clearly recognizable as the first-order optimality condition of the $\min_y\{g^*(y) + \frac{1}{2c}\|y - u^{k+1}\|_2^2\}$. Therefore, we can rewrite the above iteration in terms of convex optimization subroutines as:

$$v^{k+1} = \operatorname*{argmin}_v\{r^*(\mathbf{K}v) + \frac{1}{2c}\|v - (2y^k - u^k)\|_2^2\}, \quad (4.9)$$

$$u^{k+1} = v^{k+1} + u^k - w^k, \quad (4.10)$$

$$y^{k+1} = \operatorname*{argmin}_y\{g^*(y) + \frac{1}{2c}\|y - u^{k+1}\|_2^2\}. \quad (4.11)$$

Via Fenchel-Rockafellar duality, the dual problem to (4.9) reads as

$$x^{k+1} = \operatorname*{argmin}_x\{r(x) + \frac{c}{2}\|\mathbf{A}x + \frac{1}{c}(2y^k - u^k)\|_2^2\},$$

where the coupling between the primal and the dual variables is

$$u^{k+1} = y^k + c\mathbf{A}x^{k+1}.$$

The dual to step (4.11) reads as

$$z^{k+1} = \operatorname*{argmin}_z\{g(z) + \frac{c}{2}\|z - \frac{1}{c}u^{k+1}\|_2^2\}.$$

The coupling between primal and dual variables reads as

$$y^{k+1} = u^{k+1} - cz^{k+1}.$$

Combining all these relations, we can write the dual minimization problem as

$$x^{k+1} = \operatorname*{argmin}_x\{r(x) + \frac{c}{2}\|\mathbf{A}x - z^k + \frac{1}{c}y^k\|_2^2\},$$
$$z^{k+1} = \operatorname*{argmin}_z\{g(z) + \frac{c}{2}\|\mathbf{A}x^{k+1} - z + \frac{1}{c}y^k\|_2^2\},$$
$$y^{k+1} = y^k + c(\mathbf{A}x^{k+1} - z^{k+1})$$

which is just the standard ADMM. By this we have recovered a classical result on connection between the DR and ADMM algorithms due to Gabay (1983) and Eckstein and Bertsekas (1992).

4.2. Proximal Variant of ADMM

One of the limitations of the ADMM comes from the presence of the term $\mathbf{A}x$ in the update of x^{k+1}. The presence of this factor makes it impossible to implement the algorithm in parallel, which makes it slightly unattractive for large-scale problems in distributed optimization. Moreover, due to the result of Chen et al. (2016) the convergence of ADMM for general linear constraints does not generalize to more than two blocks. Leaving parallelization issues aside, Shefi and Teboulle Shefi and Teboulle (2014) proposed an interesting extension of the ADMM by adding further quadratic penalty terms, which allows much flexibility by suitably choosing the norms employed in the algorithm. Given some point $(x^k, z^k, y^k) \in \mathsf{X} \times \mathsf{Z} \times \mathbb{R}^m$ and two positive definite matrices $\mathbf{M}_1, \mathbf{M}_2$, we define the *proximal augmented Lagrangian* of (4.1) as

$$P_k(x, z, y) = L_c(x, z, y) + \frac{1}{2}\|x - x^k\|_{\mathbf{M}_1}^2 + \frac{1}{2}\|z - z^k\|_{\mathbf{M}_2}^2. \quad (4.12)$$

Here, $\|u\|_{\mathbf{M}}^2 = \langle u, \mathbf{M}u\rangle$ is the semi-norm induced by $\mathbf{M}$, which is a norm if $\mathbf{M}$ is positive definite.

The Alternating Direction proximal Method of Multipliers (AD-PMM)
Input: pick $(x^0, z^0, y^0) \in \mathsf{X} \times \mathsf{Z} \times \mathbb{R}^m$ and penalty parameter $c > 0$;
General step: For $k = 0, 1, \ldots$ do:

$$x^{k+1} = \operatorname{argmin}_{x\in\mathsf{X}}\{r(x) + \frac{c}{2}\|\mathbf{A}x - z^k + \frac{1}{c}y^k\|_2^2 + \frac{1}{2}\|x - x^k\|_{\mathbf{M}_1}^2\} \quad (4.13)$$

$$z^{k+1} = \operatorname{argmin}_{z\in\mathsf{Z}}\{g(z) + \frac{c}{2}\|\mathbf{A}x^{k+1} - z + \frac{1}{c}y^k\|_2^2 + \frac{1}{2}\|z - z^k\|_{\mathbf{M}_2}^2\} \quad (4.14)$$

$$y^{k+1} = y^k + c(\mathbf{A}x^{k+1} - z^{k+1}) \quad (4.15).$$

AD-PMM allows for various choices of the matrices $\mathbf{M}_1, \mathbf{M}_2$.

- With $\mathbf{M}_1 = \mathbf{M}_2 = 0$, we recover the classical ADMM. For any $c > 0$, it is known (Gabay, 1983; Glowinski and Tallec, 1989) that convergence in function values as well as global convergence to to dual multiplier are warranted. To ensure convergence of the primal sequence $(x^k)_k$, one needs to assume that $\mathbf{A}$ has full column rank.
- With the choice $\mathbf{M}_1 = \mu_1\mathbf{I}_n, \mathbf{M}_2 = \mu_2\mathbf{I}_m$ with $\mu_1, \mu_2 > 0$, the AD-PMM of (Eckstein, 1994) is recovered.

We give a brief analysis of the complexity of AD-PMM in the special case of problem (4.1). Recall that a standing hypothesis in this survey is that the smooth part f of the composite convex programming problem (P) admits a Lipschitz continuous gradient. Since $f(x) = g(\mathbf{A}x)$, the Lipschitz constant of ∇f is determined by a corresponding Lipschitz assumption on ∇g, with the constant henceforth denoted as L_g, and a bound on spectrum of the matrix $\mathbf{A}$. To highlight the primal-dual nature of the algorithm, a key element in the complexity analysis is the bifunction

$$S(x, y) = r(x) - g^*(y) + \langle y, \mathbf{A}x\rangle = L(x, 0, y).$$

Our derivation of an iteration complexity estimate of AD-PMM proceeds in two steps. First, we present an interesting "Meta-Theorem", due to Shefi and Teboulle (2014), and stated here as Proposition 4.2. It formulates general convergence guarantees for any primal-dual algorithms satisfying a specific per-iteration bound. We then apply this general result to AD-PMM, by verifying that this scheme actually satisfies these mentioned per-iteration bounds.

We start with an auxiliary technical fact.

Lemma 4.1. *Let $h : \mathbb{R}^n \to \mathbb{R}$ be a proper convex and L_h-Lipschitz continuous. Then, for any $\xi \in \mathbb{R}^n$ we have*

$$h(\xi) \le \max\{\langle \xi, u\rangle - h^*(u) : \|u\|_2 \le L_h\}. \tag{4.16}$$

Proof. Since h is convex and continuous, it agrees with its biconjugate: $h^{**} = h$. By Corollary 13.3.3 in Rockafellar (1970), $\operatorname{dom} h^*$ is bounded with $\operatorname{dom} h^* \subseteq \{u : \|u\|_2 \le L_h\}$. Hence, the definition of the conjugate gives

$$h(\xi) = \sup_{u \in \operatorname{dom} h^*} \{\langle u, \xi\rangle - h^*(u)\} \le \max_{u : \|u\|_2 \le L_h} \{\langle \xi, u\rangle - h^*(u)\}.$$

□

Proposition 4.2. *Let (x^*, y^*, z^*) be a saddle point for L. Let $\{(x^k, y^k, z^k); k \ge 0\}$ be a sequence generated by some algorithm for which the following estimate holds for any $y \in \mathbb{R}^m$:*

$$L(x^k, z^k, y) - \Psi(x^*) \le \frac{1}{2k}\left[C(x^*, z^*) + \frac{1}{c}\|y - y^0\|_2^2\right] \tag{4.17}$$

for some constant $C(x^*, z^*) > 0$. Then

$$\Psi(x^k) - \Psi(x^*) \le \frac{C_1(x^*, z^*, L_g)}{2k}.$$

where $C_1(x^*, z^*, L_g) = C(x^*, z^*) + \frac{2}{c}(L_g^2 + \|y^0\|_2^2)$.

Proof. Thanks to the Fenchel inequality

$$L(x, z, y) - S(x, y) = g(z) + g^*(y) - \langle y, z\rangle \ge 0.$$

By the definition of the convex conjugate

$$\Psi(x) = g(\mathbf{A}x) + r(x) = \sup_y\{r(x) + \langle y, \mathbf{A}x\rangle - g^*(y)\} = \sup_y S(x, y).$$

Now, since g is convex and continuous on $\mathbb{R}^m$, we know $g = g^{**}$, and we can apply Lemma 4.1 to obtain the string of inequalities:

$$\Psi(x^k) - \Psi(x^*) = \sup_y\{S(x^k, y) - \Psi(x^*)\} \le \sup_{y : \|y\|_2 \le L_g}\{S(x^k, y) - \Psi(x^*)\} \le \sup_{y : \|y\|_2 \le L_g}\{L(x^k, z^k, y) - \Psi(x^*)\}$$
$$\le \sup_{y : \|y\|_2 \le L_g}\left\{\frac{1}{2k}\left(C(x^*, z^*) + \frac{1}{c}\|y - y^0\|_2^2\right)\right\} \le \frac{1}{2k}\left[C(x^*, z^*) + \frac{2}{c}(L_g + \|y^0\|_2^2)\right].$$

□

To apply this Meta-Theorem, we need to verify that AD-PMM satisfies the condition (4.17). To make progress towards that end, Lemma 4.2 in Shefi and Teboulle (2014) proves that

$$L(x^{k+1}, z^{k+1}, y) - L(x, z, y^{k+1}) \le T_k(x, z, x^{k+1}) + R_k(x, y, z) \tag{4.18}$$

for all $(x, z, y) \in \mathsf{X} \times \mathsf{Z} \times \mathbb{R}^m$ and some explicitly given functions T_k and R_k. Furthermore, it is shown that

$$T_k(x, z, x^{k+1}) \le \frac{c}{2}\left(\|\mathbf{A}x - z^k\|_2^2 - \|\mathbf{A}x - z^{k+1}\|_2^2 + \frac{c}{2}\|\mathbf{A}x^{k+1} - z^{k+1}\|_2^2\right), \text{ and}$$
$$R_k(x, z, y) \le \frac{1}{2}\left(\Delta_k(x, \mathbf{M}_1) + \Delta_k(z, \mathbf{M}_2) + \frac{1}{c}\Delta_k(y, \mathrm{Id})\right) - \frac{c}{2}\|\mathbf{A}x^{k+1} - z^{k+1}\|_2^2,$$

where for any point z and positive semi-definite matrix $\mathbf{M}$,

$$\Delta_k(z, \mathbf{M}) = \frac{1}{2}\|z - z^k\|_{\mathbf{M}}^2 - \frac{1}{2}\|z - z^{k+1}\|_{\mathbf{M}}^2.$$

Using these bounds and summing inequality (4.18) over $k = 0, 1, \dots, N-1$, we get

$$\sum_{k=0}^{N-1}[L(x^{k+1}, z^{k+1}, y) - L(x, z, y^{k+1})] \le \frac{1}{2}\left(c\|\mathbf{A}x - z^0\|_2^2 + \|x - x^0\|_{\mathbf{M}_1}^2 + \|z - z^0\|_{\mathbf{M}_2}^2 + \frac{1}{c}\|y - y^0\|_2^2\right)$$

Dividing both sides by N and using the convexity of the Lagrangian with respect to (x, z) and the linearity in y, we easily get

$$L(\bar{x}_N, \bar{z}_N, y) - L(x, z, \bar{y}_N) \le \frac{1}{2N}\left(C(x, z) + \frac{1}{c}\|y - y^0\|_2^2\right)$$

in terms of the ergodic average

$$\bar{x}_N = \frac{1}{N}\sum_{k=0}^{N-1} x^k,\ \bar{y}_N = \frac{1}{N}\sum_{k=0}^{N-1} y^k,\ \bar{z}_N = \frac{1}{N}\sum_{k=0}^{N-1} z^k,$$

and the constant $C(x, z) = c\|\mathbf{A}x - z^0\|^2 + \|x - x^0\|_{\mathbf{M}_1}^2 + \|z - z^0\|_{\mathbf{M}_2}^2$. Therefore, we can apply Proposition 4.2 to the sequence of ergodic averages $(\bar{x}_k, \bar{z}_k, \bar{y}_k)$ generated by AD-PMM, and derive a $O(1/N)$ convergence rate in terms of the function value.

4.3. Relation to the Chambolle-Pock primal-dual splitting

In this subsection we discuss the relation between ADMM and the celebrated Chambolle-Pock (a.k.a Primal-Dual Hybrid Gradient) method (Chambolle and Pock, 2011), designed for problems in the form (2.8).

The Chambolle-Pock primal-dual algorithm (CP)
Input: pick $(x^0, y^0, p^0) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^m$ and $c, \tau > 0, \theta \in [0, 1]$;
General step: For $k = 0, 1, \dots$ do:

$$x^{k+1} = \operatorname{argmin}_x\{r(x) + \frac{1}{2\tau}\|x - (x^k - \tau\mathbf{A}^\top p^k)\|_2^2 \tag{4.19}$$
$$y^{k+1} = \operatorname{argmin}_y\{g^*(y) + \frac{1}{2c}\|y - (y^k + c\mathbf{A}x^{k+1})\|_2^2\} \tag{4.20}$$
$$p^{k+1} = y^{k+1} + \theta(y^{k+1} - y^k) \tag{4.21}.$$

For later references it is instructive to write this algorithm slightly differently in operator-theoretic notation. From the optimality condition of the step x^{k+1}, we see

$$0 \in \partial r(x^{k+1}) + \frac{1}{\tau}(x^{k+1} - w^k) \Leftrightarrow 0 \in (\mathrm{Id} + \tau\partial r)(x^{k+1}) - w^k$$

where $w^k = x^k - \tau\mathbf{A}^\top p^k$. Hence, we can give an explicit expression of the update as

$$x^{k+1} = (\mathrm{Id} + \tau\partial r)^{-1}(w^k) = (\mathrm{Id} + \tau\partial r)^{-1}(x^k - \tau\mathbf{A}^\top p^k).$$

Similarly, we can write the update y^{k+1} explicitly as

$$y^{k+1} = (\mathrm{Id} + c\partial g^*)^{-1}(y^k + c\mathbf{A}x^{k+1}).$$

When $\theta = 0$ we obtain the classical Arrow-Hurwicz primal-dual algorithm (Arrow et al., 1958). For $\theta = 1$ the last line in CP becomes $p^{k+1} = 2y^{k+1} - y^k$, which corresponds to a simple linear extrapolation based on the current and previous iterates. In this case, Chambolle and Pock (2011) provide a $O(1/N)$ non-asymptotic convergence guarantees in terms of the primal-dual gap function of the corresponding saddle-point problem. The CP primal-dual splitting method has been of immense importance in imaging and signal processing and constitutes nowadays a standard method for tackling large-scale instances in these application domains. Interestingly, if $\theta = 1$, CP is a special case of the proximal version of ADMM (AD-PMM). To establish this connection, let us set $\mathbf{M}_1 = \frac{1}{c}\mathrm{Id} - c\mathbf{A}^\top\mathbf{A}$ and $\mathbf{M}_2 = 0$. After some elementary manipulations, we arrive at the update formula for x^{k+1} in AD-PMM (4.13) as

$$x^{k+1} = \operatorname*{argmin}_x\{r(x) + \frac{1}{2\tau}\|x - (x^k - \tau\mathbf{A}^\top(y^k + c(\mathbf{A}x^k - z^k)))\|_2^2\}.$$

Introducing the variable $p^k = y^k + c(\mathbf{A}x^k - z^k)$, the above reads equivalently as

$$x^{k+1} = \operatorname*{argmin}_x\{r(x) + \frac{1}{2\tau}\|x - (x^k - \tau\mathbf{A}^\top p^k)\|_2^2\} = \mathrm{Prox}_{\tau r}(x^k - \tau\mathbf{A}^\top p^k).$$

For $\mathbf{M}_2 = 0$, the second update step in AD-PMM (4.14) reads as

$$z^{k+1} = (\mathrm{Id} + \frac{1}{c}\partial g)^{-1}\left(\mathbf{A}x^{k+1} + \frac{1}{c}y^k\right) = \mathrm{Prox}_{\frac{1}{c}g}\left(\frac{1}{c}(c\mathbf{A}x^{k+1} + y^k)\right).$$

Moreau's identity Bauschke and Combettes (2016, Proposition 23.18) states that

$$c\,\mathrm{Prox}_{\frac{1}{c}g}(u/c) + \mathrm{Prox}_{cg^*}(u) = u \quad \forall u \in \mathsf{V}. \tag{4.22}$$

Applying this fundamental identity, we see

$$cz^{k+1} + \mathrm{Prox}_{cg^*}(y^k + c\mathbf{A}x^{k+1}) = y^k + c\mathbf{A}x^{k+1}.$$

The second summand is just the y^{k+1}-update in the CP algorithm, so that we deduce

$$cz^{k+1} + y^{k+1} = y^k + c\mathbf{A}x^{k+1} \Leftrightarrow y^{k+1} = y^k + c(\mathbf{A}x^{k+1} - z^{k+1}).$$

Consequently,

$$p^{k+1} = y^{k+1} + c(\mathbf{A}x^{k+1} - z^{k+1}) = 2y^{k+1} - y^k,$$

and hence we recover the three-step iteration defining CP:

$$x^{k+1} = \operatorname*{argmin}_x\{r(x) + \frac{1}{2\tau}\|x - (x^k - \tau\mathbf{A}^\top p^k)\|_2^2\}$$

$$y^{k+1} = \operatorname*{argmin}_y\{g^*(y) + \frac{1}{2c}\|y - (y^k + c\mathbf{A}x^{k+1})\|_2^2\}$$

$$p^{k+1} = 2y^{k+1} - y^k.$$

Given the above derivations, we can summarize this subsection by the following interesting observation.

Proposition 4.3 (Proposition 3.1, Shefi and Teboulle (2014))**.** *Let (x^k, y^k, p^k) be a sequence generated by CP with $\theta = 1$. Then, the y^{k+1}-update (4.20) is equivalent to*

$$z^{k+1} = \operatorname*{argmin}_z\{g(z) + \frac{c}{2}\|\mathbf{A}x^{k+1} - z + \frac{1}{c}y^k\|_2^2\},$$

$$y^{k+1} = y^k + c(\mathbf{A}x^{k+1} - z^{k+1})$$

which corresponds to the primal z^{k+1}-minimization step (4.14) with $\mathbf{M}_2 = 0$, and to the dual multiplier update for y^{k+1} (4.15) of AD-PMM, respectively. Moreover, the minimization step with respect to x in the CP algorithm given in (4.19) together with (4.15) reduces to (4.13) of AD-PMM with $\mathbf{M}_1 = \tau\,\mathrm{Id} - c\mathbf{A}^\top\mathbf{A}$.

5. The Conditional Gradient Method

The efficiency of the Bregman proximal gradient method stands and falls with the relative ease of evaluating the Bregman proximal operator (3.7). In this section, we present a class of first-order methods which gain relevance in large-scale problems for which the computation of the projection-like operators is a significant computational bottleneck. We describe *conditional gradient* (CG) methods, a family of methods which, originating in the 1960's, have received much attention in both machine learning and optimization in the last 20 years. CG is designed to solve convex programming problems over compact convex sets. Therefore, we assume in this section that the feasible set X is a compact convex set.

Assumption 9. The set X is a compact convex subset in a finite-dimensional real vector space V.

5.1. Classical Conditional gradient

CG, also known as the *Frank-Wolfe method*, was independently developed by Frank and Wolfe Frank and Wolfe (1956) for linearly constrained quadratic problems, and by Levitin and Polyak Levitin and Polyak (1966) for general smooth convex optimization problems over compact domains:

$$\Psi_{\min}(\mathsf{X}) := \min\{f(x) | x \in \mathsf{X}\}. \tag{5.1}$$

CG attempts to solve problem (5.1) by sequentially calling a *linear oracle* (LO), a fundamental notion we introduce next.

Definition 5.1. The Operator $\mathcal{L}_\mathsf{X} : \mathsf{V}^* \to \mathsf{X}$ is a *linear oracle* (LO) over set X if for any vector $y \in \mathsf{V}^*$ we have that

$$\mathcal{L}_\mathsf{X}(y) \in \operatorname*{argmin}_{s\in\mathsf{X}} \langle y, s\rangle. \tag{5.2}$$

The practical application of an LO requires to make a selection from the set of solutions of the defining linear minimization problem. The precise definition of such a selection mechanism is not of any importance, and thus we are just concerned with any answer $\mathcal{L}_\mathsf{X}(y)$ revealed by the oracle.

The information-theoretic assumption that the optimizer can only query a linear minimization oracle is clearly the main difference between CG and other gradient-based methods discussed in Section 3. For instance, the dual averaging algorithm solves at each iteration a strongly convex subproblem of the form

$$\min_{u\in\mathsf{X}}\{\langle y, u\rangle + h(u)\}, \tag{5.3}$$

where $h \in \mathcal{H}_\alpha(\mathsf{X})$, whereas CG solves a single linear minimization problem at each iteration. This difference in the updating mechanism yields the following potential advantages of the CG method.

1. Low iteration costs: In many cases it is much easier to construct an LO rather than solving the non-linear subproblem (5.3). We emphasize that this potential benefit of CG does not depend on the structure of the objective function f, but rather on the geometry of the feasible set X. To illustrate this point, consider the spectrahedron $\mathsf{X} = \{\mathbf{X} \in \mathbb{R}^{n\times n}_{\mathrm{sym}} | \mathbf{X} \succeq 0, \mathrm{tr}(\mathbf{X}) \le 1\}$. Computing the orthogonal projection of some symmetric matrix $\mathbf{Y}$ onto the spectrahedron requires first to compute the full spectral decomposition $\mathbf{Y} = \mathbf{U}\mathbf{D}\mathbf{U}^\top$, and then for the diagonal matrix $\mathbf{D}$ computing the projection of its diagonal elements onto the simplex. The resulting projection is therefore given by
$$P_\mathsf{X}(\mathbf{Y}) = \mathbf{U}\,\mathrm{Diag}(P_{\Delta_n}(\mathrm{diag}(\mathbf{D})))\mathbf{U}^\top.$$
In contrast, computing a linear oracle over X for the symmetric matrix $\mathbf{Y}$ involves finding the eigenvector of $\mathbf{Y}$ corresponding to the minimal eigenvalue, that is $\mathcal{L}_\mathsf{X}(\mathbf{Y}) = uu^\top$, where $u^\top\mathbf{Y}u = \lambda_{\min}(\mathbf{Y})$. This operation can be typically done using numerical linear algebra techniques such as Power, Lanczos or Kaczmarz, and randomized versions thereof (see Kuczyński and Woźniakowski (1992) for general complexity results). For large-scale problems, computing such a leading eigenvector to a predefined accuracy is much more efficient than a full spectral decomposition.
2. Simplicity: The definition of an LO does not rely on a specific DGF $h \in \mathcal{H}_\alpha(\mathsf{X})$ and makes the update affine invariant.
3. Structural properties of the updates: When the feasible set X can be represented as the convex hull of a countable set of atoms ("generators"), then CG often leads to simple updates, activating only few atoms at each iteration. In particular, in the case of the spectrahedron, the LO returns a matrix of rank one, which allows for sparsity preserving iterates.

The classical form of CG takes the answer obtained from querying the LO at a given gradient feedback $y = \nabla f(x)$, and returns the target vector

$$p(x) = \mathcal{L}_\mathsf{X}(\nabla f(x)) \qquad \forall x \in \mathsf{X}. \tag{5.4}$$

It proposes then to move in the direction $p(x) - x$. As in every optimization routine, a key question is how to design efficient step-size rules to guarantee reasonable numerical performance. Letting x^{k-1} and $p^k = p(x^{k-1})$ be a current position of the method together with its implied target vector, the following policies are standard choices:

$$\text{Standard:} \quad \gamma_k = \frac{1}{2+k}, \tag{5.5}$$

$$\text{Exact line search:} \quad \gamma_k \in \operatorname*{argmin}_{t\in(0,1]} f(x^{k-1} + t(p^k - x^{k-1})), \tag{5.6}$$

$$\text{Adaptive:} \quad \gamma_k = \min\left\{\frac{\langle \nabla f(x^{k-1}), x^{k-1} - p^k\rangle}{L_f\|x^{k-1} - p^k\|^2}, 1\right\}. \tag{5.7}$$

Exact line search is conceptually attractive, but can be costly in large-scale applications when computing the function value is computationally expensive. To understand the construction of the adaptive step-size scheme, it is instructive to introduce a primal gap (merit) function to the problem defined as

$$\mathsf{e}(x) := \sup_{u\in\mathsf{X}} \langle \nabla f(x), x-u\rangle. \tag{5.8}$$

This merit function is just the gap program (see e.g. Facchinei and Pang (2003)) associated to the monotone variational inequality (2.6) in which the non-smooth part is trivial. In terms of this merit function, the descent lemma (3.15) yields immediately

$$\begin{aligned} f(x+t(p(x)-x)) &\le f(x)+t\langle\nabla f(x), p(x)-x\rangle + \frac{L_f t^2}{2}\|p(x)-x\|^2 \\ &= f(x) - t\mathsf{e}(x) + \frac{L_f t^2}{2}\|p(x)-x\|^2 = f(x)-\eta_x(t), \end{aligned}$$

where $\eta_x(t) := t\mathsf{e}(x) - \frac{L_f t^2}{2}\|p(x)-x\|^2$. Optimizing this function with respect to $t\in[0,1]$ yields the largest-possible per-iteration decrease and returns the adaptive step-size rule in (5.7). Once the optimizer decided upon the specific step-size policy, the classical CG picks one of the step sizes (5.5), (5.6), or (5.7), and performs the update

$$x^k = x^{k-1} + \gamma_k(p(x^k) - x^{k-1}).$$

The classical conditional gradient (CG)
Input: A linear oracle $\mathcal{L}_{\mathsf{X}}$, a starting point $x^0\in\mathsf{X}$.
Output: A solution x such that $\Psi(x)-\Psi_{\min}(\mathsf{X})<\varepsilon$.
General step: For $k=1,2,\dots$
Compute $p^k = \mathcal{L}_X(\nabla f(x^{k-1}))$;
Choose a step-size γ_k either by (5.5), (5.6), (5.7);
Update $x^k = x^{k-1}+\gamma_k(p^k-x^{k-1})$;
Compute $\mathsf{e}^k = \mathsf{e}(x^{k-1})$.
If $\mathsf{e}^k<\varepsilon$ return x^k.

The convergence properties of classical CG under either of the step-size variants above is well documented in the literature (see e.g. the recent text by Lan (2020), or Jaggi (2013)). We will obtain a full convergence and complexity theory under our more general analysis of the generalized CG scheme.

5.1.1. Relative smoothness

The basic ingredient in proving convergence and complexity results on the classical CG is the fundamental inequality

$$f(x+t(p(x)-x)) \le f(x) - t\mathsf{e}(x) + \frac{L_f t^2}{2}\|p(x)-x\|^2.$$

Based on the relative smoothness analysis in Section 3.3.3, it seems to be intuitively clear that we could easily prove also convergence of CG when instead of the restrictive Lipschitz gradient assumption we make a relative smoothness assumption in terms of the pair (f,h) for some DGF $h\in\mathcal{H}_\alpha(\mathsf{X})$. Indeed, if we are able to estimate a scalar $L_f^h>0$ such that $L_f^h h(x)-f(x)$ is convex on X, then the modified Descent Lemma (3.19) yields the overestimation

$$f(x+t(p-x)) \le f(x) - t\mathsf{e}(x) + L_f^h D_h(x+t(p-x),x). \tag{5.9}$$

Instead of requiring that f has a Lipschitz continuous gradient over the convex compact set X, let us alternatively require the following:

Assumption 10. There exists a DGF $h\in\mathcal{H}_\alpha(\mathsf{X})$ and a constant $L_f^h>0$, such that $L_f^h h - f$ is convex on X, and h has a finite curvature on X, that is,

$$\Omega_h^2(\mathsf{X}) := \sup_{x,u\in\mathsf{X},t\in[0,1]} \frac{2D_h(tu+(1-t)x,x)}{t^2} < \infty. \tag{5.10}$$

Note that when choosing h to be the squared Euclidean norm $h(x)=\frac{1}{2}\|x\|^2$ and $L_f^h = L_f$, then Assumption 10 is equivalent to the Lipschitz gradient assumption, where $\Omega_h(\mathsf{X})$ is the diameter of set X. On the other hand, choosing $h(x)=f(x)$ and $L_f^h=L_f$, we essentially retrieve the finite curvature assumption used by Jaggi Jaggi (2013).

Remark 5.1. It is clear that the finite curvature assumption (5.10) is not compatible with the DGF to be essentially smooth on X. We are therefore forced to work with non-steep distance-generating functions.

The analysis of CG under a relative smoothness condition and Assumption 10 runs in the same way as for the classical CG. However, the adaptive step-size is reformulated as

$$\gamma_k = \min\left\{\frac{\langle\nabla f(x^{k-1}), x^{k-1}-p^k\rangle}{L_f^h\Omega_h^2(\mathsf{X})}, 1\right\}.$$

This can be easily seen by replacing the upper model function $f(x) - t\mathsf{e}(x) + L_f^h D_h(x+t(p-x),x)$, with its more conservative bound $f(x) - t\mathsf{e}(x) + \frac{L_f^h t^2}{2}\Omega_h^2(\mathsf{X})$. Of course, in the case of the Euclidean norm this results in a smaller step-size than the adaptive step, which hints towards a deterioration of performance. Nevertheless, this trick allows us to handle convex programming problems outside the Lipschitz smooth case, which is not uncommon in various applications (Bian and Chen, 2015; Bian et al., 2015; Haeser et al., 2018).

5.2. Generalized Conditional Gradient

Introduced by Bach Bach (2015) and Nesterov (2018a), the generalized conditional gradient (GCG) method, is targeted to solve our master problem (P) over a compact set X. To handle the composite case, we need to modify our definition of a linear oracle accordingly.

Definition 5.2. Operator $\mathcal{L}_{\mathsf{X},r}:\mathsf{V}^*\to\mathsf{X}$ is a *generalized linear oracle* (GLO) over set X with respect to function r if for any vector $y\in\mathsf{V}^*$ we have that

$$\mathcal{L}_{\mathsf{X},r}(y) \in \operatorname*{argmin}_{x\in\mathsf{X}} \langle y,x\rangle + r(x).$$

Besides this more demanding oracle assumption, the resulting generalized conditional gradient method is formally identical to the classical CG. In particular, we can consider the target vector

$$p(x) = \mathcal{L}_{\mathsf{X},r}(\nabla f(x)) \qquad \forall x\in\mathsf{X} \tag{5.11}$$

and the same three step size policies as in the classical CG, with the standard step size remaining the same and the obvious modifications for the two other step size policies:

$$\text{Exact line search:}\quad \gamma_k \in \operatorname*{argmin}_{t\in[0,1]} \Psi(x^{k-1}+t(p^k-x^{k-1})), \tag{5.12}$$

$$\text{Adaptive:}\quad \gamma_k = \min\left\{\frac{r(x^{k-1})-r(p^k)+\langle\nabla f(x^{k-1}), x^{k-1}-p^k\rangle}{L_f\|x^{k-1}-p^k\|^2}, 1\right\}. \tag{5.13}$$

The adaptive step size variant is derived from an augmented merit function, taking into consideration the non-smooth composite nature of the underlying optimization problem. Indeed, as again can be learned from the basic theory of variational inequalities (see Nesterov (2007)), the natural merit function for the composite model problem (P) is the non-smooth function

$$\mathsf{e}(x) = \sup_{u\in\mathsf{X}} \Gamma(x,u), \text{ where } \Gamma(x,u) := r(x)-r(u)+\langle\nabla f(x), x-u\rangle. \tag{5.14}$$

By definition, we see that $\mathsf{e}(x)\ge 0$ for all $x\in\mathsf{X}$, with equality if and only if $x\in\mathsf{X}^*$. These basic properties justify our terminology, calling $\mathsf{e}(x)$ a merit function. Of course, $\mathsf{e}(\cdot)$ is also easily seen to be convex. Furthermore, using the convexity of f, one first sees that

$$\langle\nabla f(x), x-u\rangle \ge f(x)-f(u),$$

so that for all $x, u \in \mathrm{dom}(r)$,

$$\Gamma(x,u) \geq r(x) - r(u) + f(x) - f(u) = \Psi(x) - \Psi(u).$$

From here, one immediately arrives at the relation

$$\mathsf{e}(x) \geq \Psi(x) - \Psi_{\min}(\mathsf{X}). \tag{5.15}$$

Clearly, with $r = 0$, the above specification yields the classical CG.

5.2.1. Basic Complexity Properties of GCG

We now turn to prove that the GCG method with one of the above mentioned step-sizes converges at a rate of $O(\frac{1}{k})$. We will derive this rate under the standard Lipschitz smoothness assumption on f. This gives us access to the classical descent lemma (3.15). Combining this with the assumed convexity of the non-smooth function $r(\cdot)$, we readily obtain

$$\begin{aligned}\Psi(x^{k-1} + t(p^k - x^{k-1})) &\leq f(x^{k-1}) - t\langle \nabla f(x^{k-1}), p^k - x^{k-1}\rangle + \frac{t^2 L_f}{2}\|p^k - x^{k-1}\|^2 \\ &\quad + (1-t)r(x^{k-1}) + t r(p^k) \\ &= \Psi(x^{k-1}) - t\mathsf{e}(x^{k-1}) + \frac{t^2 L_f}{2}\|p^k - x^{k-1}\|^2.\end{aligned}$$

Based on this fundamental inequality of the per-iteration decrease, we can deduce the iteration complexity via an induction argument. First, one observes that for each of the three introduced step-size rules (standard, line search and adaptive), one obtains a recursion of the form

$$\Psi(x^{k-1} + \gamma_k(p^k - x^{k-1})) \leq \Psi(x^{k-1}) - \gamma_k \mathsf{e}(x^{k-1}) + \frac{L_f \gamma_k^2}{2}\|p^k - x^k\|^2.$$

When denoting $s^k := \Psi(x^k) - \Psi_{\min}(\mathsf{X})$, $\mathsf{e}^k = \mathsf{e}(x^{k-1})$ and $\Omega^2 \equiv \Omega^2_{\frac{1}{2}\|\cdot\|^2}(\mathsf{X}) = \max_{x,u\in\mathsf{X}} \|x-u\|^2$, this gives us

$$s^k \leq s^{k-1} - \gamma_k \mathsf{e}^k + \frac{L_f \gamma_k^2}{2}\Omega^2.$$

Applying to this recursion Lemma 13.13 in Beck (2017), we deduce the next iteration complexity result for GCG.

Theorem 5.3. *Consider algorithm GCG with one of the step size rules: standard (5.5), line search (5.12), or adaptive (5.13). Then*

$$\Psi(x^k) - \Psi_{\min}(\mathsf{X}) \leq \frac{2\max\{\Psi(x^0) - \Psi_{\min}(\mathsf{X}), L_f\Omega^2\}}{k} \quad \forall k \geq 1.$$

Proof. We give a self-contained proof of this result for the adaptive step-size policy (5.13).

If $\gamma_k = 1$, the per-iteration progress is easily seen that $\mathsf{e}^k \geq L_f\|p^k - x^{k-1}\|^2$ which implies $-\mathsf{e}^k + \frac{L_f}{2}\|p^k - x^{k-1}\|^2 \leq \frac{-\mathsf{e}^k}{2}$ and thus

$$s^k \leq s^{k-1} - \mathsf{e}^k + \frac{L_f}{2}\|p^k - x^{k-1}\|^2 \leq s^{k-1} - \frac{1}{2}\mathsf{e}^k \leq s^{k-1} - \frac{1}{2}s^{k-1} = \frac{1}{2}s^{k-1}$$

where for the last inequality we use (5.15). For $\gamma_k = \frac{r(x^k) - r(p^k) + \langle \nabla f(x^{k-1}), x^{k-1} - p^k\rangle}{L_f\|x^{k-1} - p^k\|^2} = \frac{\mathsf{e}^k}{L_f\|p^k - x^{k-1}\|^2}$, a simple computation reveals

$$s^k \leq s^{k-1} - \frac{(\mathsf{e}^k)^2}{2L_f\|p^k - x^{k-1}\|^2} \leq s^{k-1} - \frac{(\mathsf{e}^k)^2}{2L_f\Omega^2} \leq s^{k-1} - \frac{(s^{k-1})^2}{2L_f\Omega^2}.$$

Summarizing these two cases, we see

$$s^k \leq \max\left\{\frac{1}{2}s^{k-1}, s^{k-1} - \frac{(s^{k-1})^2}{2L_f\Omega^2}\right\}.$$

Thus, the convergence is split into two periods, which are split by $K := \log_2\left(\lceil \frac{s^0}{\min\{L_f\Omega^2, s^0\}}\rceil\right)$. If $k \leq K$ then $s^{k-1} \geq L_f\Omega^2$ and thus $s^k \leq \frac{1}{2}s^{k-1}$, which implies

$$s^k \leq 2^{-k}s^0, \ k \in \{0,1,\ldots,K\}.$$

However, if $k > K$ then $s^{k-1} < \min\{L_f\Omega^2, s^0\}$ and $s^k \leq s^{k-1} - \frac{1}{2L_f\Omega^2}(s^{k-1})^2$, which by induction (see for example Dunn (1979, Lemma 5.1)) implies that

$$s^k \leq \frac{s^K}{1 + \frac{s^K}{2L_f\Omega^2}(k-K)} \leq \frac{2L_f\Omega^2}{2 + (k-K)} \leq \frac{\max\{K,2\}L_f\Omega^2}{k} \leq \frac{2\max\{s^0, L_f\Omega^2\}}{k}, \ k \geq K+1,$$

where the second inequality follows from $s^K < \min\{L_f\Omega^2, s^0\}$, the third inequality follows from $\frac{a}{a+(k-K)}$ being a monotonic function in $a \geq 0$ for any $k \geq K+1$, and the last inequality follows from $K \leq \max\left\{2, \frac{s^0}{L_f\Omega^2}\right\}$. Combining these two results, we have that

$$s^k \leq \frac{2\max\{s^0, L_f\Omega^2\}}{k}.$$

□

5.2.2. Alternative assumptions and step-sizes

A key takeaway from the analysis of the generalized conditional gradient is that one needs to have a bound on the quadratic term of the upper model

$$t \mapsto Q(x, p(x), t, L_f) := \Psi(x) - t\mathsf{e}(x) + \frac{L_f t^2}{2}\|p(x) - x\|^2.$$

Such a bound was given to us essentially for free under the compactness assumption of the domain X, and the Lipschitz-smoothness assumption on the smooth part f. The resulting complexity constant is then determined by $L_f\Omega^2$. Moreover, this constant will be involved in lower bounds of the adaptive step-size rule (5.13). However, such a constant may not be known, or may be expensive to compute. Moreover, a global estimate of this constant is not actually needed for obtaining an upper bound. To see this, we proceed formally as follows. Consider an alternative quadratic function of the form

$$Q(x,p,t,M) := \Psi(x) - t\mathsf{e}(x) + \frac{t^2 M}{2}q(p,x),$$

where $q(p,x)$ is a positive function bounded by some constant C, and choose $\gamma(x,M) := \min\{1, \frac{\mathsf{e}(x)}{Mq(p(x),x)}\}$, for $p(x) = \mathcal{L}_{\mathsf{X},r}(\nabla f(x))$. Let $M > 0$ be a constant such that the point obtained by using this step-size is upper bounded by the corresponding quadratic function, *i.e.*,

$$\Psi((1-\gamma(x,M))x + \gamma(x,M)p(x)) \leq Q(x,p(x),\gamma(x,M),M) < \Psi(x). \tag{5.16}$$

Thus applying the update $x^+ := (1-\gamma(x,M))x + \gamma(x,M)p(x)$, we obtain

$$\Psi(x^+) - \Psi_{\min}(\mathsf{X}) \leq \Psi(x) - \Psi_{\min}(\mathsf{X}) - \frac{1}{2}\mathsf{e}(x) \leq \frac{1}{2}(\Psi(x) - \Psi_{\min}(\mathsf{X}))$$

if $\gamma(x,M) = 1$, and

$$\begin{aligned}\Psi(x^+) - \Psi_{\min}(\mathsf{X}) &\leq \Psi(x) - \Psi_{\min}(\mathsf{X}) - \frac{1}{2Mq(p(x),x)}\mathsf{e}(x)^2 \leq \Psi(x) - \Psi_{\min}(\mathsf{X}) \\ &\quad - \frac{1}{2MC}(\Psi(x) - \Psi_{\min}(\mathsf{X}))^2\end{aligned}$$

if $\gamma(x,M) = \frac{\mathsf{e}(x)}{Mq(p(x),x)}$. If $(x^k)_{k\geq 0}$ is the trajectory defined in this specific way, we get the familiar recursion

$$s^k \leq \min\{\frac{1}{2}s^{k-1}, s^{k-1} - \frac{1}{2M_kC}(s^{k-1})^2\}$$

in terms of the approximation error $s^k := \Psi(x^k) - \Psi_{\min}(\mathsf{X})$, and the local estimates $(M_k)_{k\geq 0}$. Thus, as we are able to bound M_k from above for all iterations of the algorithm, the same convergence as for GCG can be achieved.

Based on this observation, and knowing that M_k must be bounded for Lipschitz smooth objective functions, we can try to determine M_k via a backtracking procedure, as suggested in Pedregosa et al. (2020). By construction, the resulting iterates x^k will induce monotonically decreasing function values so that the whole trajectory x^k will be contained in the level set $\{x \in \mathsf{X} | \Psi(x) \leq \Psi(x^0)\}$. Hence, it is sufficient for $Q(x^k, p^k, t, M)$ to be an upper bound on $\Psi(x_t)$ for any point $x_t = (1-t)x^{k-1} + tp^k$ such that $\Psi(x_t) \leq \Psi(x^0)$. Thus, the Lipschitz continuity (or curvature) can be assumed only on the appropriate level set and there is no need to insist on global Lipschitz smoothness on the entire set X. This insight enabled, for example, proving the $O(1/k)$ convergence rate of CG with adaptive and exact step-size rules when applied to self-concordant functions, which are not necessarily Lipschitz smooth on the predefined set X (Carderera et al., 2021; Dvurechensky et al., 2020a; 2020; Zhao and Freund, 2020).

However, this observation need not apply to the standard step size rule (5.5), since the standard step-size choice does not guarantee that all the iterates remain in the appropriate level set.

To conclude, we reiterate that the step-size choices analyzed here are the most common, but there may be many more choices of step-size which provide similar guarantees. For example, Freund and Grigas (2016) suggests new step-size rules based on an alternative analysis of the CG method that utilizes an updated duality gap. (Nesterov, 2018a) discusses recursive step-size rules, and in Dvurechensky et al. (2020a); Odor et al. (2016) new step-size rules are suggested based on additional assumptions on the problem structure.

5.3. Variants of CG

One of the main drawbacks of CG method is that, in general, it comes with worse complexity bounds than BPGM for strongly convex functions. Indeed, it was shown as early as in 1968 by Canon and Cullum (1968) (see also Lan (2013, 2020)) that the rate of $O(\frac{1}{k})$ is in fact tight, even when the function f is strongly convex. This slow convergence is due to the well-documented zig-zagging effect between different extreme points in X. In the smooth case, where $r = 0$, and the objective function f and the feasible set X are both strongly convex, only a rate of $O(\frac{1}{k^2})$ can be shown (Garber and Hazan, 2015), whereas (Nesterov, 2018a) showed an accelerated $O(\frac{1}{k^2})$ rate of convergence for GCG with strongly convex r ($\mu > 0$). Linear convergence of the CG method can only be proved under additional assumptions regarding the problem structure or location of the optimal solution (see e.g. Beck and Teboulle (2004); Dunn (1979); Epelman and Freund (2000); Guélat and Marcotte (1986); Levitin and Polyak (1966)).

Departing from these somewhat negative results, variants of the classical CG were suggested in order to obtain the desired linear convergence in the case of strongly convex function f. We will discuss four of these variants: Away-step CG, Fully-corrective CG, CG based on a local linear optimization oracle (LLOO), and CG with sliding.

5.3.1. Away-step CG

The away-step variation of CG (AW-CG), first suggested by Wolfe (1970), treats the case where X is a polyhedron. It requires two calls of the LO at each iteration. The first call generates $p^k = \mathcal{L}_{\mathsf{X}}(\nabla f(x^{k-1}))$, defined in the original CG algorithm, while the second call generates an additional vector $u^k = \mathcal{L}_{\mathsf{X}}(-\nabla f(x^{k-1}))$. The two vectors p^k and u^k define the *forward direction* $d^k_{FW} = p^k - x^{k-1}$ and the *away direction* $d^k_A = x^{k-1} - u^k$, respectively. By construction, both of this directions are non-ascent directions. The effectively chosen direction at iteration k is obtained by

$$d^k = \operatorname*{argmax}_{d \in \{d^k_{FW}, d^k_A\}} \langle -\nabla f(x^{k-1}), d\rangle,$$

ensuring the chosen direction is a descent direction for non optimal x^{k-1}, with a corresponding updating step

$$x^k = x^{k-1} + \gamma_k d^k.$$

Here, the choice of the step-size γ_k will also depend on the direction chosen. The first analysis of this algorithm by Guélat and Marcotte (1986) assumes that the step-size is chosen using exact line search over $\gamma_k \in [0, \gamma_{\max}]$, where $\gamma_{\max} := \max\{t \geq 0 : x^{k-1} + td^k \in \mathsf{X}\}$. Under this step-size choice, they prove linear convergence of CG for strongly convex f. However, this rate estimate depends on the distance between the optimal solution and the boundary of set $T \subset \mathsf{X}$, which is the minimal face of X containing the optimal solution. This result was later extended in Lacoste-Julien and Jaggi (2015), with a slight variation on the original algorithm. In this variation, the set X is represented as the convex hull of a finite set of atoms $\mathcal{A}$ (not necessarily containing only its vertices), and a representation of the current iterate as a convex combination of these atoms is maintained throughout the algorithm, *i.e.*, $x^k = \sum_{S^k} \lambda^k_a a$ where $S^k = \{a \in \mathcal{A} : \lambda^k_a > 0\}$ is defined as the set of active atoms. Thus, the AW-CG produces $p^k \in \mathcal{A}$ and $u^k \in S^k$, and the away step maximal step size is respecified as $\gamma_{\max} = \frac{\lambda_{u^k}}{1-\lambda_{u^k}}$. This implies, that using the maximal away-step step-size will not necessarily result on a point on the boundary of X. Thus, when f is strongly convex, Jaggi and Lacoste-Julian Lacoste-Julien and Jaggi (2015) show a linear convergence of AW-CG with a rate that only depends on the geometry of set X, which is captured by the *pyramidal width* parameter. The Pairwise variant of AW-CG, which is also presented and analyzed in Lacoste-Julien and Jaggi (2015), takes $d^k = u^k - p^k$ and $\gamma_{\max} = \lambda_{u^k}$, and has similar analysis.

In Beck and Shtern (2017), Beck and Shtern extend the linear convergence results of AS-CG to functions of the form $f(x) = g(\mathbf{A}x) + \langle b, x\rangle$ where g is a strongly convex function. The linear rate depends on a parameter based on the Hoffman constant, which captures both on the geometry of X as well as matrix $\mathbf{A}$. It is also worth mentioning, a stream of work which shows linear convergence of AS-CG where the strong convexity assumption is replaced by the assumption that sufficient second order optimality conditions, known as Robinson conditions (Robinson, 1982), are satisfied (see for example Damla et al. (2008)).

5.3.2. Fully-corrective CG

The Fully-corrective variant of CG (FC-CG) also involves polyhedral X, and aims to reduce the number of calls to the linear oracle, by replacing them with a more accurate minimization over a convex-hull of some subset $\mathcal{A}^k \subseteq \mathcal{A}$. The heart of the method is a correction routine, which updates the correction atoms $\mathcal{A}^k$ and iterate x^k, and satisfy the following:

$$\begin{aligned} S^k &\subseteq \mathcal{A}^k \\ f(x^k) &\leq \min_{t\in[0,1]} f((1-t)x^{k-1} + tp^k) \\ \epsilon &\geq \max_{s\in S^k} \langle \nabla f(x^k), s - x^k\rangle \end{aligned}$$

where $p^k = \mathcal{L}_{\mathsf{X}}(\nabla f(x^{k-1}))$, and ϵ is a given accuracy parameter. The FC-CG was known by various names depending on the updating scheme of $\mathcal{A}^k$ and x^k (Holloway, 1974; Von Hohenbalken, 1977), and was unified and analyzed to show linear convergence in Lacoste-Julien and Jaggi (2015). The convergence analysis of FC-CG is similar to that of AW-CG, and is based on the correction routine guaranteeing that the forward step is larger than the away-step computed in the previous iteration.

In order to apply FC-CG one must choose a correction routine, and the linear convergence analysis does not take into account the computational cost of this routine. One choice of a correction routine is to apply AS-CG on the subset $\mathcal{A}^k = S^{k-1} \cup \{p^k\}$ until the conditions are satisfied. This correction routine is wise only if efficient linear oracles $\mathcal{L}_{\mathcal{A}^k}$ can be constructed for all k such that their low computational cost balances the routine's iteration complexity.

5.3.3. Enhanced LO based CG

A variant of CG which is based on an enhanced linear minimization oracle, was suggested by Garber and Hazan Garber and Hazan (2016). In this variant, the linear oracle $\mathcal{L}_{\mathsf{X}}(c)$ is replaced by a *local oracle* $\mathcal{L}_{\mathsf{X},\rho}(c, x, \delta)$ with some constant $\rho \geq 1$, which takes an additional radius input δ and returns a point $p \in \mathsf{X}$ satisfying

$$\begin{aligned} \|p - x\| &\leq \rho\delta \\ \langle p, c\rangle &\leq \min_{u\in\mathsf{X}:\|u-x\|\leq\delta} \langle u, c\rangle. \end{aligned}$$

Thus, the only deviation from the CG algorithm is that p^k is obtained by applying $\mathcal{L}_{X,\rho}(\nabla f(x^k), x^k, \delta_k)$ for a suitably chosen sequence $(\delta_k)_k$. The linear convergence for the case where the smooth part f is strongly convex, is obtained by a specific update of δ_k at each step of the algorithm. This update depends on the Lipschitz constant L_f, the strong convexity constant of f, and the parameter ρ. Moreover, despite the fact that LLOO-CG can theoretically be applied to any set X, constructing a general LLOO is challenging. In Garber and Hazan (2016), the authors sug-

gest an LLOO with $\rho = \sqrt{n}$ when the set X is the unit simplex, and generalize it for convex polytopes with $\rho = \sqrt{n}\tilde{\rho}$ where $\tilde{\rho}$ depends on some geometric properties the polytope which may generally not tractably computed. Thus, while the strong convexity and geometric properties of the problem are only used for the analysis of the AW-CG and FC-CG, the associated parameters are explicitly used in the execution of LLOO-CG. The difficulty of accurately estimating the strong convexity and the geometric parameters renders the LLOO-CG less applicable in practice.

5.3.4. CG with gradient sliding

Each iteration of CG requires one call to the linear minimization oracle and one gradient evaluation. Coupled with our knowledge about the iteration complexity of CG, this fact implies that CG requires $O(1/\varepsilon)$ gradient evaluations of the objective function. This is suboptimal, when compared with the $O(1/\sqrt{\varepsilon})$ gradient evaluations for smooth convex optimization, as we will see in Section 6. While it is known that within the linear minimization oracle, the order estimate $O(1/\varepsilon)$ for the number of calls of the LO is unimprovable, in this section we review a method based on the linear minimization oracle which can skip the computation of gradients from time to time. This improves the complexity of LO-based methods and leads us to the *conditional gradient sliding* (S-CG) algorithm introduced by Lan and Zhou Lan and Zhou (2016). S-CG is a numerical optimization method which runs in epochs and overall contains some similarities with accelerated methods, to be thoroughly surveyed in Section 6. S-CG has been described in the context of the smooth convex programming problem for which $r = 0$.

The conditional gradient sliding methods (S-CG)
Input: A linear oracle $\mathcal{L}_X$ a starting point $x^0 \in X$.
$(\beta_k)_k, (\gamma_k)_k$ parameter sequence such that

$$\gamma_1 = 1,\ L_f\gamma_k \le \beta_k,$$
$$\frac{\beta_k\gamma_k}{\Gamma_k} \ge \frac{\beta_{k-1}\gamma_{k-1}}{\Gamma_{k-1}},$$

where

$$\Gamma_k = \begin{cases} 1 & \text{if } k = 1, \\ \Gamma_{k-1}(1-\gamma_k) & \text{if } k \ge 2. \end{cases} \tag{5.17}$$

General step: For $k = 1, 2, \ldots$
Compute

$$z^k = (1-\gamma_k)y^{k-1} + \gamma_k x^{k-1},$$
$$x^k = \mathrm{CndG}(\nabla f(z^k), x^{k-1}, \beta_k, \eta_k),$$
$$y^k = (1-\gamma_k)y^{k-1} + \gamma_k x^k.$$

Similarly to accelerated methods, S-CG keeps track of three sequentially updated sequences. The update of the sequence (x^k) is stated in terms of a procedure CndG, which describes an inner loop of conditional gradient steps. This subroutine aims at approximately solving for the proximal step

$$\min_{x\in X} f(z^k) + \langle \nabla f(z^k), x - z^k\rangle + \frac{\beta_k}{2}\|x - x^{k-1}\|^2$$

up to an accuracy of η_k. As will become clear later, the S-CG can thus be thought of as an approximate version of the accelerated scheme presented in Section 6.1.

The main performance guarantee of the algorithm S-CG is summarized in the following theorem:

Theorem 5.4. *For all $k \ge 1$ and $u \in X$, we have*

$$f(y^k) - f(u) \le \frac{\beta\gamma_k\Omega^2}{2} + \Gamma_k \sum_{i=1}^{k} \frac{\eta_i\gamma_i}{\Gamma_i}, \tag{5.18}$$

The procedure CndG(g, u, β, η)
Input: $u_1 = u, t = 1$.
Output: point $u^+ = \mathrm{CndG}(g, u, \beta, \eta)$.
General step: Let $v_t = \mathrm{argmax}_{x\in X}\langle g + \beta(u_t - u), u_t - x\rangle$
If $V_{g,u,\beta}(u_t) = \langle g + \beta(u_t - u), u_t - v_t\rangle \le \eta$, set $u^+ = u_t$;
else, set $u_{t+1} = (1-\alpha_t)u_t + \alpha_t v_t$, where

$$\alpha_t = \min\left\{1, \frac{\langle \beta(u - u_t) - g, v_t - u_t\rangle}{\beta\|v_t - u_t\|^2}\right\}.$$

Set $t \leftarrow t + 1$. Repeat General step.

where $\Omega \equiv \Omega_{\frac{1}{2}\|\cdot\|^2}(X)$. The number of calls of the linear minimization oracle is bounded by $\lceil \frac{6\beta_k\Omega^2}{\eta_k} \rceil$. In particular, if the parameter sequences in S-CG are chosen as

$$\beta_k = \frac{3L_f}{k+1}, \gamma_k = \frac{3}{k+2}, \eta_k = \frac{L_f\Omega^2}{k(k+1)},$$

then

$$f(y^k) - f(u) \le \frac{15L_f\Omega^2}{2(k+1)(k+2)}.$$

As a consequence, the total number of calls of the function gradients and the LO oracle is bounded by $O\left(\sqrt{\frac{L_f\Omega^2}{\varepsilon}}\right)$, and $O(L_f\Omega^2/\varepsilon)$, respectively.

6. Accelerated Methods

In previous sections we focused on simple first-order methods with sublinear convergence guarantees in the convex case, and linear convergence in the strongly convex case. Towards the end of the discussion in Section 3, we pointed out the possibility to accelerate simple iterative schemes via suitably defined extrapolation steps. In this last section of the survey, we are focusing on such *accelerated methods*. The idea of acceleration dates back to 1980's. The rationale for this research direction is the desire to understand the computational boundaries of solving optimization problems. Of particular interest has been the unconstrained smooth, and strongly convex optimization problem. This would be covered by our generic model (P) by setting $r = 0, X = V = \mathbb{R}^n$ and f strongly convex with parameter $\mu_f > 0$ and L_f-smooth. The standard approach to quantify the computational hardness of optimization problems is through the *oracle model* of optimization; Upon receiving a query point x, the oracle reports the corresponding function value $f(x)$, and in first-order models, the function gradient $\nabla f(x)$ as well. In their seminal work, Nemirovski and Yudin (1983) showed that for any first-oder optimization algorithm, there exists an L_f-smooth (with some $L_f > 0$) and convex function $f : \mathbb{R}^n \to \mathbb{R}$ such that the number of queries required to obtain an ε-optimal solution x^* which satisfies

$$f(x^*) < \min_x f(x) + \varepsilon,$$

is at least of the order of $\min\{n, \sqrt{L_f/\mu_f}\}\ln(1/\varepsilon)$ if $\mu_f > 0$ and $\min\{n\ln(1/\varepsilon), \sqrt{L_f/\varepsilon}\}$, if $\mu_f = 0$. This bound, obtained by information-theoretical arguments, turned out to be tight. Nemirovski Nemirovski (1982) proposed a method achieving the optimal rate $O(1/k^2)$ via a combination of standard gradient steps with the classical center of gravity method, which required additional small-dimensional minimization, see also a recent paper (Nesterov et al., 2020). Nesterov (1983) proposed an optimal method with explicit step-sizes, which is nowadays known as Nesterov's accelerated gradient method.

6.1. Accelerated Gradient Method

In this section we consider one of the multiple variants of an Accelerated Gradient Method. This variant is close to the accelerated proximal

method in Tseng (2008), which has been very influential to the field. Another very influential version of the accelerated method, especially in applications, is the FISTA algorithm (Beck and Teboulle, 2009a), which is excellently described in Beck (2017). A recent review on accelerated methods is d'Aspremont et al., 2021. The version we present here is inspired by the *Method of Similar Triangles* (Gasnikov and Nesterov, 2018; Nesterov, 2018b). For an illustration see Figure 1

Accelerated schemes generically produce three sequences (u^k, x^k, y^k), which are iteratively constructed via specifically designed inertial, relaxation and gradient steps. The relative magnitude of inertia and relaxations are governed by control sequences $(A_k)_k$ and $(\alpha_k)_k$.

The version we present below, is very flexible and allows one to obtain accelerated methods for many settings. As a particular example, below in Section 6.4, we show how a slight modification of this method allows one to obtain universal accelerated gradient method.

Our aim is to solve the composite model problem (P) within a general Bregman proximal setup, formulated in Section 3.2. We are given a DGF $h \in \mathcal{H}_1(\mathsf{X})$. The scaling of the strong convexity parameter to the value 1 actually is without loss of generality, modulo a constant rescaling of the employed DGF

The Accelerated Bregman Proximal Gradient Method (A-BPGM)

Input: pick $x^0 = u^0 = y^0 \in \mathrm{dom}(r) \cap \mathsf{X}^\circ$, set $A_0 = 0$

General step: For $k = 0, 1, \dots$ do:

Find α_{k+1} from quadratic equation $A_k + \alpha_{k+1} = L_f \alpha_{k+1}^2$. Set $A_{k+1} = A_k + \alpha_{k+1}$.

Set $y^{k+1} = \frac{\alpha_{k+1}}{A_{k+1}} u^k + \frac{A_k}{A_{k+1}} x^k$.

Set

$$u^{k+1} = \mathcal{P}^h_{\alpha_{k+1} r}(u^k, \alpha_{k+1} \nabla f(y^{k+1})) = \mathrm{argmin}_{x \in \mathsf{X}} \left\{ \alpha_{k+1} \left(f(y^{k+1}) + \langle \nabla f(y^{k+1}), x - y^{k+1} \rangle + r(x) \right) + D_h(x, u^k) \right\}.$$

Set $x^{k+1} = \frac{\alpha_{k+1}}{A_{k+1}} u^{k+1} + \frac{A_k}{A_{k+1}} x^k$.

We start the analysis A-BPGM applying the descent Lemma property (3.15) which holds for any two points due to L_f-smoothness:

$$\Psi(x^{k+1}) = f(x^{k+1}) + r(x^{k+1}) \le f(y^{k+1}) + \langle \nabla f(y^{k+1}), x^{k+1} - y^{k+1} \rangle + \frac{L_f}{2} \|x^{k+1} - y^{k+1}\|^2 + r(x^{k+1}). \tag{6.1}$$

Let us next consider the squared norm term. Using the definition of x^{k+1}, y^{k+1} and the quadratic equation for α_{k+1} given in the listing of A-BPGM, as well as the 1-strong convexity of the Bregman divergence, i.e. (3.5), we obtain

$$\begin{aligned} \frac{L_f}{2} \|x^{k+1} - y^{k+1}\|^2 &= \frac{L_f}{2} \left\| \frac{\alpha_{k+1}}{A_{k+1}} u^{k+1} + \frac{A_k}{A_{k+1}} x^k - \left(\frac{\alpha_{k+1}}{A_{k+1}} u^k + \frac{A_k}{A_{k+1}} x^k \right) \right\|^2 \\ &= \frac{L_f \alpha_{k+1}^2}{2 A_{k+1}^2} \|u^{k+1} - u^k\|^2 = \frac{1}{2 A_{k+1}} \|u^{k+1} - u^k\|^2 \le \frac{1}{A_{k+1}} D_h(u^{k+1}, u^k). \end{aligned} \tag{6.2}$$

Next, we consider the remaining terms in the r.h.s. of (6.1). Substituting x^{k+1} and using $A_{k+1} = A_k + \alpha_{k+1}$, we obtain

$$\begin{aligned} & f(y^{k+1}) + \langle \nabla f(y^{k+1}), x^{k+1} - y^{k+1} \rangle + r(x^{k+1}) \\ &= \left(\frac{\alpha_{k+1}}{A_{k+1}} + \frac{A_k}{A_{k+1}} \right) f(y^{k+1}) + \left\langle \nabla f(y^{k+1}), \frac{\alpha_{k+1}}{A_{k+1}} u^{k+1} + \frac{A_k}{A_{k+1}} x^k - \left(\frac{\alpha_{k+1}}{A_{k+1}} + \frac{A_k}{A_{k+1}} \right) y^{k+1} \right\rangle \\ &\quad + r\left(\frac{\alpha_{k+1}}{A_{k+1}} u^{k+1} + \frac{A_k}{A_{k+1}} x^k \right) \\ &\le \frac{A_k}{A_{k+1}} \left(f(y^{k+1}) + \langle \nabla f(y^{k+1}), x^k - y^{k+1} \rangle + r(x^k) \right) \\ &\quad + \frac{\alpha_{k+1}}{A_{k+1}} \left(f(y^{k+1}) + \langle \nabla f(y^{k+1}), u^{k+1} - y^{k+1} \rangle + r(u^{k+1}) \right) \\ &\le \frac{A_k}{A_{k+1}} \left(f(x^k) + r(x^k) \right) + \frac{\alpha_{k+1}}{A_{k+1}} \left(f(y^{k+1}) + \langle \nabla f(y^{k+1}), u^{k+1} - y^{k+1} \rangle + r(u^{k+1}) \right) \\ &= \frac{A_k}{A_{k+1}} \Psi(x^k) + \frac{\alpha_{k+1}}{A_{k+1}} \left(f(y^{k+1}) + \langle \nabla f(y^{k+1}), u^{k+1} - y^{k+1} \rangle + r(u^{k+1}) \right), \end{aligned} \tag{6.3}$$

where in the first inequality used the convexity of r, and in the second inequality we used the convexity of f. Now we plug (6.2) and (6.3) into (6.1) to obtain

$$\begin{aligned} \Psi(x^{k+1}) &\le \frac{A_k}{A_{k+1}} \Psi(x^k) + \frac{\alpha_{k+1}}{A_{k+1}} \left(f(y^{k+1}) + \langle \nabla f(y^{k+1}), u^{k+1} - y^{k+1} \rangle + r(u^{k+1}) \right) \\ &\quad + \frac{1}{A_{k+1}} D_h(u^{k+1}, u^k) \\ &= \frac{A_k}{A_{k+1}} \Psi(x^k) + \frac{1}{A_{k+1}} \left[\alpha_{k+1} \left(f(y^{k+1}) + \langle \nabla f(y^{k+1}), u^{k+1} - y^{k+1} \rangle + r(u^{k+1}) \right) \right. \\ &\quad \left. + D_h(u^{k+1}, u^k) \right]. \end{aligned} \tag{6.4}$$

Given the definition of u^{k+1} as a Prox-Mapping, we can apply (3.12) by substituting $x^+ = u^{k+1}$, $x = u^k$, $\gamma = \alpha_{k+1}$. In this way, we obtain, for any $u \in \mathsf{X}$,

$$\begin{aligned} \Psi(x^{k+1}) &\le \frac{A_k}{A_{k+1}} \Psi(x^k) + \frac{1}{A_{k+1}} \left(\alpha_{k+1} (f(y^{k+1}) + \langle \nabla f(y^{k+1}), u^{k+1} - y^{k+1} \rangle + r(u^{k+1})) \right. \\ &\quad \left. + D_h(u^{k+1}, u^k) \right) \\ &\overset{(3.12)}{\le} \frac{A_k}{A_{k+1}} \Psi(x^k) + \frac{1}{A_{k+1}} \left(\alpha_{k+1} (f(y^{k+1}) + \langle \nabla f(y^{k+1}), u - y^{k+1} \rangle + r(u)) \right. \\ &\quad \left. + D_h(u, u^k) - D_h(u, u^{k+1}) \right) \\ &\le \frac{A_k}{A_{k+1}} \Psi(x^k) + \frac{\alpha_{k+1}}{A_{k+1}} (f(u) + r(u)) + \frac{1}{A_{k+1}} D_h(u, u^k) - \frac{1}{A_{k+1}} D_h(u, u^{k+1}) \\ &= \frac{A_k}{A_{k+1}} \Psi(x^k) + \frac{\alpha_{k+1}}{A_{k+1}} \Psi(u) + \frac{1}{A_{k+1}} D_h(u, u^k) - \frac{1}{A_{k+1}} D_h(u, u^{k+1}), \end{aligned} \tag{6.5}$$

where we also used convexity of f. Multiplying both sides of the last inequality by A_{k+1}, summing these inequalities from $k = 0$ to $k = N - 1$, and using that $A_N - A_0 = \sum_{k=0}^{N-1} \alpha_{k+1}$, we obtain

$$A_N \Psi(x^N) \le A_0 \Psi(x^0) + (A_N - A_0) \Psi(u) + D_h(u, u^0) - D_h(u, u^N). \tag{6.6}$$

Since $A_0 = 0$, we can choose $u = x^* \in \mathrm{argmin}\{ D_h(u, u^0) | u \in \mathsf{X}^* \} \subseteq \mathsf{X}^*$ and $D_h(x^*, u^N) \ge 0$, so that, for all $N \ge 1$,

$$\Psi(x^N) - \Psi_{\min}(\mathsf{X}) \le \frac{D_h(x^*, u^0)}{A_N}, \quad D_h(x^*, u^N) \le D_h(x^*, u^0). \tag{6.7}$$

So, we see from the second inequality that the Bregman distance between the iterates $\{u^N\}_{N \ge 0}$ and the solution x^* is bounded by the Bregman distance between the starting point and the solution x^*. Then, from the inequality $D_h(x^*, u^N) \ge \frac{1}{2} \|x^* - u^N\|^2$ it follows that $\|x^* - u^N\|$ is bounded for any N, which leads to the existence of a subsequence converging to x^* by the continuity of Ψ. To obtain the convergence rate in terms of the objective residual it remains to estimate the sequence A_N from below.

We prove by induction that $A_k \ge \frac{(k+1)^2}{4 L_f}$. For $k = 1$ this inequality holds as equality since $A_0 = 0$, and, hence, $A_1 = \alpha_1 = \frac{1}{L_f}$. Let us prove the induction step. From the quadratic equation $A_k + \alpha_{k+1} = L_f \alpha_{k+1}^2$, we have

$$\alpha_{k+1} = \frac{1}{2 L_f} + \sqrt{\frac{1}{4 L_f^2} + \frac{A_k}{L_f}} \ge \frac{1}{2L} + \sqrt{\frac{A_k}{L_f}} \ge \frac{1}{2 L_f} + \frac{k+1}{2 L_f} = \frac{k+2}{2 L_f}. \tag{6.8}$$

$$A_{k+1} = A_k + \alpha_{k+1} \ge \frac{(k+1)^2}{4 L_f} + \frac{k+2}{2 L_f} = \frac{k^2 + 2k + 1 + 2k + 4}{4 L_f} \ge \frac{(k+2)^2}{4 L_f}. \tag{6.9}$$

Thus, combining (6.9) with (6.7), we obtain that the A-BPGM has optimal convergence rate:

$$\Psi(x^N) - \Psi_{\min}(\mathsf{X}) \le \frac{4 L_f D_h(x^*, u^0)}{(N+1)^2}. \tag{6.10}$$

Closing Remarks Mainly driven by applications in imaging and machine learning, the research on acceleration techniques has been very productive in the last 20 years. During this time span it received extensions to composite optimization (Beck and Teboulle, 2009a; Nesterov,

2013), general proximal setups (Nesterov, 2005b; 2018b), stochastic optimization problems (Dvurechensky et al., 2018a; Dvurechensky and Gasnikov, 2016; Ghadimi and Lan, 2012; 2013; Lan, 2012), optimization with inexact oracle (Cohen et al., 2018; d'Aspremont, 2008; Devolder et al., 2014; Dvurechensky and Gasnikov, 2016; Stonyakin et al., 2020), variance reduction methods (Allen-Zhu, 2017; Frostig et al., 2015; Lan and Zhou, 2017; Lin et al., 2015; Zhang and Xiao, 2015), random coordinate descent (Fercoq and Richtárik, 2015; Lee and Sidford, 2013; Lin et al., 2014; Nesterov, 2012; Nesterov and Stich, 2017; Shalev-Shwartz and Zhang, 2014) and other randomized methods such as randomized derivative-free methods (Dvurechensky et al., 2017; Gorbunov et al., 2018; Nesterov and Spokoiny, 2017; Vorontsova et al., 2019b) and randomized directional search (Dvurechensky et al., 2017; 2021; Vorontsova et al., 2019a), second-order methods (Monteiro and Svaiter, 2013; Nesterov, 2008), and even high-order methods (Baes, 2009; Gasnikov et al., 2019; Nesterov, 2019). Under additional assumptions on the Bregman divergence it is possible to propose accelerated Bregman proximal gradient method in the setting of relative smoothness (Hanzely et al., 2021) and relative strong convexity (Dvurechensky et al., 2021; Hendrikx et al., 2020) (see Section 3.3.3 for the definition of relative smoothness). Yet, the negative result of (Dragomir et al., 2021) suggest that, in general, the acceleration in the relative smoothness setting is not possible.

As it was mentioned above, accelerated gradient method in the form of A-BPGM can serve as a template for many acceleration techniques. The examples of accelerated methods which have a close form include primal-dual accelerated methods (Dvurechensky et al., 2018b; Lin et al., 2019; Tseng, 2008), random coordinate descent and other randomized algorithms (Diakonikolas and Orecchia, 2018; Dvurechensky et al., 2017; Fercoq and Richtárik, 2015), methods for stochastic optimization (Dvurechensky et al., 2018a; Lan, 2012), methods with inexact oracle (Cohen et al., 2018) and inexact model of the objective (Gasnikov and Tyurin, 2019; Stonyakin et al., 2020). Moreover, only using this one-projection version it was possible to obtain accelerated gradient methods with inexact model of the objective (Gasnikov and Tyurin, 2019), accelerated decentralized distributed algorithms for stochastic convex optimization (Dvinskikh et al., 2019; Gorbunov et al., 2019; Rogozin et al., 2021), and accelerated method for stochastic optimization with heavy-tailed noise (Gorbunov et al., 2020; 2021). The key to the last two results is the proof that the sequence generated by the one-projection accelerated gradient method is bounded with large probability, which, to our knowledge, is not possible to prove for other types of accelerated methods applied to stochastic optimization problems.

6.2. Linear Convergence

Under additional assumptions, we can use the scheme A-BPGM to obtain a linear convergence rate, or, in other words, logarithmic in the desired accuracy complexity bound. One such possible assumption is that $\Psi(x)$ satisfies a quadratic error bound condition for some $\mu > 0$:

$$\Psi(x) - \Psi_{\min}(X) \geq \frac{\mu}{2}\|x - x^*\|^2. \tag{6.11}$$

This is a weaker assumption than the assumption that $\Psi(x)$ is μ-strongly convex with $\mu > 0$. For a review of different additional conditions which allow to obtain linear convergence rate we refer the reader to Bolte et al. (2017); Necoara et al. (2019). The linear convergence rate can be obtained under quadratic error bound condition by a widely used restart technique, which dates back to Nemirovskii and Nesterov (1985); Nesterov (1983).

To apply the restart technique, we make several additional assumptions. First, without loss of generality, we assume that $0 \in X$, $0 = \arg\min_{x \in X} h(x)$ and $h(0) = 0$. Second, we assume that we are given a starting point $x^0 \in X$ and a number $R_0 > 0$ such that $\|x^0 - x^*\|^2 \leq R_0^2$. Finally, we make the assumption that h is bounded on the unit ball (Juditsky and Nesterov, 2014) in the following sense. Assume that x^* is some fixed point and x is such that $\|x - x^*\|^2 \leq R^2$, then

$$h\left(\frac{x - x^*}{R}\right) \leq \frac{\Omega}{2}, \tag{6.12}$$

where Ω is some known number. For example, in the Euclidean setup $\Omega = 1$, and other examples are given in Juditsky and Nesterov (2014, Section 2.3), where typically $\Omega = O(\ln n)$.

The Restarted Accelerated Bregman Proximal Gradient Method (R-A-BPGM)
Input: $z^0 \in \mathrm{dom}(r) \cap X^\circ$ such that $\|z^0 - x^*\|^2 \leq R_0^2$, Ω, L_f, μ.
General step: For $p = 0, 1, \ldots$ do:
Set $h_p(x) = R_p^2 h\left(\frac{x - z^p}{R_p}\right)$, where $R_p := R_{p-1}/2 = R_0 \cdot 2^{-p}$.
Make $N = \left\lceil 2\sqrt{\frac{\Omega L_f}{\mu}} \right\rceil - 1$ steps of A-BPGM with starting point $x^0 = z^p$ and proximal setup given by DFG $h_p(x)$
Set $z^{p+1} = x^N$.

We next use the above assumptions to show the accelerated logarithmic complexity of R-A-BPGM, i.e. that the number of Bregman proximal steps to find a point $\hat{x}$ such that $f(\hat{x}) - f(x^*) \leq \varepsilon$ is proportional to $\sqrt{L_f/\mu}\log_2(1/\varepsilon)$ instead of $(L_f/\mu)\log_2(1/\varepsilon)$ for the BPGM under the error bound condition. The idea of the proof is to show by induction that, for all $p \geq 0$, $\|z^p - x^*\|^2 \leq R_p^2$. For $p = 0$ this holds by the assumption on z^0 and R_0. So, next we prove an induction step from $p-1$ to p. Using the definition of h_{p-1}, assumptions about h, and the inductive assumption, we have

$$D_{h_{p-1}}(x^*, z^{p-1}) \leq h_{p-1}(x^*) = R_{p-1}^2 h\left(\frac{z^{p-1} - x^*}{R_{p-1}}\right) \overset{(6.12)}{\leq} \frac{\Omega R_{p-1}^2}{2}. \tag{6.13}$$

Thus, applying the error bound condition (6.11), the bound (6.10) and our choice of the number of steps N, we obtain

$$\frac{\mu}{2}\|z^p - x^*\|^2 \overset{(6.11)}{\leq} \Psi(z^p) - \Psi_{\min}(X) = \Psi(x^N) - \Psi_{\min}(X) \overset{(6.10)}{\leq} \frac{L_f D_{h_{p-1}}(x^*, z^{p-1})}{(N+1)^2} \overset{(6.13)}{\leq} \frac{L_f \Omega R_{p-1}^2}{2(N+1)^2}$$
$$\leq \frac{\mu R_{p-1}^2}{8} = \frac{\mu R_p^2}{2}.$$

So, we obtain that $\|z^p - x^*\| \leq R_p = R_0 \cdot 2^{-p}$ and $\Psi(z^p) - \Psi_{\min}(X) \leq \frac{\mu R_0^2 \cdot 2^{-2p}}{2}$. To estimate the total number of basic steps of A-BPGM to achieve $\Psi(z^p) - \Psi_{\min}(X) \leq \varepsilon$, we need to multiply the sufficient number of restarts $\hat{p} = \left\lceil \frac{1}{2}\log_2 \frac{\mu R_0^2}{2\varepsilon} \right\rceil$ by the number of A-BPGM steps N in each restart. This leads to the complexity estimate $O\left(\sqrt{\frac{\Omega L_f}{\mu}}\log_2 \frac{\mu R_0^2}{\varepsilon}\right)$ which is optimal (Nemirovski and Yudin, 1983; Nesterov, 2018b) for first-order methods applied to smooth strongly convex optimization problems.

Closing Remarks The restart technique which we used above was extended in the past 20 years to many settings including problems with non-quadratic error bound condition (Juditsky and Nesterov, 2014; Roulet and d'Aspremont, 2017), stochastic optimization problems (Bayandina et al., 2018; Dvurechensky and Gasnikov, 2016; Gasnikov and Dvurechensky, 2016; Ghadimi and Lan, 2013; Juditsky and Nesterov, 2014), methods with inexact oracle (Dvurechensky and Gasnikov, 2016; Gasnikov and Dvurechensky, 2016), randomized methods (Allen-Zhu and Hazan, 2016; Fercoq and Qu, 2020), conditional gradient (Kerdreux et al., 2019; Lan, 2013), variational inequalities and saddle-point problems (Stonyakin et al., 2018; 2020), methods for constrained optimization problems (Bayandina et al., 2018). There are versions of this technique even for discrete and submodular optimization problems (Pokutta, 2020).

A possible drawback of the restart scheme is that one has to know an estimate R_0 for $\|z^0 - x^*\|$. It is possible to avoid this by

directly incorporating the parameter μ into the steps of A-BPGM, see e.g. d'Aspremont et al. (2021); Devolder (2013); Lan (2020); Nesterov (2018b); Stonyakin et al. (2020). Yet, in this case, a stronger assumption that $\Psi(x)$ is strongly convex or relatively strongly convex (Lu et al., 2018) is used. The second drawback of the restart technique and direct incorporation of μ into the steps, is that both require to know the value of the parameter μ. This is in contrast to non-accelerated BPGM, which using the same step-size as in the non-strongly convex case automatically has linear convergence rate and complexity $O\left(\frac{L_f}{\mu}\log_2\frac{\mu R_0^2}{\varepsilon}\right)$, see e.g. Stonyakin et al. (2020, 2019). Several recipes on how to restart accelerated methods with only rough estimates of the parameter μ are proposed in Fercoq and Qu (2020) and a parameter-free accelerated method is proposed in Carderera et al. (2021); Nesterov (2013).

6.3. Smooth minimization of non-smooth functions

An important observation made during the last 20 years of development of first-order methods for convex programming is that there is a large gap between the optimal convergence rate for black-box non-smooth optimization problems, i.e. $O(1/\sqrt{N})$ and the optimal convergence rate for black-box smooth optimization problems, i.e. $O(1/N^2)$. Certainly, there arises the need to understand how this significant gap can be reduced. An important step towards that direction is *Nesterov's smoothing technique* (Nesterov, 2005b). To motivate this approach, let us make the following thought experiment. Assume that we minimize a smooth function by N steps of A-BPGM, i.e. solve problem (P) with $r = 0$. Then at each iteration we observe first-order information $(f(y^{k+1}), \nabla f(y^{k+1}))$ and can construct a *non-smooth* piecewise linear approximation of f as $g(x) = \max_{k=1,\dots,N}\{f(y^{k+1}) + \langle \nabla f(y^{k+1}), x - y^{k+1}\rangle\}$. If we now make N steps of A-BPGM with the same starting point to minimize $g(x)$, and choose the appropriate subgradients of $g(\cdot)$, the steps will be absolutely the same as when we minimized $f(x)$, and we will be able to minimize a *non-smooth* function g with much faster rate $1/N^2$ than the lower bound $1/\sqrt{N}$. This leads to a way of trying to find a sufficiently wide class of non-smooth functions which can be efficiently minimized by A-BPGM. To do so in a systematic way, we have to leave the pure black-box model of convex programming.

Consider the model problem (P), with the added assumption that the non-smooth part admits a Fenchel representation of the form

$$r(x) = \max_{w\in\mathsf{W}}\{\langle \mathbf{A}x, w\rangle - \kappa(w)\}. \tag{6.14}$$

Here, $\mathsf{W} \subseteq \mathsf{E}$ is a compact convex subset of a finite-dimensional real vector space E, and $\kappa : \mathsf{W} \to \mathbb{R}$ is a continuous convex function on W. $\mathbf{A}$ is a linear operator from V to E^*. This additional structure of the problem gives rise to a min-max formulation of (P), given by

$$\min_{x\in\mathsf{X}}\max_{w\in\mathsf{W}}\{f(x) + \langle \mathbf{A}x, w\rangle - \kappa(w)\}. \tag{6.15}$$

The main idea of Nesterov is based on the observation that the function r can be well approximated by a class of smooth convex functions, defined as follows. Let $h_w \in \mathcal{H}_1(\mathsf{W})$ with a nonrestrictive assumption that $\min_{z\in\mathsf{W}} h_w(z) = 0$. For given $\tau > 0$, define the function

$$\Psi_\tau(x) := f(x) + \max_{w\in\mathsf{W}}\{\langle \mathbf{A}x, w\rangle - \kappa(w) - \tau h_w(w)\}. \tag{6.16}$$

Since h_w is 1-strongly convex on the compact convex set W, the inner maximization problem is strongly concave and hence admits a unique solution. We denote by $\widehat{w}_\tau(x)$ this optimal solution for a fixed x. The main technical lemma needed for the analysis is the following.

Proposition 6.1 (Nesterov (2005b)). *The function $\Psi_\tau(x)$ is well defined, convex and continuously differentiable at any $x \in \mathsf{X}$ with $\nabla\Psi_\tau(x) = \nabla f(x) + \mathbf{A}^*\widehat{w}_\tau(x)$. Moreover, $\nabla\Psi_\tau(x)$ is Lipschitz continuous with constant $L_\tau = L_f + \frac{\|\mathbf{A}\|^2_{\mathsf{V},\mathsf{E}}}{\tau}$.*

Here the adjoint operator $\mathbf{A}^*$ is defined by equality $\langle \mathbf{A}x, w\rangle_{\mathsf{E}} = \langle \mathbf{A}^*w, x\rangle_{\mathsf{V}}$ and the norm of the operator $\|\mathbf{A}\|_{\mathsf{V},\mathsf{E}}$ is defined by $\|\mathbf{A}\|_{\mathsf{V},\mathsf{E}} = \max_{x,w}\{\langle \mathbf{A}x, w\rangle : \|x\|_{\mathsf{V}} = 1, \|w\|_{\mathsf{E}} = 1\}$. Since W is bounded, $\Psi_\tau(x)$ is a uniform approximation for the function Ψ, namely, for all $x \in \mathsf{X}$,

$$\Psi_\tau(x) \le \Psi(x) \le \Psi_\tau(x) + \tau\Theta_{h_w}(\mathsf{W}), \tag{6.17}$$

where $\Theta_{h_w}(\mathsf{W}) := \max\{h_w(z) | z \in \mathsf{W}\}$, assumed to be a finite number. Then, the idea is to choose τ sufficiently small and apply accelerated gradient method to minimize $\Psi_\tau(x)$ on X with a DGF $h_x \in \mathcal{H}_1(\mathsf{X})$ with a nonrestrictive assumption that $\min_{u\in\mathsf{X}} h_x(u) = 0$. Doing this, and assuming that $\Theta_{h_x}(\mathsf{X}) = \max\{h_x(u) | u \in \mathsf{X}\} < \infty$, we can apply the result (6.10) to $\Psi_\tau(x)$ and, using (6.17), to obtain

$$\begin{aligned}0 \le \Psi(x^N) - \Psi_{\min}(\mathsf{X}) &\le \Psi_\tau(x^N) + \tau\Theta_{h_w}(\mathsf{W}) - \Psi_\tau(x^*) \le \Psi_\tau(x^N) + \tau\Theta_{h_w}(\mathsf{W}) - \Psi_\tau(x^*_\tau) \le \tau\Theta_{h_w}(\mathsf{W})\\ &\quad + \frac{4L_\tau\Theta_{h_x}(\mathsf{X})}{(N+1)^2}\\ &= \tau\Theta_{h_w}(\mathsf{W}) + \frac{4\|\mathbf{A}\|^2_{\mathsf{V},\mathsf{E}}\Theta_{h_x}(\mathsf{X})}{\tau(N+1)^2} + \frac{4L_f\Theta_{h_x}(\mathsf{X})}{(N+1)^2}.\end{aligned}$$

Choosing τ to minimize the r.h.s., i.e. $\tau = \frac{2\|\mathbf{A}\|_{\mathsf{V},\mathsf{E}}}{N+1}\sqrt{\frac{\Theta_{h_x}(\mathsf{X})}{\Theta_{h_w}(\mathsf{W})}}$, we obtain

$$0 \le \Psi(x^N) - \Psi_{\min}(\mathsf{X}) \le \frac{4\|\mathbf{A}\|_{\mathsf{V},\mathsf{E}}\sqrt{\Theta_{h_x}(\mathsf{X})\Theta_{h_w}(\mathsf{W})}}{N+1} + \frac{4L_f\Theta_{h_x}(\mathsf{X})}{(N+1)^2}. \tag{6.18}$$

A more careful analysis in the proof of Nesterov (2005b, Theorem 3), allows also to obtain an approximate solution to the conjugate problem

$$\max_{w\in\mathsf{W}}\{\psi(w) := -\kappa(w) + \min_{x\in\mathsf{X}}(\langle \mathbf{A}x, w\rangle + f(x))\}. \tag{6.19}$$

In each iteration of A-BPGM, the optimizer needs to calculate $\nabla\Psi_\tau(y^{k+1})$, which requires to calculate $\widehat{w}_\tau(y^{k+1})$. This information is aggregated to obtain the vector $\widehat{w}^N = \sum_{k=0}^{N-1}\frac{a_{k+1}}{A_{k+1}}\widehat{w}_\tau(y^{k+1})$ and is used to obtain the following primal-dual result

$$0 \le \Psi(x^N) - \Psi_{\min}(\mathsf{X}) \le \Psi(x^N) - \psi(\widehat{w}^N) \le \frac{4\|\mathbf{A}\|_{\mathsf{V},\mathsf{E}}\sqrt{\Theta_{h_x}(\mathsf{X})\Theta_{h_w}(\mathsf{W})}}{N+1} + \frac{4L_f\Theta_{h_x}(\mathsf{X})}{(N+1)^2}. \tag{6.20}$$

In both cases using the special structure of the problem it is possible to obtain convergence rate $O(1/N)$ for non-smooth optimization, which is better than the lower bound $O(1/\sqrt{N})$ for general non-smooth optimization problems.

We illustrate the smoothing technique by two examples of piecewise-linear minimization; see also Figure 2

Example 6.1 (Uniform fit)**.** Consider the problem of finding a uniform fit of some signal $b \in \mathsf{E}$, given linear observations $\mathbf{A}x$. where $\mathbf{A} : \mathsf{V} \to \mathsf{E}$ is a bounded linear operator. This problem amount to minimize the non-smooth function $\|\mathbf{A}x - b\|_\infty$. Of course, this problem can be equivalently formulated as an LP, however in case where the dimensionality of the parameter vector x is large, such a direct approach could turn out to be not very practical. Adopting the just introduced smoothing technology, the representation (6.15) can be obtained using the definition of the dual norm $\|\cdot\|_1$, i.e. $\|\mathbf{A}x - b\|_\infty = \max_{w:\|w\|_1\le 1}\langle \mathbf{A}x - b, w\rangle$. Yet, a better representation is obtained using the unit simplex $\mathsf{W} = \{w \in \mathbb{R}^{2m}_+ | \sum_{i=1}^{2m} w_i = 1\}$, matrix $\hat{\mathbf{A}} = [\mathbf{A}; -\mathbf{A}]$, and vector $\hat{b} = [b; -b]$. For the set W, a natural Bregman setup is the norm $\|w\|_{\mathsf{E}} = \|w\|_1$ and the Boltzmann-Shannon entropy $h_w(w) = \ln 2m + \sum_{i=1}^m w_i \ln w_i$. This gives

$$\Psi_\tau(x) = \max_{w\in\mathsf{W}}\{\langle \hat{\mathbf{A}}x - \hat{b}, w\rangle - \tau h_w(w)\} = \tau\ln\left(\frac{1}{2m}\sum_{i=1}^m \exp\left(\frac{\langle a_i, x\rangle - b_i}{\tau}\right) + \exp\left(-\frac{\langle a_i, x\rangle - b_i}{\tau}\right)\right),$$

which is recognized as a softmax function.

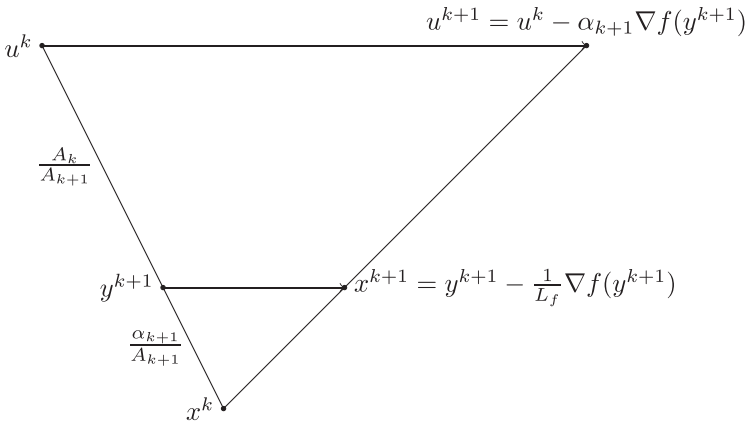

Fig. 1. Illustration of the three sequences of the A-BPGM in the unconstrained case $X = \mathbb{R}^n$, $r = 0$, $h = \frac{1}{2}\|x\|_2^2$. In this simple case it is easy to see that $u^{k+1} = u^k - \alpha_{k+1}\nabla f(y^{k+1})$, and the sequence u^k accumulates the previous gradient, while helping to keep momentum. Also by the similarity of the triangles, $x^{k+1} = y^{k+1} - \alpha_{k+1}\nabla f(y^{k+1}) \cdot \frac{\alpha_{k+1}}{A_{k+1}} = y^{k+1} - \frac{1}{L_f}\nabla f(y^{k+1})$, i.e. y^k is the sequence obtained by gradient descent steps. Finally, the sequence x^k is a convex combination of the momentum step and the gradient step. The illustration is inspired by personal communication with Yu. Nesterov on the Method of Similar Triangles (Gasnikov and Nesterov, 2018; Nesterov, 2018b).

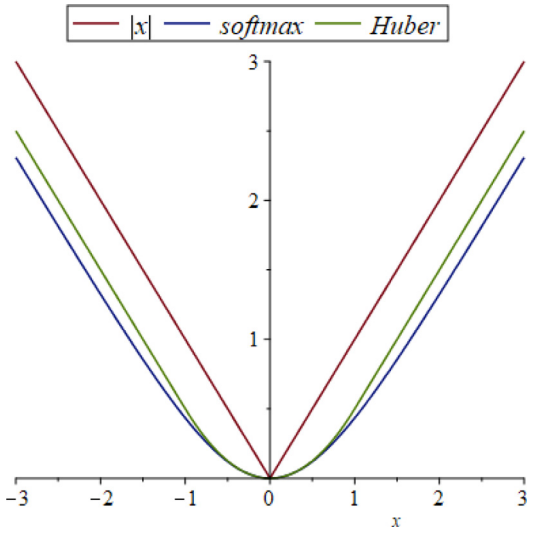

Fig. 2. Absolute value function $|x|$, its softmax smoothing and Huber smoothing, both with $\tau = 1$.

Example 6.2 (. ℓ_1-fit) In compressed sensing Candes and Tao (2007); Candes et al. (2006); Donoho (2006) one encounters the problem to minimize the ℓ_1 norm of the residual vector $\mathbf{A}x - b$ over a given closed convex set X. While it is well-known that this problem can in principle again be reformulated as an LP, the typical high-dimensionality of such problems makes this direct approach often not practicable. Adopting the smoothing technology, it is natural to choose $W = \{w \in \mathbb{R}^m | \|w\|_\infty \le 1\}$ and $h_w(w) = \frac{1}{2}\sum_{i=1}^m \|a_i\|_{E,*} w_i^2$, which gives

$$\Psi_\tau(x) = \max_{w \in W}\{\langle \mathbf{A}x - b, w\rangle - \tau h_w(w)\} = \sum_{i=1}^m \|a_i\|_{x,*}\psi_\tau\left(\frac{|\langle a_i, x\rangle - b_i|}{\|a_i\|_{E,*}}\right),$$

where $\psi_\tau(t)$ is the Huber function equal to $t^2/(2\tau)$ for $0 \le t \le \tau$ and $t - \tau/2$ if $t \ge \tau$.

For the particular case of smoothing the absolute value function $|x|$, Figure 2 gives the plot of the original function, its softmax smoothing and Huber smoothing, both with $\tau = 1$. Potentially, other ways of smoothing a non-smooth function can be applied, see Beck and Teboulle (2012) for a general framework.

Closing Remarks Let us make several remarks on the related literature. A close approach is proposed in Nemirovski (2004), where the problem (6.15) is considered directly as a min-max saddle-point problem. These classes of equilibrium problems are typically solved via tools from monotone variational inequalities, whose performance is typically worse than the performance of optimization algorithms. In particular, contrasting the above rate estimate with the one reported in Nemirovski (2004), one observes that the bound in Nemirovski (2004) has a similar to (6.20) structure, yet with the second term being non-accelerated, i.e. proportional to $1/N$. This approach was generalized to obtain an accelerated method for a special class of variational inequalities in Chen et al. (2017), where an optimal iteration complexity $O(L/\sqrt{\varepsilon})$ to reach an ε-close solution is reported. In the original paper (Nesterov, 2005b), the smoothing parameter is fixed and requires to know the parameters of the problem in advance. This has been improved in Nesterov (2005a), where an adaptive version of the smoothing techniques is proposed. This framework was extended in Alacaoglu et al. (2017); Tran-Dinh et al. (2020); Tran-Dinh and Cevher (2014); Tran-Dinh et al. (2018) for structured composite optimization problems in the form (2.7) and a related primal-dual representation (2.8). A related line of works studies minimization of strongly convex functions under linear constraints. Similarly to (6.16) the objective in the Lagrange dual problem has Lipschitz gradient, yet the challenge is that the feasible set in the dual problem is not bounded. Despite that it is possible to obtain accelerated primal-dual methods (Anikin et al., 2017; Chernov et al., 2016; Dvurechensky et al., 2016; 2018b; Guminov et al., 2021; 2019; Ivanova et al., 2020; Kroshnin et al., 2019; Nesterov et al., 2020; Tran-Dinh and Cevher, 2014; Tran-Dinh et al., 2018). In particular, this allows to obtain improved complexity bounds for different types of optimal transport problems (Dvurechensky et al., 2018a; 2018b; Guminov et al., 2021; Kroshnin et al., 2019; Lin et al., 2020; 2019; 2019; Tupitsa et al., 2020; Uribe et al., 2018).

6.4. Universal Accelerated Method

As it was discussed in the previous subsection, there is a gap in the convergence rate between the class of non-smooth convex optimization problems and the class of smooth convex optimization problems. In this subsection, we present a unifying framework (Nesterov, 2015) for these two classes which allows to obtain uniformly optimal complexity bounds for both classes by a single method without the need to know whether the objective is smooth or non-smooth.

Consider the Problem (P) with f which belongs to the class of functions with Hölder-continuous subgradients, i.e. for some $L_\nu > 0$ and $\nu \in [0, 1]$ it holds that $\|f'(x) - f'(y)\|_* \le L_\nu \|x - y\|^\nu$ for all $x, y \in \mathrm{dom}\, f$ and all $f'(x) \in \partial f(x)$ and $f'(y) \in \partial f(y)$. If $\nu = 1$, we recover the L_f-smoothness condition (2.1). If $\nu = 0$ we have that f has bounded variation of the subgradient, which is essentially equivalent to the bounded subgradient Assumption 8. The main observation (Devolder et al., 2014; Nesterov, 2015) is that this Hölder condition allows to prove an inexact version of the "descent Lemma" inequality (3.15). More precisely Nesterov (2015, Lemma 2), for any $x, y \in \mathrm{dom}\, f$ and any $\delta > 0$,

$$f(y) \le f(x) + \langle f'(x), y - x\rangle + \frac{L_\nu}{1+\nu}\|y - x\|^{1+\nu} \le f(x) + \langle f'(x), y - x\rangle + \frac{L}{2}\|y - x\|^2 + \delta,$$

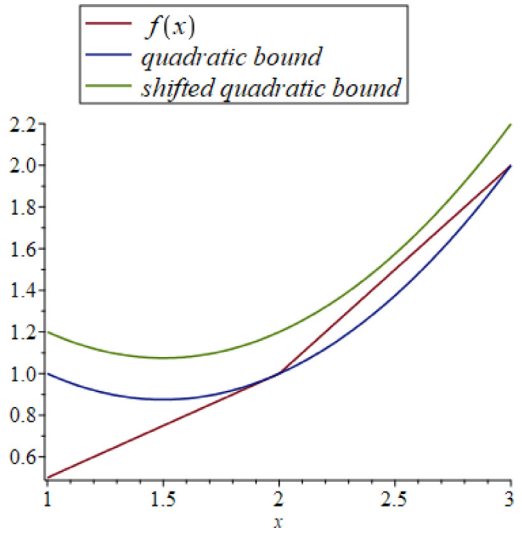

Fig. 3. Non-smooth function $f(x) = \max\{x - 1, x/2\}$, a quadratic function constructed using the first-order information at the point $x = 2$, and a shifted quadratic function constructed using the first-order information at the point $x = 2$. As one can see, adding a shift allows to obtain an upper quadratic bound for the objective, which is then minimized to obtain a new test point.

(6.21)

where

$$L \geq L(\delta) := \left(\frac{1-\nu}{1+\nu}\frac{1}{\delta}\right)^{\frac{1-\nu}{1+\nu}} L_\nu^{\frac{2}{1+\nu}} \tag{6.22}$$

with the convention that $0^0 = 1$. We illustrate this by Figure 3 where we plot a quadratic bound in the r.h.s. of (6.21) with $\delta = 0$ and a shifted quadratic bound in the r.h.s. of (6.21) with some $\delta > 0$. The first quadratic bound can not be an upper bound for $f(y)$ for any $L > 0$, and the positive shift allows to construct an upper bound. Thus, it is sufficient to equip the A-BPGM with a backtracking line-search to obtain a universal method. The resulting algorithm is listed below as Universal Accelerated Bregman Proximal Gradient Method (U-A-BPGM). A key step is the potentially non-monotone adjustment of the local Lipschitz gradient estimate L_k.

The Universal Accelerated Bregman Proximal Gradient Method (U-A-BPGM)

Input: Pick $x^0 = u^0 = y^0 \in \mathrm{dom}(r) \cap \mathsf{X}^\circ$, $\varepsilon > 0$, $0 < L_0 < L(\varepsilon/2)$, set $A_0 = 0$

General step: For $k = 0, 1, \ldots$ do:

Find the smallest integer $i_k \geq 0$ such that if one defines α_{k+1} from the quadratic equation $A_k + \alpha_{k+1} = 2^{i_k - 1} L_k \alpha_{k+1}^2$, sets $A_{k+1} = A_k + \alpha_{k+1}$,

sets $y^{k+1} = \frac{\alpha_{k+1}}{A_{k+1}} u^k + \frac{A_k}{A_{k+1}} x^k$,

sets $u^{k+1} = \mathrm{argmin}_{x \in \mathsf{X}} \left\{\alpha_{k+1}\left(f(y^{k+1}) + \langle f'(y^{k+1}), x - y^{k+1}\rangle + r(x)\right) + D_h(x, u^k)\right\}$,

sets $x^{k+1} = \frac{\alpha_{k+1}}{A_{k+1}} u^{k+1} + \frac{A_k}{A_{k+1}} x^k$,

then it holds that $f(x^{k+1}) \leq f(y^{k+1}) + \langle f'(y^{k+1}), x^{k+1} - y^{k+1}\rangle + \frac{2^{i_k-1}L_k}{2}\|x^{k+1} - y^{k+1}\|^2 + \frac{\varepsilon\alpha_{k+1}}{2A_{k+1}}$.

Set $L_{k+1} = 2^{i_k - 1} L_k$ and go to the next iterate k.

We first observe that for sufficiently large i_k, $2^{i_k-1}L_k \geq L\left(\frac{\varepsilon\alpha_{k+1}}{2A_{k+1}}\right)$, see Nesterov (2015, p.396). This means that the process of finding i_k is finite since the condition which is checked for each i_k is essentially (6.21) with $\delta = \frac{\varepsilon\alpha_{k+1}}{2A_{k+1}}$. The proof of convergence of (U-A-BPGM) follows essentially the same steps as the derivation of the convergence rate for A-BPGM in Section 6.1. The first thing which is changed is equation (6.1), where now the inexact descent Lemma is used instead of the exact one. The only difference is that L_f is changed to its local approximation L_{k+1} and add the error term $\frac{\varepsilon\alpha_{k+1}}{2A_{k+1}}$ appears in the r.h.s. In (6.2) the new quadratic equation with L_{k+1} is used and the inequality remains the same. This eventually leads to (6.5) with the only change being an additive error term $\frac{\varepsilon\alpha_{k+1}}{2A_{k+1}}$ in the r.h.s. Finally, this leads to the bound

$$\Psi(x^N) - \Psi_{\min}(\mathsf{X}) \leq \frac{D_h(u^*, u^0)}{A_N} + \frac{\varepsilon}{2}.$$

After some algebraic manipulation, Nesterov (2015, p.397) obtains an inequality $A_N \geq \frac{N^{\frac{1+3\nu}{1+\nu}} \varepsilon^{\frac{1-\nu}{1+\nu}}}{2^{\frac{2+4\nu}{1+\nu}} L_\nu^{\frac{2}{1+\nu}}}$. Substituting, we obtain

$$\Psi(x^N) - \Psi_{\min}(\mathsf{X}) \leq \frac{2^{\frac{2+4\nu}{1+\nu}} D_h(u^*, u^0) L_\nu^{\frac{2}{1+\nu}}}{N^{\frac{1+3\nu}{1+\nu}} \varepsilon^{\frac{1-\nu}{1+\nu}}} + \frac{\varepsilon}{2}.$$

Since the method does not require to know ν and L_ν, the iteration complexity to achieve accuracy ε is

$$N = O\left(\inf_{\nu \in [0,1]} \left(\frac{L_\nu}{\varepsilon}\right)^{\frac{2}{1+3\nu}} \left(D_h(u^*, u^0)\right)^{\frac{1+\nu}{1+3\nu}}\right).$$

It is easy to see that the oracle complexity, i.e. the number of proximal operations, is approximately the same. Indeed, the number of oracle calls for each k is $2(i_k + 1)$. Further, $L_{k+1} = 2^{i_k - 1} L_k$, which means that the total number of the oracle calls up to iteration N is $\sum_{k=0}^{N-1} 2(i_k + 1) = \sum_{k=0}^{N-1} 2(2\log_2 \frac{L_{k+1}}{L_k}) = 4N + 2\log_2 \frac{L_N}{L_0}$, i.e. is, up to a logarithmic term, four times larger than N. The obtained oracle complexity coincides up to a constant factor with the lower bound (Nemirovski and Yudin, 1983) for first-order methods applied to minimization of functions with Hölder-continuous gradients. In the particular case $\nu = 0$, we obtain the complexity $O\left(\frac{L_0^2 D_h(u^*, u^0)}{\varepsilon^2}\right)$, which corresponds to the convergence rate $1/\sqrt{k}$, which is typical for general non-smooth minimization. In the opposite case of smooth minimization corresponding to $\nu = 1$, we obtain the complexity $O\left(\sqrt{\frac{L_1 D_h(u^*, u^0)}{\varepsilon}}\right)$, which corresponds to the optimal convergence rate $1/k^2$. The same idea can be used to obtain universal version of BPGM (Nesterov, 2015). One can also use the strong convexity assumption to obtain faster convergence rate of U-A-BPGM either by restarts (Kamzolov et al., 2020; Roulet and d'Aspremont, 2017), or by incorporating the strong convexity parameter in the steps (Stonyakin et al., 2020). The same backtracking line-search can be applied in a much simpler way if one knows that f is L_f-smooth with some unknown Lipschitz constant or to achieve acceleration in practice caused by a pessimistic estimate for L_f (Dvinskikh et al., 2020; Dvurechensky et al., 2016; 2018b; Malitsky and Pock, 2018; Nesterov, 2013; Tran-Dinh et al., 2018). The idea is to use the standard exact "descent Lemma" inequality in each step of the accelerated method.

The idea of universal methods turned out to be very productive and several extensions have been proposed in the literature, including universal primal-dual methods (Baimurzina et al., 2019), universal method for convex and non-convex optimization (Ghadimi et al., 2019), and a universal primal-dual hybrid of accelerated gradient method with conjugate gradient method using additional one-dimensional minimization (Nesterov et al., 2020). The above-described method is not the only way to obtain adaptive and universal methods for smooth and non-smooth optimization problems. An alternative way which uses the norm of the current (sub)gradient to define the step-size was initiated probably by Polyak (1987) and became very popular in stochastic optimization for machine learning after the paper Duchi et al., 2011. On this avenue it was possible to obtain for $\nu \in \{0, 1\}$ universal accelerated optimization method (Levy et al., 2018) and universal methods for variational inequalities and saddle-point problems (Bach and Levy, 2019).

6.5. Connection between Accelerated method and Conditional Gradient

In this subsection we describe how a variant of conditional gradient method can be obtained as a particular case of A-BPGM with inexact Bregman Proximal step. We assume that f is L_f-smooth and for simplicity choose h to be the squared Euclidean norm $h(x) = \frac{1}{2}\|x\|^2$. Since we consider a conditional gradient method, it is natural to assume that the set X is bounded with $\max_{x,u\in\mathsf{X}} D_h(x,u) \leq \frac{\Omega^2}{2} \equiv \frac{1}{2}\Omega^2_{\frac{1}{2}\|\cdot\|^2}(\mathsf{X})$. We follow the idea of Ben-Tal and Nemirovski, 2020, where the main observation of is that the Prox-Mapping in A-BPGM can be calculated inexactly by applying the generalized linear oracle given in Definition 5.2. The idea is very similar to the conditional gradient sliding described in Section 5.3.4 with the difference that here we implement an approximate Bregman Proximal step using only *one* step of the generalized conditional gradient method. The resulting algorithm is listed below with the only difference with A-BPGM being the change of the Bregman Proximal step $u^{k+1} = \mathcal{P}_{\alpha_{k+1}r}(u^k, \alpha_{k+1}\nabla f(y^{k+1}))$ to the step $u^{k+1} = \mathcal{L}_{\mathsf{X},\alpha_{k+1}r}(\alpha_{k+1}\nabla f(y^{k+1}))$ given by generalized linear oracle.

Conditional Gradient Method by A-BPGM with Approximate Bregman Proximal Step

Input: pick $x^0 = u^0 = y^0 \in \mathrm{dom}(r) \cap \mathsf{X}^\circ$, set $A_0 = 0$

General step: For $k = 0, 1, \ldots$ do:

Find α_{k+1} from quadratic equation $A_k + \alpha_{k+1} = L_f\alpha^2_{k+1}$. Set $A_{k+1} = A_k + \alpha_{k+1}$.

Set $y^{k+1} = \frac{\alpha_{k+1}}{A_{k+1}}u^k + \frac{A_k}{A_{k+1}}x^k$.

Set (Approximate Bregman proximal step by generalized linear oracle)

$u^{k+1} = \mathrm{argmin}_{x\in\mathsf{X}}\left\{\alpha_{k+1}\left(f(y^{k+1}) + \langle\nabla f(y^{k+1}), x - y^{k+1}\rangle + r(x)\right)\right\} = \mathcal{L}_{\mathsf{X},\alpha_{k+1}r}(\alpha_{k+1}\nabla f(y^{k+1}))$.

Set $x^{k+1} = \frac{\alpha_{k+1}}{A_{k+1}}u^{k+1} + \frac{A_k}{A_{k+1}}x^k$.

Since the difference between such conditional gradient method and A-BPGM is in one simple change of the step for u^{k+1}, to obtain the convergence rate of the former, it is sufficient to track, what changes such approximate Bregman Proximal step entails in the convergence rate proof for A-BPGM. In other words, we need to understand what happens with the proof for A-BPGM if the Bregman Proximal step is made inexactly by applying the generalized linear oracle. The first important difference is that we need an inexact version of inequality (3.12), which was used in the convergence proof of A-BPGM and which the result of the exact Bregman Proximal step. To obtain its inexact version, let us denote

$$\varphi(x) = \alpha_{k+1}\left(f(y^{k+1}) + \langle\nabla f(y^{k+1}), x - y^{k+1}\rangle + r(x)\right).$$

Then generalized linear oracle actually minimizes this function on the set X to obtain u^{k+1}. Thus, by the optimality condition, we have that there exists $\xi \in \partial\varphi(u^{k+1})$ such that $\langle\xi, u^{k+1} - x\rangle \leq 0$ for all $x \in \mathsf{X}$. Now we remind that the Bregman Proximal step in A-BPGM minimizes $\varphi(x) + D_h(x, u^k)$. These observations allow to estimate the inexactness of the Bregman Proximal step implemented via generalized linear oracle. Indeed, for $u^{k+1} = \mathcal{L}_{\mathsf{X},\alpha_{k+1}r}(\alpha_{k+1}\nabla f(y^{k+1}))$

$$\begin{aligned}\langle\xi + \nabla h(u^{k+1}) - \nabla h(u^k), u^{k+1} - x\rangle &\leq \langle\nabla h(u^{k+1}) - \nabla h(u^k), u^{k+1} - x\rangle \\ &= -D_h(x,u^k) + D_h(x,u^{k+1}) + D_h(u^{k+1},u^k) \leq \Omega^2, \quad (6.23)\end{aligned}$$

where we used three-point identity in Lemma 3.3 and that $\max_{x,u\in\mathsf{X}} D_h(x,u) \leq \frac{\Omega^2}{2}$. This inequality provides inexact version of the optimality condition (3.10) in the problem $\min_{x\in\mathsf{X}}\{\varphi(x) + D_h(x,u^k)\}$, i.e. (3.13) with $\Delta = \Omega^2$. This in order leads to (3.14) with $\Delta = \Omega^2$, which is the desired inexact version of (3.12).

Let us now see, how this affects the convergence rate proof of A-BPGM. Inequality (3.12) was used in the analysis only in (6.5). This means that the change of (3.12) to (3.14) with $\Delta = \Omega^2$ leads to an additive term $\frac{\Omega^2}{A_{k+1}}$ in the r.h.s. of (6.5):

$$\Psi(x^{k+1}) \leq \frac{A_k}{A_{k+1}}\Psi(x^k) + \frac{\alpha_{k+1}}{A_{k+1}}\Psi(u) + \frac{1}{A_{k+1}}D_h(u,u^k) - \frac{1}{A_{k+1}}D_h(u,u^{k+1}) + \frac{\Omega^2}{A_{k+1}}, \quad u \in \mathsf{X}. \quad (6.24)$$

Multiplying both sides of the last inequality by A_{k+1}, summing these inequalities from $k = 0$ to $k = N-1$, and using that $A_N - A_0 = \sum_{k=0}^{N-1}\alpha_{k+1}$, we obtain

$$A_N\Psi(x^N) \leq A_0\Psi(x^0) + (A_N - A_0)\Psi(u) + D_h(u,u^0) - D_h(u,u^N) + N\Omega^2. \quad (6.25)$$

Since $A_0 = 0$, we can choose $u = x^* \in \mathrm{argmin}\{D_h(u,u^0)|u \in \mathsf{X}^*\}$, so that, for all $N \geq 1$,

$$\Psi(x^N) - \Psi_{\min}(\mathsf{X}) \leq \frac{D_h(x^*,u^0)}{A_N} + \frac{N\Omega^2}{A_N} \leq \frac{\Omega^2}{2A_N} + \frac{N\Omega^2}{A_N},$$

which, given the lower bound $A_N \geq \frac{(N+1)^2}{4L_f}$ leads to the final result for the convergence rate of this inexact A-BPGM implemented via generalized linear oracle:

$$\Psi(x^N) - \Psi_{\min}(\mathsf{X}) \leq \frac{2L_f\Omega^2}{(N+1)^2} + \frac{4L_f\Omega^2}{N+1}.$$

Thus, we obtain a variant of conditional gradient method with the same convergence rate $1/N$ as for the standard conditional gradient method. Using the same approach, but with U-A-BPGM as the basis method, one can obtain a universal version of conditional gradient method (Stonyakin et al., 2020) for minimizing objectives with Hölder-continuous gradient. The bounds in this case a similar to the ones obtained in a more direct universal method in Nesterov (2018a). Similar bounds were also recently obtained in Zhao and Freund (2020).

7. Conclusion

We close this survey, with a very important fact which Nesterov writes in the introduction of his important textbook (Nesterov, 2018b): *in general, optimization problems are unsolvable.* Convex programming stands out from this general fact, since it describes a significantly large class of model problems, with important practical applications, for which general solution techniques have been developed within the mathematical framework of interior-point techniques. However, modern optimization problems are large-scale in nature, which renders these polynomial time methods impractical. First-order methods have become the gold standard in balancing cheap iterations with low solution accuracy, and many theoretical and practical advances having been made in the last 20 years.

Despite the fact that convex optimization is approaching the state of being a primitive similar to linear algebra techniques, we foresee that the development of first-order methods has not come to a halt yet. In connection with stochastic inputs, the combination of acceleration techniques with other performance boosting tricks, like variance reduction, incremental techniques, as well as distributed optimization, still promises to produce some new innovations. On the other hand, there is also still much room for improvement of algorithms for optimization problems which do not admit a prox-friendly geometry. Distributed optimization, in particular in the context of federated learning is now a very active area of research, see Kairouz et al. (2021) for a recent review of federated learning and (Gorbunov et al., d) for a recent review of distributed optimization. Another important focus in the research in optimization methods is now on numerical methods for non-convex optimization motivated by training of deep neural networks, see Danilova et al. (2020); Sun (2019) for a recent review. A number of open questions remain in the theory of first-order methods for variational inequalities and saddle-point problems, mainly in the case of variational inequalities with non-monotone operators. In particular, recently the

authors of (Cohen et al., 2021) observed a connection between extra-gradient methods for monotone variational inequalities and accelerated first-order methods. Thus, as we emphasize in this survey, new connections, that are still continuously being discovered between different methods and different formulations, can lead to new understanding and developments in this lively field of first-order methods.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors are grateful to Yurii Nesterov and Alexander Gasnikov for fruitful discussions. Mathias Staudigl thanks the COST Action CA16228 (European Network for Game Theory), the FMJH Program PGMO and from the support of EDF (Project "Privacy preserving algorithms for distributed control of energy markets") for its support. The research of Pavel Dvurechensky is supported by the Ministry of Science and Higher Education of the Russian Federation (Goszadaniye) No.075-00337-20-03, project No. 0714-2020-0005.

References

Alacaoglu, A., Tran Dinh, Q., Fercoq, O., Cevher, V., 2017. Smooth primal-dual coordinate descent algorithms for nonsmooth convex optimization. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Vol. 30. Curran Associates, Inc., pp. 5852–5861.

Allen-Zhu, Z., 2017. Katyusha: The first direct acceleration of stochastic gradient methods. In: Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017. ACM, New York, NY, USA, pp. 1200–1205.

Allen-Zhu, Z., Hazan, E., 2016. Optimal black-box reductions between optimization objectives. In: Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Vol. 29. Curran Associates, Inc., pp. 1614–1622.

Alvarez, F., Bolte, J., Brahic, O., 2004. Hessian Riemannian gradient flows in convex programming. SIAM Journal on Control and Optimization 43 (2), 477–501.

Andersen, E.D., Andersen, K.D., 2000. The Mosek Interior Point Optimizer for Linear Programming: An Implementation of the Homogeneous Algorithm. Springer US, Boston, MA, pp. 197–232.

Anikin, A.S., Gasnikov, A.V., Dvurechensky, P.E., Tyurin, A.I., Chernov, A.V., 2017. Dual approaches to the minimization of strongly convex functionals with a simple structure under affine constraints. Computational Mathematics and Mathematical Physics 57 (8), 1262–1276.

Arrow, K., Hurwicz, L., Uzawa, H., 1958. Studies in linear and non-linear programming. In: Chenery, H., Johnson, S., Karlin, S., Marschak, T., Solow, R. (Eds.), Stanford Mathematical Studies in the Social Sciences, Vol. II. Stanford University Press, Stanford.

Attouch, H., Bolte, J., Redont, P., Teboulle, M., 2004. Singular Riemannian barrier methods and gradient-projection dynamical systems for constrained optimization. Optimization 53 (5-6), 435–454.

Attouch, H., Cabot, A., Frankel, P., Peypouquet, J., 2011. Alternating proximal algorithms for linearly constrained variational inequalities: application to domain decomposition for pde's. Nonlinear Analysis: Theory, Methods & Applications 74 (18), 7455–7473.

Attouch, H., Chbani, Z., Fadili, J., Riahi, H., 2020. First-order optimization algorithms via inertial systems with hessian driven damping. Mathematical Programming doi:10.1007/s10107-020-01591-1.

Attouch, H., Chbani, Z., Peypouquet, J., Redont, P., 2018. Fast convergence of inertial dynamics and algorithms with asymptotic vanishing viscosity. Mathematical Programming 168 (1-2), 123–175.

Attouch, H., Redont, P., Soubeyran, A., 2007. A new class of alternating proximal minimization algorithms with costs-to-move. SIAM Journal on Optimization 18 (3), 1061–1081.

Attouch, H., Teboulle, M., 2004. Regularized lotka-volterra dynamical system as continuous proximal-like method in optimization. Journal of Optimization Theory and Applications 121 (3), 541–570.

Auslender, A., Teboulle, M., 2006. Asymptotic cones and functions in optimization and variational inequalities. Springer Science & Business Media.

Auslender, A., Teboulle, M., 2006. Interior gradient and proximal methods for convex and conic optimization. SIAM Journal on Optimization 16 (3), 697–725.

Auslender, A., Teboulle, M., 2009. Projected subgradient methods with non-euclidean distances for non-differentiable convex minimization and variational inequalities. Mathematical Programming 120 (1), 27–48.

Bach, F., 2015. Duality between subgradient and conditional gradient methods. SIAM Journal on Optimization 25 (1), 115–129.

Bach, F., Levy, K.Y., 2019. A universal algorithm for variational inequalities adaptive to smoothness and noise. In: Beygelzimer, A., Hsu, D. (Eds.), Proceedings of the Thirty-Second Conference on Learning Theory, Vol. 99 of Proceedings of Machine Learning Research, PMLR, Phoenix, USA, pp. 164–194.

Baes, M., 2009. Estimate sequence methods: extensions and approximations, Institute for Operations Research. ETH, Zürich, Switzerland.

Bah, B., Rauhut, H., Terstiege, U., Westdickenberg, M., 2019. Learning deep linear neural networks: Riemannian gradient flows and convergence to global minimizers. Information and Inference: A Journal of the IMAIaaa039.

Baimurzina, D.R., Gasnikov, A.V., Gasnikova, E.V., Dvurechensky, P.E., Ershov, E.I., Kubentaeva, M.B., Lagunovskaya, A.A., 2019. Universal method of searching for equilibria and stochastic equilibria in transportation networks. Computational Mathematics and Mathematical Physics 59 (1), 19–33.

Banert, S., BoŢ, R.I., Csetnek, E.R., 2021. Fixing and extending some recent results on the admm algorithm. Numerical Algorithms 86 (3), 1303–1325.

Bauschke, H., Borwein, J., Combettes, P., 2003. Bregman monotone optimization algorithms. SIAM Journal on Control and Optimization 42 (2), 596–636.

Bauschke, H.H., Bolte, J., Teboulle, M., 2016. A descent lemma beyond lipschitz gradient continuity: First-order methods revisited and applications. Mathematics of Operations Research 42 (2), 330–348.

Bauschke, H.H., Combettes, P.L., 2016. Convex Analysis and Monotone Operator Theory in Hilbert Spaces. Springer - CMS Books in Mathematics.

Bayandina, A., Dvurechensky, P., Gasnikov, A., Stonyakin, F., Titov, A., 2018. Mirror descent and convex optimization problems with non-smooth inequality constraints. In: Giselsson, P., Rantzer, A. (Eds.), Large-Scale and Distributed Optimization. Springer International Publishing, pp. 181–215. Ch. 8

Beck, A., 2017. First-Order Methods in Optimization. Society for Industrial and Applied Mathematics.

Beck, A., Guttmann-Beck, N., 2019. FOM –a matlab toolbox of first-order methods for solving convex optimization problems. Optimization Methods and Software 34 (1), 172–193.

Beck, A., Shtern, S., 2017. Linearly convergent away-step conditional gradient for non-strongly convex functions. Mathematical Programming 164 (1-2), 1–27.

Beck, A., Teboulle, M., 2003. Mirror descent and nonlinear projected subgradient methods for convex optimization. Operations Research Letters 31 (3), 167–175.

Beck, A., Teboulle, M., 2004. A conditional gradient method with linear rate of convergence for solving convex linear systems. Mathematical Methods of Operations Research 59 (2), 235–247.

Beck, A., Teboulle, M., 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences 2 (1), 183–202.

Beck, A., Teboulle, M., 2009. Gradient-based algorithms with applications to signal recovery. In: Palomar, D.P., Eldar, Y.C. (Eds.), Convex optimization in signal processing and communications. Cambridge University Press, pp. 42–88.

Beck, A., Teboulle, M., 2012. Smoothing and first order methods: A unified framework. SIAM Journal on Optimization 22 (2), 557–580.

Ben-Tal, A., Margalit, T., Nemirovski, A., 2001. The ordered subsets mirror descent optimization method with applications to tomography. SIAM Journal on Optimization 12 (1), 79–108.

Ben-Tal, A., Nemirovski, A., 2020. Lectures on Modern Convex Optimization (Lecture Notes). Personal web-page of A. Nemirovski. URL https://www2.isye.gatech.edu/~nemirovs/LMCOLN2020WithSol.pdf

Benaïm, M., 1998. Recursive algorithms, urn processes, and the chaining number of chain recurrent sets. Ergodic Theory and Dynamical Systems 18, 53–87.

Benveniste, A., Métivier, M., Priouret, P., 1990. Adaptive Algorithms and Stochastic Approximations. Springer, Berlin.

Bian, W., Chen, X., 2015. Linearly constrained non-lipschitz optimization for image restoration. SIAM Journal on Imaging Sciences 8 (4), 2294–2322.

Bian, W., Chen, X., Ye, Y., 2015. Complexity analysis of interior point algorithms for non-lipschitz and nonconvex minimization. Mathematical Programming 149 (1), 301–327.

Bolte, J., Nguyen, T.P., Peypouquet, J., Suter, B.W., 2017. From error bounds to the complexity of first-order descent methods for convex functions. Mathematical Programming 165 (2), 471–507.

Bolte, J., Teboulle, M., 2003. Barrier operators and associated gradient-like dynamical systems for constrained minimization problems. SIAM Journal on Control and Optimization 42 (4), 1266–1292.

Bomze, I.M., Mertikopoulos, P., Schachinger, W., Staudigl, M., 2019. Hessian barrier algorithms for linearly constrained optimization problems. SIAM Journal on Optimization 29 (3), 2100–2127.

Bottou, L., Curtis, F., Nocedal, J., 2018. Optimization methods for large-scale machine learning. SIAM Review 60 (2), 223–311.

Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine learning 3 (1), 1–122.

Boyd, S., Vandenberghe, L., 2004. Convex optimization. Cambridge university press.

Bruckstein, A., Donoho, D., Elad, M., 2009. From sparse solutions of systems of equations to sparse modeling of signals and images. SIAM Review 51 (1), 34–81.

Bubeck, S., 2015. Convex optimization: Algorithms and complexity. Foundations and Trends in Machine Learning 8 (3-4), 231–357.

Bühlmann, P., van de Geer, S., 2011. Statistics for high-dimensional data. Springer Series in Statistics. Springer-Verlag Berlin Heidelberg.

Bùi, M.N., Combettes, P.L., 2021. Bregman forward-backward operator splitting. Set-Valued and Variational Analysis 29 (3), 583–603.

Candes, E., Tao, T., 2007. The Dantzig selector: Statistical estimation when p is much larger than n. The Annals of Statistics 35 (6), 2313–2351.

Candes, E.J., Romberg, J., Tao, T., 2006. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Transactions on Information Theory 52 (2), 489–509.

Canon, M.D., Cullum, C.D., 1968. A tight upper bound on the rate of convergence of frank-wolfe algorithm. SIAM Journal on Control 6 (4), 509–516.

Carderera, A., Besancon, M., Pokutta, S.. Simple steps are all you need: Frank-wolfe and generalized self-concordant functions.

Carderera, A., Diakonikolas, J., Lin, C.Y., Pokutta, S., 2021. Parameter-free locally accelerated conditional gradients. In: Meila, M., Zhang, T. (Eds.), Proceedings of the 38th International Conference on Machine Learning, Vol. 139 of Proceedings of Machine Learning Research, PMLR, pp. 1283–1293.

Censor, Y., Zenios, S.A., 1992. Proximal minimization algorithm withd-functions. Journal of Optimization Theory and Applications 73 (3), 451–464.

Cesa-Bianchi, N., Lugosi, G., 2006. Prediction, Learning, and Games. Cambridge University Press.

Chambolle, A., Pock, T., 2011. A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision 40 (1), 120–145.

Chen, C., He, B., Ye, Y., Yuan, X., 2016. The direct extension of admm for multi-block convex minimization problems is not necessarily convergent. Mathematical Programming 155 (1-2), 57–79.

Chen, G., Teboulle, M., 1993. Convergence analysis of a proximal-like minimization algorithm using Bregman functions. SIAM Journal on Optimization 3 (3), 538–543.

Chen, Y., Lan, G., Ouyang, Y., 2017. Accelerated schemes for a class of variational inequalities. Mathematical Programming. 10.1007/s10107-017-1161-4

Chernov, A., Dvurechensky, P., Gasnikov, A., 2016. Fast primal-dual gradient method for strongly convex minimization problems with linear constraints. In: Kochetov, Y., Khachay, M., Beresnev, V., Nurminski, E., Pardalos, P. (Eds.), Discrete Optimization and Operations Research: 9th International Conference, DOOR 2016, Vladivostok, Russia, September 19-23, 2016, Proceedings. Springer International Publishing, pp. 391–403.

Cohen, M., Diakonikolas, J., Orecchia, L., 2018. On acceleration with noise-corrupted gradients. In: Dy, J., Krause, A. (Eds.), Proceedings of the 35th International Conference on Machine Learning, Vol. 80 of Proceedings of Machine Learning Research, PMLR. Stockholmsmässan, Stockholm Sweden, pp. 1019–1028.

Cohen, M.B., Sidford, A., Tian, K., 2021. Relative lipschitzness in extragradient methods and a direct recipe for acceleration. In: Lee, J.R. (Ed.), 12th Innovations in Theoretical Computer Science Conference (ITCS 2021), pp. 62:1–62:18. doi:10.4230/LIPIcs.ITCS.2021.62. Vol. 185 of Leibniz International Proceedings in Informatics (LIPIcs), Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany

Combettes, P.L., Pesquet, J.C., 2011. Proximal splitting methods in signal processing. Springer, pp. 185–212.

Combettes, P.L., Wajs, V.R., 2005. Signal recovery by proximal forward-backward splitting. Multiscale Modeling & Simulation 4 (4), 1168–1200.

Cox, B., Juditsky, A., Nemirovski, A., 2014. Dual subgradient algorithms for large-scale nonsmooth learning problems. Mathematical Programming 148 (1), 143–180.

Damla Ahipasaoglu, S., Sun, P., Todd, M.J., 2008. Linear convergence of a modified frank–wolfe algorithm for computing minimum-volume enclosing ellipsoids. Optimisation Methods and Software 23 (1), 5–19.

Danilova, M., Dvurechensky, P., Gasnikov, A., Gorbunov, E., Guminov, S., Kamzolov, D., Shibaev, I., 2020. Recent theoretical advances in non-convex optimization. arXiv:2012.06188Accepted to be a part of Springer volume "High Dimensional Optimization and Probability".

d'Aspremont, A., 2008. Smooth optimization with approximate gradient. SIAM J. on Optimization 19 (3), 1171–1183.

d'Aspremont, A., Scieur, D., Taylor, A.. Acceleration methods.

Daubechies, I., Defrise, M., De Mol, C., 2004. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Communications on pure and applied mathematics 57 (11), 1413–1457.

Davis, D., Drusvyatskiy, D., Kakade, S., Lee, J.D., 2020. Stochastic subgradient method converges on tame functions. Foundations of Computational Mathematics 20 (1), 119–154.

Devolder, O., 2013. Exactness, inexactness and stochasticity in first-order methods for large-scale convex optimization Ph.d. thesis.

Devolder, O., Glineur, F., Nesterov, Y., 2014. First-order methods of smooth convex optimization with inexact oracle. Mathematical Programming 146 (1), 37–75.

Diakonikolas, J., Orecchia, L., 2018. Alternating randomized block coordinate descent. In: Dy, J., Krause, A. (Eds.), Proceedings of the 35th International Conference on Machine Learning, Vol. 80 of Proceedings of Machine Learning Research, PMLR, Stockholmsmässan, Stockholm Sweden, pp. 1224–1232.

Doljansky, M., Teboulle, M., 1998. An interior proximal algorithm and the exponential multiplier method for semidefinite programming. SIAM Journal on Optimization 9 (1), 1–13.

Donoho, D.L., 2006. Compressed sensing. IEEE Transactions on Information Theory 52 (4), 1289–1306.

Dragomir, R.-A., Taylor, A.B., d'Aspremont, A., Bolte, J., 2021. Optimal complexity and certification of bregman first-order methods. Mathematical Programming doi:10.1007/s10107-021-01618-1.

Duchi, J., Hazan, E., Singer, Y., 2011. Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research 12, 2121–2159. (Jul.)

Duchi, J., Shalev-Shwartz, S., Singer, Y., Tewari, A., 2010. Composite objective mirror descent. In: COLT 2010 - The 23rd Conference on Learning Theory, pp. 14–26.

Duchi, J.C., Ruan, F., 2018. Stochastic methods for composite and weakly convex optimization problems. SIAM Journal on Optimization 28 (4), 3229–3259.

Dunn, J.C., 1979. Rates of convergence for conditional gradient algorithms near singular and nonsingular extremals. SIAM Journal on Control and Optimization 17 (2), 187–211.

Dvinskikh, D., Gorbunov, E., Gasnikov, A., Dvurechensky, P., Uribe, C.A., 2019. On primal and dual approaches for distributed stochastic convex optimization over networks. In: 2019 IEEE 58th Conference on Decision and Control (CDC), pp. 7435–7440.

Dvinskikh, D., Ogaltsov, A., Gasnikov, A., Dvurechensky, P., Spokoiny, V., 2020. On the line-search gradient methods for stochastic optimization. IFAC-PapersOnLine 53 (2), 1715–1720.

Dvurechensky, P., Dvinskikh, D., Gasnikov, A., Uribe, C.A., Nedić, A., 2018. Decentralize and randomize: Faster algorithm for Wasserstein barycenters. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Vol. 31. Curran Associates, Inc., pp. 10783–10793. NeurIPS 2018

Dvurechensky, P., Gasnikov, A., 2016. Stochastic intermediate gradient method for convex problems with stochastic inexact oracle. Journal of Optimization Theory and Applications 171 (1), 121–145.

Dvurechensky, P., Gasnikov, A., Gasnikova, E., Matsievsky, S., Rodomanov, A., Usik, I., 2016. Primal-dual method for searching equilibrium in hierarchical congestion population games. In: Supplementary Proceedings of the 9th International Conference on Discrete Optimization and Operations Research and Scientific School (DOOR 2016) Vladivostok, Russia, September 19 - 23, 2016, pp. 584–595.

Dvurechensky, P., Gasnikov, A., Kroshnin, A., 2018. Computational optimal transport: Complexity by accelerated gradient descent is better than by Sinkhorn's algorithm. In: Dy, J., Krause, A. (Eds.), Proceedings of the 35th International Conference on Machine Learning, Vol. 80 of Proceedings of Machine Learning Research, pp. 1367–1376.

Dvurechensky, P., Gasnikov, A., Tiurin, A., Zholobov, V.. Unifying framework for accelerated randomized methods in convex optimization.

Dvurechensky, P., Gorbunov, E., Gasnikov, A., 2021. An accelerated directional derivative method for smooth stochastic convex optimization. European Journal of Operational Research 290 (2), 601–621.

Dvurechensky, P., Kamzolov, D., Lukashevich, A., Lee, S., Ordentlich, E., Uribe, C.A., Gasnikov, A.. Hyperfast second-order local solvers for efficient statistically preconditioned distributed optimization.

Dvurechensky, P., Ostroukhov, P., Safin, K., Shtern, S., Staudigl, M., 2020. Self-concordant analysis of frank-Wolfe algorithms. In: Daumé III, H., Singh, A. (Eds.), Proceedings of the 37th International Conference on Machine Learning, Vol. 119 of Proceedings of Machine Learning Research, PMLR, Virtual, pp. 2814–2824.

Dvurechensky, P., Safin, K., Shtern, S., Staudigl, M.. Generalized self-concordant analysis of Frank-Wolfe algorithms.

Dvurechensky, P.E., Gasnikov, A.V., Nurminski, E.A., Stonyakin, F.S., 2020. Advances in Low-Memory Subgradient Optimization. Springer International Publishing, Cham, pp. 19–59.

Eckstein, J., 1994. Some saddle-function splitting methods for convex programming. Optimization Methods and Software 4 (1), 75–83.

Eckstein, J., Bertsekas, D.P., 1992. On the douglas—rachford splitting method and the proximal point algorithm for maximal monotone operators. Mathematical Programming 55 (1-3), 293–318.

Epelman, M., Freund, R.M., 2000. Condition number complexity of an elementary algorithm for computing a reliable solution of a conic linear system. Mathematical Programming 88 (3), 451–485.

Facchinei, F., Pang, J.S., 2003. Finite-dimensional variational inequalities and complementarity problems - volume i and volume II. Springer Series in Operations Research.

Feizollahi, M.J., Ahmed, S., Sun, A., 2017. Exact augmented lagrangian duality for mixed integer linear programming. Mathematical Programming 161 (1), 365–387.

Fercoq, O., Qu, Z., 2020. Restarting the accelerated coordinate descent method with a rough strong convexity estimate. Computational Optimization and Applications 75 (1), 63–91.

Fercoq, O., Richtárik, P., 2015. Accelerated, parallel, and proximal coordinate descent. SIAM Journal on Optimization 25 (4), 1997–2023.

Frank, M., Wolfe, P., et al., 1956. An algorithm for quadratic programming. Naval research logistics quarterly 3 (1-2), 95–110.

Freund, R.M., Grigas, P., 2016. New analysis and results for the frank–wolfe method. Mathematical Programming 155 (1-2), 199–230.

Frostig, R., Ge, R., Kakade, S., Sidford, A., 2015. Un-regularizing: approximate proximal point and faster stochastic algorithms for empirical risk minimization. In: Bach, F., Blei, D. (Eds.), Proceedings of the 32nd International Conference on Machine Learning, Vol. 37 of Proceedings of Machine Learning Research, PMLR, Lille, France, pp. 2540–2548.

Gabay, D., 1983. Applications of the method of multipliers to variational inequalities, 15. Elsevier, pp. 299–331. Ch. ix

Garber, D., Hazan, E., 2015. Faster rates for the frank-wolfe method over strongly-convex sets. In: 32nd International Conference on Machine Learning, ICML 2015.

Garber, D., Hazan, E., 2016. A linearly convergent variant of the conditional gradient algorithm under strong convexity, with applications to online and stochastic optimization. SIAM Journal on Optimization 26 (3), 1493–1528.

Gasnikov, A., Dvurechensky, P., Gorbunov, E., Vorontsova, E., Selikhanovych, D., Uribe, C.A., Jiang, B., Wang, H., Zhang, S., Bubeck, S., Jiang, Q., Lee, Y.T., Li, Y., Sidford, A., 2019. Near optimal methods for minimizing convex functions with lipschitz p-th derivatives. In: Beygelzimer, A., Hsu, D. (Eds.), Proceedings of the Thirty-Second Conference on Learning Theory, Vol. 99 of Proceedings of Machine Learning Research, PMLR, Phoenix, USA, pp. 1392–1393.

Gasnikov, A.V., Dvurechensky, P.E., 2016. Stochastic intermediate gradient method for convex optimization problems. Doklady Mathematics 93 (2), 148–151.

Gasnikov, A.V., Nesterov, Y.E., 2018. Universal method for stochastic composite optimization problems. Computational Mathematics and Mathematical Physics 58 (1), 48–64.

Gasnikov, A.V., Tyurin, A.I., 2019. Fast gradient descent for convex minimization problems with an oracle producing a (δ, l)-model of function at the requested point. Computational Mathematics and Mathematical Physics 59 (7), 1085–1097.
Ghadimi, S., Lan, G., 2012. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework. SIAM Journal on Optimization 22 (4), 1469–1492.
Ghadimi, S., Lan, G., 2013. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, ii: Shrinking procedures and optimal algorithms. SIAM Journal on Optimization 23 (4), 2061–2089.
Ghadimi, S., Lan, G., Zhang, H., 2019. Generalized uniformly optimal methods for nonlinear programming. Journal of Scientific Computing 79 (3), 1854–1881.
Glowinski, R., Tallec, P.L., 1989. Augmented Lagrangian and operator-splitting methods in nonlinear mechanics. SIAM.
Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. In: Advances in neural information processing systems, pp. 2672–2680.
Gorbunov, E., Danilova, M., Gasnikov, A., 2020. Stochastic optimization with heavy-tailed noise via accelerated gradient clipping. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (Eds.), Advances in Neural Information Processing Systems, Vol. 33. Curran Associates, Inc., pp. 15042–15053.
Gorbunov, E., Danilova, M., Shibaev, I., Dvurechensky, P., Gasnikov, A.. Near-optimal high probability complexity bounds for non-smooth stochastic optimization with heavy–tailed noise.
Gorbunov, E., Dvinskikh, D., Gasnikov, A.. Optimal decentralized distributed algorithms for stochastic convex optimization.
Gorbunov, E., Dvurechensky, P., Gasnikov, A.. An accelerated method for derivative-free smooth stochastic convex optimization.
Gorbunov, E., Rogozin, A., Beznosikov, A., Dvinskikh, D., Gasnikov, A., d. Recent theoretical advances in decentralized distributed convex optimization. arXiv preprint arXiv:2011.13259Accepted to be a part of Springer volume "High Dimensional Optimization and Probability".
Gower, R.M., Schmidt, M., Bach, F., Richtárik, P., 2020. Variance-reduced methods for machine learning. In: Proceedings of the IEEE, Vol. 108, pp. 1968–1983.
Guélat, J., Marcotte, P., 1986. Some comments on wolfe's 'away step'. Mathematical Programming 35 (1), 110–119.
Guminov, S., Dvurechensky, P., Tupitsa, N., Gasnikov, A., 2021. On a combination of alternating minimization and Nesterov's momentum. In: Meila, M., Zhang, T. (Eds.), Proceedings of the 38th International Conference on Machine Learning, Vol. 139 of Proceedings of Machine Learning Research, PMLR, Virtual, pp. 3886–3898.
Guminov, S.V., Nesterov, Y.E., Dvurechensky, P.E., Gasnikov, A.V., 2019. Accelerated primal-dual gradient descent with linesearch for convex, nonconvex, and nonsmooth optimization problems. Doklady Mathematics 99 (2), 125–128.
Haeser, G., Liu, H., Ye, Y., 2018. Optimality condition and complexity analysis for linearly-constrained optimization without differentiability on the boundary. Mathematical Programming doi:10.1007/s10107-018-1290-4.
Hanzely, F., Richtárik, P., Xiao, L., 2021. Accelerated bregman proximal gradient methods for relatively smooth convex optimization. Computational Optimization and Applications 79 (2), 405–440.
Helmke, U., Moore, J.B., 1996. Optimization and dynamical systems. Communications & Control Engineering. Springer Berlin Heidelberg.
Hendrikx, H., Xiao, L., Bubeck, S., Bach, F., Massoulie, L., 2020. Statistically preconditioned accelerated gradient method for distributed optimization. In: Daumé III, H., Singh, A. (Eds.), Proceedings of the 37th International Conference on Machine Learning, Vol. 119 of Proceedings of Machine Learning Research, PMLR, pp. 4203–4227.
Hiriart-Urrut, J.-B., Lemaréchal, C., 2001. Fundamentals of Convex Analysis. Springer.
Holloway, C.A., 1974. An extension of the frank and wolfe method of feasible directions. Mathematical Programming 6 (1), 14–27.
Ivanova, A., Dvurechensky, P., Gasnikov, A., Kamzolov, D., 2020. Composite optimization for the resource allocation problem. Optimization Methods and Software 0 (0), 1–35. doi:10.1080/10556788.2020.1712599.
Jaggi, M., 2013. Revisiting frank-wolfe: Projection-free sparse convex optimization. International Conference on Machine Learning 427–435.
Jain, P., Kar, P., 2017. Non-convex optimization for machine learning. Found. Trends Mach. Learn. 10 (3-4), 142–336.
Juditsky, A., Kılınç Karzan, F., Nemirovski, A., 2013. Randomized first order algorithms with applications to ℓ_1-minimization. Mathematical Programming 142 (1), 269–310.
Juditsky, A., Nazin, A.V., Tsybakov, A.B., Vayatis, N., 2005. Recursive aggregation of estimators by the mirror descent algorithm with averaging. Problems of Information Transmission 41 (4), 368–384.
Juditsky, A., Nemirovski, A., 2011. First order methods for nonsmooth convex large-scale optimization, i: general purpose methods. In: Optimization for Machine Learning. MIT Press, pp. 121–148. Ch. 5
Juditsky, A., Nemirovski, A., 2011. First order methods for nonsmooth convex large-scale optimization, ii: utilizing problems structure. In: Optimization for Machine Learning. MIT Press, pp. 149–183. Ch. 6
Juditsky, A., Nesterov, Y., 2014. Deterministic and stochastic primal-dual subgradient algorithms for uniformly convex minimization. Stochastic Systems 4 (1), 44–80.
Kairouz, P., McMahan, H.B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A.N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., D'Oliveira, R.G.L., Eichner, H., Rouayheb, S.E., Evans, D., Gardner, J., Garrett, Z., Gascón, A., Ghazi, B., Gibbons, P.B., Gruteser, M., Harchaoui, Z., He, C., He, L., Huo, Z., Hutchinson, B., Hsu, J., Jaggi, M., Javidi, T., Joshi, G., Khodak, M., Konecný, J., Korolova, A., Koushanfar, F., Koyejo, S., Lepoint, T., Liu, Y., Mittal, P., Mohri, M., Nock, R., Özgür, A., Pagh, R., Qi, H., Ramage, D., Raskar, R., Raykova, M., Song, D., Song, W., Stich, S.U., Sun, Z., Suresh, A.T., Tramèr, F., Vepakomma, P., Wang, J., Xiong, L., Xu, Z., Yang, Q., Yu, F.X., Yu, H., Zhao, S., 2021. Advances and open problems in federated learning. Foundations and Trends®in Machine Learning 14 (1-2), 1–210. doi:10.1561/2200000083.
Kamzolov, D., Dvurechensky, P., Gasnikov, A.V., 2020. Universal intermediate gradient method for convex problems with inexact oracle. Optimization Methods and Software 0 (0), 1–28. doi:10.1080/10556788.2019.1711079.
Kerdreux, T., d'Aspremont, A., Pokutta, S., 2019. Restarting frank-wolfe. In: Chaudhuri, K., Sugiyama, M. (Eds.), Proceedings of Machine Learning Research, Vol. 89 of Proceedings of Machine Learning Research, PMLR, pp. 1275–1283.
Kroshnin, A., Tupitsa, N., Dvinskikh, D., Dvurechensky, P., Gasnikov, A., Uribe, C., 2019. On the complexity of approximating Wasserstein barycenters. In: Chaudhuri, K., Salakhutdinov, R. (Eds.), Proceedings of the 36th International Conference on Machine Learning, Vol. 97 of Proceedings of Machine Learning Research, PMLR, Long Beach, California, USA, pp. 3530–3540.
Kuczyński, J., Woźniakowski, H., 1992. Estimating the largest eigenvalue by the power and lanczos algorithms with a random start. SIAM Journal on Matrix Analysis and Applications 13 (4), 1094–1122.
Kushner, H.J., 1984. Approximation and Weak Convergence Methods for Random Processes. The MIT Press.
Lacoste-Julien, S., Jaggi, M., 2015. On the global linear convergence of frank-wolfe optimization variants. Advances in neural information processing systems 28, 496–504.
Lan, G.. The complexity of large-scale convex programming under a linear optimization oracle.
Lan, G., 2012. An optimal method for stochastic composite optimization. Mathematical Programming 133 (1), 365–397.
Lan, G., 2020. First-order and Stochastic Optimization Methods for Machine Learning. Springer.
Lan, G., Zhou, Y., 2017. An optimal randomized incremental gradient method. Mathematical Programming doi:10.1007/s10107-017-1173-0.
Lan, G., Zhou, Y., 2016. Conditional gradient sliding for convex optimization. SIAM Journal on Optimization 26 (2), 1379–1409.
Lee, Y.T., Sidford, A., 2013. Efficient accelerated coordinate descent methods and faster algorithms for solving linear systems. In: Proceedings of the 2013 IEEE 54th Annual Symposium on Foundations of Computer Science, FOCS '13. IEEE Computer Society, Washington, DC, USA, pp. 147–156.
Levitin, E.S., Polyak, B.T., 1966. Constrained minimization methods. USSR Computational mathematics and mathematical physics 6 (5), 1–50.
Levy, K.Y., Yurtsever, A., Cevher, V., 2018. Online adaptive methods, universality and acceleration. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Vol. 31. Curran Associates, Inc., pp. 6500–6509.
Lin, F., Fardad, M., Jovanović, M.R., 2012. Sparse feedback synthesis via the alternating direction method of multipliers. In: 2012 American Control Conference (ACC). IEEE, pp. 4765–4770.
Lin, H., Mairal, J., Harchaoui, Z., 2015. A universal catalyst for first-order optimization. In: Proceedings of the 28th International Conference on Neural Information Processing Systems, NIPS'15. MIT Press, Cambridge, MA, USA, pp. 3384–3392.
Lin, Q., Lu, Z., Xiao, L., 2014. An accelerated proximal coordinate gradient method. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems, Vol. 27. Curran Associates, Inc., pp. 3059–3067.
Lin, T., Ho, N., Chen, X., Cuturi, M., Jordan, M., 2020. Fixed-support wasserstein barycenters: Computational hardness and fast algorithm. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (Eds.), Advances in Neural Information Processing Systems, Vol. 33. Curran Associates, Inc., pp. 5368–5380.
Lin, T., Ho, N., Cuturi, M., Jordan, M.I.. On the complexity of approximating multimarginal optimal transport.
Lin, T., Ho, N., Jordan, M., 2019. On efficient optimal transport: An analysis of greedy and accelerated mirror descent algorithms. In: Chaudhuri, K., Salakhutdinov, R. (Eds.), Proceedings of the 36th International Conference on Machine Learning, Vol. 97 of Proceedings of Machine Learning Research, PMLR, Long Beach, California, USA, pp. 3982–3991.
Ljung, L., Pflug, G., Walk, H., 2012. Stochastic approximation and optimization of random systems, 17. Birkhäuser.
Lu, H., Freund, R., Nesterov, Y., 2018. Relatively smooth convex optimization by first-order methods, and applications. SIAM Journal on Optimization 28 (1), 333–354.
Malitsky, Y., Pock, T., 2018. A first-order primal-dual algorithm with linesearch. SIAM Journal on Optimization 28 (1), 411–432.
Martinet, B., 1970. Régularisation d'inéquations variationnelles par approximations successives. Revue française d'informatique et de recherche opérationnelle. Série rouge 4 (R3), 154–158.
Mertikopoulos, P., Staudigl, M., 2018. On the convergence of gradient-like flows with noisy gradient input. SIAM Journal on Optimization 28 (1), 163–197.
Mertikopoulos, P., Staudigl, M., 2018. Stochastic mirror descent dynamics and their convergence in monotone variational inequalities. Journal of Optimization Theory and Applications 179 (3), 838–867.
Monteiro, R., Svaiter, B., 2013. An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. SIAM Journal on Optimization 23 (2), 1092–1125.
Moreau, J.J., 1965. Proximité et dualité dans un espace hilbertien. Bulletin de la Société mathématique de France 93, 273–299.
Necoara, I., Nesterov, Y., Glineur, F., 2019. Linear convergence of first order methods for non-strongly convex optimization. Mathematical Programming 175 (1), 69–107.
Nemirovski, A., 1982. Orth-method for smooth convex optimization. Izvestia AN SSSR, Transl.: Eng. Cybern. Soviet J. Comput. Syst. Sci 2, 937–947.

Nemirovski, A., 2004. Prox-method with rate of convergence $o(1/t)$ for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. SIAM Journal on Optimization 15 (1), 229–251.
Nemirovski, A.S., Yudin, D.B., 1983. Problem Complexity and Method Efficiency in Optimization. Wiley, New York, NY.
Nemirovskii, A., Nesterov, Y., 1985. Optimal methods of smooth convex minimization. USSR Computational Mathematics and Mathematical Physics 25 (2), 21–30.
Nesterov, Y., 2019. Implementable tensor methods in unconstrained convex optimization. Mathematical Programming doi:10.1007/s10107-019-01449-1.
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.
Nesterov, Y., 2005. Excessive gap technique in nonsmooth convex minimization. SIAM Journal on Optimization 16 (1), 235–249.
Nesterov, Y., 2005. Smooth minimization of non-smooth functions. Mathematical Programming 103 (1), 127–152.
Nesterov, Y., 2007. Dual extrapolation and its applications to solving variational inequalities and related problems. Mathematical Programming 109 (2), 319–344.
Nesterov, Y., 2008. Accelerating the cubic regularization of newton's method on convex problems. Mathematical Programming 112 (1), 159–181.
Nesterov, Y., 2009. Primal-dual subgradient methods for convex problems. Mathematical Programming 120 (1), 221–259.
Nesterov, Y., 2012. Efficiency of coordinate descent methods on huge-scale optimization problems. SIAM Journal on Optimization 22 (2), 341–362.
Nesterov, Y., 2013. Gradient methods for minimizing composite functions. Mathematical Programming 140 (1), 125–161.
Nesterov, Y., 2015. Universal gradient methods for convex optimization problems. Mathematical Programming 152 (1), 381–404.
Nesterov, Y., 2018. Complexity bounds for primal-dual methods minimizing the model of objective function. Mathematical Programming 171 (1-2), 311–330.
Nesterov, Y., 2018. Lectures on convex optimization. Vol. 137 of Springer Optimization and Its Applications. Springer International Publishing.
Nesterov, Y., Gasnikov, A., Guminov, S., Dvurechensky, P., 2020. Primal-dual accelerated gradient methods with small-dimensional relaxation oracle. Optimization Methods and Software 1–28. doi:10.1080/10556788.2020.1731747.
Nesterov, Y., Nemirovski, A., 1994. Interior Point Polynomial methods in Convex programming. SIAM Publications.
Nesterov, Y., Spokoiny, V., 2017. Random gradient-free minimization of convex functions. Found. Comput. Math. 17 (2), 527–566.
Nesterov, Y., Stich, S.U., 2017. Efficiency of the accelerated coordinate descent method on structured optimization problems. SIAM Journal on Optimization 27 (1), 110–123. First presented in May 2015 http://www.mathnet.ru:8080/PresentFiles/11909/7_nesterov.pdf
Odor, G., Li, Y.-H., Yurtsever, A., Hsieh, Y.-P., Tran-Dinh, Q., Halabi, M.E., Cevher, V., 2016. Frank-wolfe works for non-lipschitz continuous gradient objectives: scalable poisson phase retrieval. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Ieee, pp. 6230–6234.
Opial, Z., 1967. Weak convergence of the sequence of successive approximations for nonexpansive mappings. Bulletin of the American Mathematical Society 73 (4), 591–597.
Parikh, N., Boyd, S., 2014. Proximal algorithms. Foundations and Trends® in Optimization 1 (3), 127–239.
Pedregosa, F., Negiar, G., Askari, A., Jaggi, M., 2020. Linearly convergent frank-wolfe with backtracking line-search. In: International Conference on Artificial Intelligence and Statistics, PMLR, pp. 1–10.
Pokutta, S., 2020. Restarting algorithms: Sometimes there is free lunch. In: Hebrard, E., Musliu, N. (Eds.), Integration of Constraint Programming, Artificial Intelligence, and Operations Research. Springer International Publishing, Cham, pp. 22–38.
Polyak, B.T., 1964. Some methods of speeding up the convergence of iteration methods. USSR Computational Mathematics and Mathematical Physics 4 (5), 1–17.
Polyak, B.T., 1987. Introduction to Optimization. Optimization Software.
Robbins, H., Monro, S., 1951. A stochastic approximation method. The Annals of Mathematical Statistics 22 (3), 400–407.
Robinson, S.M., 1982. Generalized equations and their solutions, part ii: applications to nonlinear programming. In: Optimality and Stability in Mathematical Programming. Springer, pp. 200–221.
Rockafellar, R.T., 1970. Convex Analysis. Princeton University Press, Princeton.
Rockafellar, R.T., 1976. Augmented lagrangians and applications of the proximal point algorithm in convex programming. Mathematics of operations research 1 (2), 97–116.
Rockafellar, R.T., 1976. Monotone operators and the proximal point algorithm. SIAM journal on control and optimization 14 (5), 877–898.
Rockafellar, R.T., Wets, R.J.B., 1998. Variational analysis. Vol. 317 of A Series of Comprehensive Studies in Mathematics. Springer-Verlag, Berlin.
Rogozin, A., Bochko, M., Dvurechensky, P., Gasnikov, A., Lukoshkin, V., 2021. An accelerated method for decentralized distributed stochastic optimization over time-varying graphs. In: 2021 60th IEEE Conference on Decision and Control (CDC).
Roulet, V., d'Aspremont, A., 2017. Sharpness, restart and acceleration. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Vol. 30. Curran Associates, Inc., pp. 1119–1129.
Shalev-Shwartz, S., Zhang, T., 2014. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. In: Xing, E.P., Jebara, T. (Eds.), Proceedings of the 31st International Conference on Machine Learning, Vol. 32 of Proceedings of Machine Learning Research, PMLR, Bejing, China, pp. 64–72.
Shapiro, A., Dentcheva, D., Ruszczyński, A., 2009. Lectures on stochastic programming. Society for Industrial and Applied Mathematics.
Shefi, R., Teboulle, M., 2014. Rate of convergence analysis of decomposition methods based on the proximal method of multipliers for convex minimization. SIAM Journal on Optimization 24 (1), 269–297.
Shi, B., Du, S.S., Su, W., Jordan, M.I., 2019. Acceleration via symplectic discretization of high-resolution differential equations. Advances in Neural Information Processing Systems 5744–5752.
Sorin, S., 2000. A First-Course on Zero-Sum Repeated Games. Springer.
Stonyakin, F., Gasnikov, A., Dvurechensky, P., Alkousa, M., Titov, A.. Generalized Mirror Prox for monotone variational inequalities: Universality and inexact oracle.
Stonyakin, F., Tyurin, A., Gasnikov, A., Dvurechensky, P., Agafonov, A., Dvinskikh, D., Alkousa, M., Pasechnyuk, D., Artamonov, S., Piskunova, V., 2020. Inexact model: A framework for optimization and variational inequalities. Optimization Methods and Software doi:10.1080/10556788.2021.1924714.
Stonyakin, F.S., Dvinskikh, D., Dvurechensky, P., Kroshnin, A., Kuznetsova, O., Agafonov, A., Gasnikov, A., Tyurin, A., Uribe, C.A., Pasechnyuk, D., Artamonov, S., 2019. Gradient methods for problems with inexact model of the objective. In: Khachay, M., Kochetov, Y., Pardalos, P. (Eds.), Mathematical Optimization Theory and Operations Research. Springer International Publishing, Cham, pp. 97–114.
Sturm, J.F., 1999. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. Optimization methods and software 11 (1-4), 625–653.
Su, W., Boyd, S., Candes, E.J., 2016. A differential equation for modeling nesterov's accelerated gradient method: Theory and insights. Journal of Machine Learning Research.
Sun, A.X., Phan, D.T., Ghosh, S., 2013. Fully decentralized ac optimal power flow algorithms. In: 2013 IEEE Power & Energy Society General Meeting. IEEE, pp. 1–5.
Sun, R.. Optimization for deep learning: theory and algorithms.
Teboulle, M., 1992. Entropic proximal mappings with applications to nonlinear programming. Mathematics of Operations Research 17, 670–690.
Teboulle, M., 2018. A simplified view of first order methods for optimization. Mathematical Programming 170 (1), 67–96.
Todd, M.J., 2016. Minimum-volume ellipsoids. Society for Industrial and Applied Mathematics.
Tran-Dinh, Q., Alacaoglu, A., Fercoq, O., Cevher, V., 2020. An adaptive primal-dual framework for nonsmooth convex minimization. Mathematical Programming Computation 12 (3), 451–491.
Tran-Dinh, Q., Cevher, V., 2014. Constrained convex minimization via model-based excessive gap. In: Proceedings of the 27th International Conference on Neural Information Processing Systems, NIPS'14. MIT Press, Cambridge, MA, USA, pp. 721–729.
Tran-Dinh, Q., Fercoq, O., Cevher, V., 2018. A smooth primal-dual optimization framework for nonsmooth composite convex minimization. SIAM Journal on Optimization 28 (1), 96–134.
Tseng, P., 1991. Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. SIAM Journal on Control and Optimization 29 (1), 119–138.
Tseng, P., 2008. On accelerated proximal gradient methods for convex-concave optimization. Tech. rep.. MIT.
Tupitsa, N., Dvurechensky, P., Gasnikov, A., Uribe, C.A., 2020. Multimarginal optimal transport by accelerated alternating minimization. In: 2020 59th IEEE Conference on Decision and Control (CDC), pp. 6132–6137.
Uribe, C.A., Dvinskikh, D., Dvurechensky, P., Gasnikov, A., Nedić, A., 2018. Distributed computation of Wasserstein barycenters over networks. In: 2018 IEEE Conference on Decision and Control (CDC), pp. 6544–6549.
Van Nguyen, Q., 2017. Forward-backward splitting with bregman distances. Vietnam Journal of Mathematics 45 (3), 519–539.
Von Hohenbalken, B., 1977. Simplicial decomposition in nonlinear programming algorithms. Mathematical Programming 13 (1), 49–68.
Vorontsova, E.A., Gasnikov, A.V., Gorbunov, E.A., 2019. Accelerated directional search with non-euclidean prox-structure. Automation and Remote Control 80 (4), 693–707.
Vorontsova, E.A., Gasnikov, A.V., Gorbunov, E.A., Dvurechenskii, P.E., 2019. Accelerated gradient-free optimization methods with a non-euclidean proximal operator. Automation and Remote Control 80 (8), 1487–1501.
Wibisono, A., Wilson, A.C., Jordan, M.I., 2016. A variational perspective on accelerated methods in optimization. Proceedings of the National Academy of Sciences 113 (47), E7351.
Wolfe, P., 1970. Convergence theory in nonlinear programming. In: Abadie, J. (Ed.), Integer and nonlinear programming. North-Holland, Amsterdam.
Wright, S.J., 2018. Optimization algorithms for data analysis. The Mathematics of Data 25, 49.
Yang, J., Zhang, Y., 2011. Alternating direction algorithms for \ell_1-problems in compressive sensing. SIAM journal on scientific computing 33 (1), 250–278.
Yuan, X., 2012. Alternating direction method for covariance selection models. Journal of Scientific Computing 51 (2), 261–273.
Zhang, Y., Xiao, L., 2015. Stochastic primal-dual coordinate method for regularized empirical risk minimization. In: Bach, F., Blei, D. (Eds.), Proceedings of the 32nd International Conference on Machine Learning, Vol. 37 of Proceedings of Machine Learning Research, PMLR, Lille, France, pp. 353–361.
Zhao, R., Freund, R.M.. Analysis of the Frank-Wolfe method for logarithmically-homogeneous barriers, with an extension.