

Exploiting problem structure in optimization under uncertainty via online convex optimization

Nam Ho-Nguyen¹  · Fatma Kılınç-Karzan¹ 

Received: 11 November 2016 / Accepted: 16 March 2018 / Published online: 30 March 2018
© Springer-Verlag GmbH Germany, part of Springer Nature and Mathematical Optimization Society 2018

Abstract In this paper, we consider two paradigms that are developed to account for uncertainty in optimization models: robust optimization (RO) and joint estimation-optimization (JEO). We examine recent developments on efficient and scalable iterative first-order methods for these problems, and show that these iterative methods can be viewed through the lens of online convex optimization (OCO). The standard OCO framework has seen much success for its ability to handle decision-making in dynamic, uncertain, and even adversarial environments. Nevertheless, our applications of interest present further flexibility in OCO via three simple modifications to standard OCO assumptions: we introduce two new concepts of *weighted regret* and *online saddle point problems* and study the possibility of making *lookahead* (anticipatory) decisions. Our analyses demonstrate that these flexibilities introduced into the OCO framework have significant consequences whenever they are applicable. For example, in the strongly convex case, minimizing unweighted regret has a proven optimal bound of $O(\log(T)/T)$, whereas we show that a bound of $O(1/T)$ is possible when we consider weighted regret. Similarly, for the smooth case, considering 1-lookahead decisions results in a $O(1/T)$ bound, compared to $O(1/\sqrt{T})$ in the standard OCO setting. Consequently, these OCO tools are instrumental in exploiting structural properties of functions and results in improved convergence rates for RO and JEO. In certain cases, our results for RO and JEO match the best known or optimal rates in the corresponding problem classes without data uncertainty.

✉ Fatma Kılınç-Karzan
fkilinc@andrew.cmu.edu

Nam Ho-Nguyen
hnhanh@andrew.cmu.edu

¹ Tepper School of Business, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Mathematics Subject Classification 90C25 (Convex programming) · 90C06 (Large-scale problems)

1 Introduction

We consider the following convex optimization problem with given input data $u = [u^1; \dots; u^m]$:

$$\min_x \left\{ f(x) : f^i(x, u^i) \leq 0, \forall i = 1, \dots, m, x \in X \right\}, \quad (1)$$

where X is a convex domain, and f, f^1, \dots, f^m are all convex functions of $x \in X$. Often, the data u defining the problem (1) are uncertain or misspecified (only approximations of the true data are available). In many applications, optimization with noisy data can have a large negative effect on performance. As an example, in portfolio optimization, the covariance matrix is difficult to estimate, and the mean-variance model is notoriously sensitive to data perturbations [16]. To address this, several methodologies have been developed to handle the uncertainty or misspecification of data in (1). In this paper, we consider iterative solution methods for two different paradigms with tractable models that handle uncertainty, namely, robust optimization and joint estimation-optimization problems, and establish a deeper connection between such iterative approaches for these problems and online convex optimization.

Robust optimization (RO) addresses data uncertainty in (1) by seeking a solution $x \in X$ that is feasible for all data realizations u^i from a fixed uncertainty set U^i for each constraint $f^i, i = 1, \dots, m$. More specifically, convex RO seeks to solve

$$\min_x \left\{ f(x) : \sup_{u^i \in U^i} f^i(x, u^i) \leq 0, i = 1, \dots, m, x \in X \right\}. \quad (2)$$

RO has been extensively studied in the literature, and we refer the reader to the paper by Ben-Tal and Nemirovski [6], the book by Ben-Tal et al. [4] and surveys [7–9, 13] for a detailed account of RO theory and its numerous applications.

The traditional solution method for RO is based on reformulating it first into an equivalent deterministic robust counterpart problem via duality theory, and then solving the robust counterpart as a deterministic convex optimization problem. However, the robust counterpart approach often leads to larger and much less scalable problems than the associated nominal problem of (2) where the uncertain data (noise) $[u^1; \dots; u^m]$ is fixed to a given value. For example, it is well-known that the robust counterpart of a second-order cone program with ellipsoidal uncertainty is a semidefinite program. Recently, there have been several interesting developments on *iterative methods* for solving RO problems that bypass the robust counterpart approach, see e.g., [5, 19, 28]. These methods solve (2) by iteratively updating the solution x and the noise $[u^1; \dots; u^m]$ to approximate their optimum values.

Joint estimation-optimization (JEO) considers the setting where we only have uncertainty u in the objective $f(x, u)$, and that the ‘correct’ data value u^* may be

learned through a distinct learning process, i.e., it is characterized as a solution to a separate optimization problem $\min_u \{g(u) : u \in U\}$. More precisely, JEO aims to solve

$$\min_x \{f(x, u^*) : x \in X\} \quad (\text{Opt}(u^*))$$

$$\text{where } u^* \in \arg \min_u \{g(u) : u \in U\}. \quad (\text{Est})$$

In many practical situations, JEO is solved via a *sequential* method: first minimize $g(u)$ to find u^* , then minimize $f(x, u^*)$ to solve the problem. However, we often cannot solve for u^* exactly, but instead must settle for an approximation $\bar{u} \approx u^*$. With such a strategy, under mild Lipschitz continuity assumptions, the accuracy of $(\text{Opt}(u^*))$ is controlled by the norm of $\|\bar{u} - u^*\|$. Nevertheless, this creates the following ‘inconsistency’ problem: when minimizing $f(x, \bar{u})$, we create a sequence of points $x_t \in X, t \geq 1$, which converge to the minimum of $f(x, \bar{u})$; however, the sequence will not converge to the desired minimum $(\text{Opt}(u^*))$, and in fact will only be within $O(\|\bar{u} - u^*\|)$ accuracy. That is, this approach cannot provide asymptotically accurate solutions x_t . It is possible to achieve consistency via a naïve scheme by creating a sequence of approximations u_t such that $\|u_t - u^*\| \rightarrow 0$, and for each u_t , minimizing $f(x, u_t)$ up to accuracy $O(\|u_t - u^*\|)$ to obtain x_t . Then the sequence x_t will be consistent, i.e., $\lim_{t \rightarrow \infty} f(x_t, u_t)$ converges to the optimum value of $(\text{Opt}(u^*))$. This naïve scheme comes with two disadvantages: each step t involves solving a complete minimization problem up to some accuracy, and furthermore the accuracy must improve at each new step. The main problem is that at each step t , the information from the previous steps cannot be utilized, hence they are essentially wasted. To address this, Jiang and Shanbhag [21, 22] and Ahmadi and Shanbhag [2] propose a scheme that *jointly* solves the estimation and optimization problems, which we refer to as JEO. With this scheme, they can efficiently generate a sequence of points x_t and u_t such that $f(x_t, u_t)$ will indeed converge to the desired minimum $(\text{Opt}(u^*))$, and give corresponding non-asymptotic error rates. In particular, their scheme can exploit previous information in a principled manner by ensuring that the effort in each step consists only of first-order updates.

The iterative RO methods of [5, 19, 28] and the simultaneous JEO approach of [2, 21, 22] both build a solution \bar{x} in very similar ways: iteratively generate a *solution* sequence x_t and a *data* sequence u_t (for $t \geq 1$) that approximate the ‘ideal’ solution and data points respectively, then perform averaging after a finite number of iterations T to build an approximate solution \bar{x} . A key feature in both approaches is that generating the next *solution* point x_t uses information from the *data* sequence u_1, \dots, u_{t-1} up to iteration $t-1$, and vice versa. This intricacy is handled via tools from *online convex optimization* (OCO) in the case of RO in [19]; we will demonstrate later that the simultaneous approach of JEO can also be viewed through the lens of OCO.

OCO is part of the broader online learning (or sequential prediction) framework, which was introduced as a method to optimize decisions in a dynamic environment where the objective is changing at every time period, and at each time period we are allowed to adapt to our changing environment based on accumulated information. The

origin of the online learning model can be traced back to the work of Robbins [35] on compound statistical decision problems. This framework has found a diverse set of applications in many fields; for further details see [14, 17, 36].

In standard OCO, we are given a convex domain X and a finite time horizon T . In each time period $t = 1, \dots, T$, an online player chooses a decision $x_t \in X$ based on *past* information from time steps $1, \dots, t - 1$ only. Then, a convex loss function $f_t : X \rightarrow \mathbb{R}$ is revealed, and the player suffers loss $f_t(x_t)$ and gets some feedback typically in the form of first-order information $\nabla f_t(x_t)$. We call this restriction on the player *non-anticipatory*, since the player cannot anticipate the next loss f_t ahead of deciding x_t .¹ In addition, it is usually assumed that the functions f_t are set in advance—possibly by an all-powerful adversary that has full knowledge of our learning algorithm—and we know of only the general class of these functions. As such, it is unreasonable to compare the loss of the player across the time horizon to the best possible loss, which would require full knowledge of f_t in advance of choosing x_t . Instead, the player's sequence of decisions x_t is evaluated against the best fixed decision in hindsight, and the (average) difference is defined to be the *regret*:

$$\frac{1}{T} \sum_{t=1}^T f_t(x_t) - \inf_{x \in X} \frac{1}{T} \sum_{t=1}^T f_t(x). \quad (3)$$

The goal in OCO is to design efficient *regret minimizing* algorithms that generate the points x_t so that the regret tends to zero as T increases. Therefore, in OCO we seek non-anticipatory algorithms to choose x_t that guarantee

$$\frac{1}{T} \sum_{t=1}^T f_t(x_t) - \inf_{x \in X} \frac{1}{T} \sum_{t=1}^T f_t(x) \leq r(T), \quad \lim_{T \rightarrow \infty} r(T) = 0,$$

and the performance of our algorithms is measured by how quickly $r(T)$ tends to 0. While regret may seem like a weak evaluation metric, the fact that regret minimizing algorithms exist for *any* sequence of functions f_t is quite powerful. In particular, it allows us to handle the intricacies of simultaneously generating x_t and u_t .

In this paper, we view iterative approaches to both RO and JEO in a *unified* manner through the lens of OCO, and demonstrate how such a view opens up the possibility of introducing simple flexibilities to standard OCO that are instrumental in exploiting structural properties of the problems and results in improved convergence rates for RO and JEO. Our nonstandard yet flexible OCO framework is obtained through three simple modifications to the standard OCO assumptions. These modifications are as follows:

- (i) We introduce the concept of *weighted regret*, where instead of taking uniform averages with weights $\theta_t = 1/T$ in (3), we are allowed to use *nonuniform* weighted averages. From a modeling perspective, this allows us to capture situations where decisions x_t at different time steps t have varying importance.

¹ This is also referred to as a *0-lookahead* framework.

- (ii) We introduce the *online saddle point (SP) problem*, where at each step we receive a convex-concave function $\phi_t(x, y)$ and must choose x and y . This is an extension of the well-studied offline convex-concave SP problem, and can be thought of as a dynamic zero-sum two-player game where at each step the players are restricted to make only one move.
- (iii) We explore the implications of *1-lookahead* or *anticipatory* decisions, where the learner can receive limited information on the function f_t before making the decision x_t . This is in contrast to most OCO settings where the learner must choose x_t before any information on f_t is revealed.

Under this new OCO framework with flexibilities, we present and discuss algorithms accompanied with new regret bounds that can be better than the standard OCO ones when favorable problem structure is present. Our algorithms are based on online adaptations of two commonly used offline first-order methods (FOMs) from convex optimization, namely Mirror Descent and Mirror Prox. We present our developments in the flexible proximal setup of Juditsky and Nemirovski [23, 24] which can be further customized to the geometry of the domains.

Our analyses demonstrate that these flexibilities introduced into the OCO framework have significant consequences whenever they are applicable. For example, in the strongly convex case, minimizing unweighted regret has a proven optimal bound of $O(\log(T)/T)$, whereas we show that a bound of $O(1/T)$ is possible when we consider weighted regret. Similarly, for the smooth case, considering 1-lookahead decisions results in a $O(1/T)$ bound, compared to $O(1/\sqrt{T})$ in the standard OCO setting (see Remarks 6 and 12).

Consequently, these new regret bounds are pivotal in exploiting structural properties of functions to achieve improved convergence rates for both RO and JEO. For example, in the case of RO, we demonstrate that it is possible to achieve a convergence rate of $O(1/T)$, improving over the standard $O(1/\sqrt{T})$ rate, when the functions f^i satisfy certain strong convexity (or smoothness) assumptions. These new developments then allow us to partially resolve an open question from [5] on the complexity lower bounds for solving RO via iterative techniques. For JEO, in addition to covering the standard setups from [2] in a unified manner and extending them to the more general proximal setup, we explore a setting which was not covered in the work of [2, 21, 22], when f is non-smooth and strongly convex. In this setting, we provide an improved convergence rate of $O(1/T)$, which is the optimal rate even if we had the correct data u^* upfront.

1.1 Related work

For the RO problem (2), Mutapcic and Boyd [28] analyzed an iterative cutting-plane-type approach, which has an exponential-in-dimension convergence guarantee of $(1 + O(1/\epsilon))^n$ iterations to obtain an ϵ -optimal solution. Ben-Tal et al. [5] suggests an approach using online convex optimization, which guarantees convergence in $O(1/\epsilon^2)$ iterations. Each iteration of [28] and [5] requires solving at least a nominal version of (2), which can be expensive. The recent work of [19] provides a unifying framework for both approaches [28] and [5] via OCO, and presents a refined analysis which allows for a significant reduction in the computational effort of each

iteration to simple first-order updates only, while enjoying a convergence guarantee of $O(1/\epsilon^2 \log(1/\epsilon))$. This reduction in the per-iteration computational cost in the approach of [19] is enough to offset the extra $\log(1/\epsilon)$ factor in the overall number of iterations; see [19, Sect. 4.4] for a detailed discussion. In this paper, we examine the OCO-based framework of [19], and provide improved convergence results under structural assumptions on the properties of functions f^i for this framework.

Jiang and Shanbhag [21, 22] introduced and studied the JEO problem ($\text{Opt}(u^*)$)-(**Est**) in a stochastic setting, and Ahmadi and Shanbhag [2] examined the deterministic case. In this paper, we consider the deterministic JEO problem, for which [2] provided some remarkable convergence results. Specifically, they analyze the setting when g is strongly convex and both f and g are smooth. In [2, Proposition 3], when f is also strongly convex, a gradient descent-type algorithm is given with error bound of $O(T\beta^T)$ after T iterations, for some $0 < \beta < 1$. In [2, Proposition 4], when f is only convex, the same algorithm (with different tuning parameters) ensures an error bound of $O(1/T)$. Furthermore, when f does not enjoy strong convexity or smoothness, [2, Proposition 6] provides an error bound of $O(1/\sqrt{T})$. These results demonstrate that, despite access to only estimates of the true data with increasing accuracy, the simultaneous first-order JEO approach of [2] can achieve error bounds which are asymptotically as good or almost as good as first-order methods equipped with exact data. Similar to RO, we show that the JEO problem can be viewed through the lens of OCO, and explore possible improvements through our flexible OCO framework.

To our knowledge, the concept of weighted regret in OCO has not been studied before. However, modification of aggregation weights as a means to speed up convergence has been explored in the stochastic optimization setting under strong convexity assumptions; see [18, 26, 31, 32]. Our work can be seen as an extension of these results to the adversarial setting, and in fact, one of our results, Theorem 2, is a simple generalization of a result from [26]. Nevertheless, by stating the result in the general adversarial setting of OCO, we are able to apply it to RO and JEO, which do not fit within the stochastic optimization framework.

Mahdavi et al. [27] introduce a special case of online SP problems to handle difficult constraints in OCO problems. The difficult constraints $s^i(x) \leq 0$ are embedded into each loss function $f_t(x)$ by aggregation with Lagrange dual multipliers y , to form a new loss function $\phi_t(x, y) = f_t(x) + \sum_{i=1}^m y^{(i)} s^i(x)$, which is convex in x and concave in y . Both primal and dual variables x, y are then updated each time step to obtain bounds on the regret and the violation $\sum_{t=1}^T s^i(x)$. The papers [20, 25] also use similar duality ideas for handling difficult constraints and objectives in online settings. Nevertheless, the convergence rates given in these papers are the usual $O(1/\sqrt{T})$ or slower. In this paper, we analyze online SP problems more generally, and explore faster rates in the 1-lookahead setting.

Online settings with 1-lookahead naturally arise in metrical task systems [3, 10, 12] and online display advertising [20]. In these settings, the variation of the decisions x_1, \dots, x_T across the time horizon is also penalized, and the performance of the sequence is measured as the competitive ratio of the realized loss with the best possible loss [3, 10, 12] or as a dynamic regret term [20]. Both competitive ratio and dynamic regret objectives do not fit to our framework. Moreover, [3, Sect. 4] show that standard regret and competitive ratio cannot be simultaneously optimized.

From an algorithmic point of view, Mahdavi et al. [27], Chiang et al. [15] and Yang et al. [38] examine online variants of the Mirror Prox algorithm when it is limited to work with *only past information*. In particular, [15, 38] provide regret bounds with ‘gradual variation’ terms, which capture how quickly the sequence of functions f_t vary. Rakhlin and Sridharan [33, 34] analyze 1-lookahead decisions in OCO through the lens of *predictable sequences*. They explore how one can exploit information from a single sequence M_1, \dots, M_T in an online framework, where each term M_t is revealed to the player *prior* to choosing the decision x_t . They provide the Optimistic Mirror Descent algorithm, which is essentially a generalization of Mirror Prox [29], to exploit the sequence M_1, \dots, M_T . In [33, 34], they focus on uncoupled dynamics and zero-sum games, whereas our work focuses on more general and flexible OCO problems, and designing and applying proper generalizations of FOMs such as Mirror Prox to more flexible OCO problems arising in the context of coupled optimization problems. That said, our work in Sect. 3.3 is related to exploiting a specific predictable sequence; we elaborate on this in Remark 11.

1.2 Outline

In Sect. 2, we derive the concepts of weighted regret and online SP problems via the notion of affine regret, thereby allowing us to approach both problems through a common algorithmic framework, which we describe in Sect. 3. After introducing the basic proximal setup in Sect. 3.1, we analyze weighted regret OCO and online SP problems via the online Mirror Descent algorithm, and derive the standard $O(1/\sqrt{T})$ convergence rates in Sect. 3.2. We also show how strong convexity assumptions on the loss functions allow us to improve this to $O(1/T)$. In Sect. 3.3, we introduce and analyze an online variant of the Mirror Prox algorithm that achieves $O(1/T)$ convergence rates under 1-lookahead and smoothness assumptions. In Sects. 4 and 5, we apply the developments of Sects. 2 and 3 to the RO and JEO problems respectively. We close with a summary of our results and some future directions in Sect. 6.

1.3 Notation

For a positive integer $n \in \mathbb{N}$, we let $[n] = \{1, \dots, n\}$ and define $\Delta_n := \{x \in \mathbb{R}_+^n : \sum_{i \in [n]} x_i = 1\}$ to be the standard simplex. Throughout the paper, the subscript, e.g., $x_t, y_t, z_t, f_t, \phi_t$, is used to attribute items to the t th time period or iteration. We use the notation $\{x_t\}_{t=1}^T$ to denote the collection of items $\{x_1, \dots, x_T\}$. Given a vector $x \in \mathbb{R}^n$, we let $x^{(j)}$ denote its j th coordinate for $j \in [n]$. One exception we make to this notation is that we always denote the convex combination weights $\theta \in \Delta_T$ with θ_t . We use Matlab notation for vectors and matrices, i.e., $[x; y]$ denotes the concatenation of two column vectors x, y . Given $x, y \in \mathbb{R}^n$, $\langle x, y \rangle$ corresponds to the usual inner product of x and y . Given a norm $\|\cdot\|$, we let $\|\cdot\|_*$ denote the corresponding dual norm. For $x \in \mathbb{R}^n$, $\|x\|_2$ denotes the Euclidean ℓ_2 -norm of x defined as $\|x\|_2 = \sqrt{\langle x, x \rangle}$. We let $\partial f(x)$ be the subdifferential of f taken at x . We abuse notation slightly by denoting $\nabla f(x)$ for both the gradient of function f at x if f is differentiable and a subgradient

of f at x , even if f is not differentiable. If ϕ is of the form $\phi(x, y)$, then $\nabla_x \phi(x, y)$ denotes the subgradient of ϕ at x while keeping the other variables fixed at y .

2 Generalized regret in online convex optimization

In this section, we examine a number of generalizations of the regret concept and show how they can all be unified via an affine regret concept. We start with *affine regret* given by

$$\sum_{t=1}^T \langle \xi_t, x_t \rangle - \inf_{x \in X} \sum_{t=1}^T \langle \xi_t, x \rangle = \sup_{x \in X} \sum_{t=1}^T \langle \xi_t, x_t - x \rangle, \quad (4)$$

where ξ_t is a given loss vector at time t . Suppose that when the player makes a decision $x_t \in X$, the adversary returns $\xi_t = \nabla f_t(x_t)$, where $f_t : X \rightarrow \mathbb{R}$ is some convex function. Then by the subgradient inequality we have $f_t(x_t) - f_t(x) \leq \langle \nabla f_t(x_t), x_t - x \rangle = \langle \xi_t, x_t - x \rangle$ and hence

$$\frac{1}{T} \sum_{t=1}^T f_t(x_t) - \inf_{x \in X} \frac{1}{T} \sum_{t=1}^T f_t(x) \leq \sup_{x \in X} \frac{1}{T} \sum_{t=1}^T \langle \xi_t, x_t - x \rangle.$$

This implies that the standard regret in OCO is upper bounded by the affine regret (4) where the loss vectors ξ_t are the subgradients $\nabla f_t(x_t)$. Then, to minimize usual regret, it is enough to minimize the affine regret. That said, as will be discussed in Sect. 3, in order to obtain improved rates of convergence, we must go beyond affine regret and exploit further structural properties of the functions f_t . Even then, all the bounds from Sect. 3 involve upper bounding the affine regret (4) in some fashion.

2.1 OCO with weighted regret

The first flexibility we introduce to the OCO framework is scaling each time step t by weights $\theta_t > 0$. With $\xi_t = \nabla f_t(x_t)$, and applying the subgradient inequality, this results in

$$\sum_{t=1}^T \theta_t f_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t f_t(x) \leq \sup_{x \in X} \sum_{t=1}^T \theta_t \langle \xi_t, x_t - x \rangle. \quad (5)$$

We define the left hand side of this inequality to be the *weighted regret*. From a modeling perspective, weighted regret enables us to model situations where later decisions x_t carry higher importance by placing higher weights θ_t on subsequent periods t (or vice versa). For example, in a repeated game where performance of a player is aggregated from the loss at each stage, we may want to weigh the later stages more heavily than the earlier stages, since earlier stages might be used to explore the opponents' strategy, whereas in later stages we expect the player to have converged to a (near)-optimal strategy.

On the practical side, weighted regret lets us choose weights θ_t to speed up convergence. In particular, when the functions f_t are strongly convex, our bounds of

$O(1/T)$ for weighted regret improve on the optimal regret bounds $O(\log(T)/T)$ for the uniform weight case. Furthermore, weighted regret bounds become important when solving RO problems, where we combine two different regret terms *which must have the same weights* θ to obtain bounds. We discuss these practical aspects fully in Sects. 3 and 4.

Finally, note that while it is possible to view weighted regret as a rescaling of the functions f_t with weights θ_t , such a view will inevitably change the parameters associated with functions f_t such as strong convexity. In contrast, working with the weighted regret concept circumvents this issue; see Sect. 3.2.2 for our study on exploiting strong convexity.

Because we are interested in taking a weighted average, henceforth we will assume that we have convex combination weights $\theta := (\theta_1, \dots, \theta_T) \in \Delta_T$. Thus, we seek OCO algorithms for selecting x_t that minimize the weighted regret and guarantee

$$\sum_{t=1}^T \theta_t f_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t f_t(x) \leq r(T), \quad \lim_{T \rightarrow \infty} r(T) = 0. \quad (6)$$

Regret bounds for online convex optimization algorithms naturally result in optimality gap bounds for the corresponding offline problems.

Remark 1 When the functions f_t remain the same throughout the time horizon, i.e., $f_t = f$ for all $t \in [T]$, and \bar{x} is taken to be the weighted sum of $\{x_t\}_{t=1}^T$ with weights $\theta \in \Delta_T$, the weighted regret in (6) naturally bounds the standard optimality gap of solution \bar{x} in the associated offline convex minimization problem $\min_{x \in X} f(x)$. \square

2.2 Online saddle point problems

The standard convex-concave saddle point (SP) problem is defined as

$$SV = \inf_{x \in X} \sup_{y \in Y} \phi(x, y) = \sup_{y \in Y} \inf_{x \in X} \phi(x, y), \quad (7)$$

where X, Y are nonempty compact convex sets in Euclidean spaces $\mathbb{E}_x, \mathbb{E}_y$ and the function $\phi(x, y)$ is convex in x and concave in y . Note that the latter equality in (7) holds because of the minimax theorem (see [37]) under assumptions of compactness and convexity of the sets X and Y , and ϕ admitting a convex-concave structure.

Any convex-concave SP problem (7) gives rise to two convex optimization problems that are dual to each other:

$$\begin{aligned} \text{Opt}(P) &= \inf_{x \in X} [\bar{\phi}(x) := \sup_{y \in Y} \phi(x, y)] \quad (P) \\ \text{Opt}(D) &= \sup_{y \in Y} [\underline{\phi}(y) := \inf_{x \in X} \phi(x, y)] \quad (D) \end{aligned}$$

with $\text{Opt}(P) = \text{Opt}(D) = SV$. SP problem (7) also leads to a monotone *variational inequality* (VI) problem on $\mathcal{Z} = X \times Y$:

$$\text{find } z_* \in \mathcal{Z} \text{ such that } \langle F(z), z - z_* \rangle \geq 0 \text{ for all } z \in \mathcal{Z},$$

where $F : \mathcal{Z} \mapsto \mathbb{E}_x \times \mathbb{E}_y$ is the monotone gradient operator given by

$$F(x, y) = [\nabla_x \phi(x, y); -\nabla_y \phi(x, y)].$$

It is well-known that the solutions to (7)—the saddle points of ϕ on $X \times Y$ —are exactly the pairs $[x; y]$ formed by optimal solutions to the problems (P) and (D) . They are also exactly the solutions to the associated VI problem.

We quantify the accuracy of a candidate solution $[\bar{x}, \bar{y}]$ to SP problem (7) with the *saddle point gap* given by

$$\epsilon_{\text{sad}}^\phi(\bar{x}, \bar{y}) := \bar{\phi}(\bar{x}) - \underline{\phi}(\bar{y}) = \underbrace{[\bar{\phi}(\bar{x}) - \text{Opt}(P)]}_{\geq 0} + \underbrace{[\text{Opt}(D) - \underline{\phi}(\bar{y})]}_{\geq 0}. \quad (8)$$

In order to solve (7) to accuracy $\epsilon > 0$, we must find $[x^\epsilon; y^\epsilon]$ such that the SP gap $\epsilon_{\text{sad}}^\phi(x^\epsilon, y^\epsilon) \leq \epsilon$, i.e., it is small.

When $\phi(x, y)$ is convex in x , so is the function $\bar{\phi}(x) = \sup_{y \in Y} \phi(x, y)$. Hence, (7) has the interpretation of simply minimizing a convex function $\bar{\phi}(x)$ over the domain X . However, taking the supremum over $y \in Y$ in $\bar{\phi}(\cdot)$ may destroy some important structural properties of $\phi(x, y)$ such as smoothness. The main motivation for designing specific FOMs to solve offline SP problems in [29, 30] is to exploit such structural properties of ϕ via the monotone gradient operator F and rather not work with $\bar{\phi}(x)$ explicitly.

A natural extension of convex-concave SP problems to an online setup is as follows: We are given domains X, Y and a time horizon T . At each time period $t \in [T]$, we simultaneously select $[x_t; y_t] \in X \times Y$ and learn $\phi_t(x_t, y_t)$ based on a convex-concave function $\phi_t(x, y)$ revealed at the time period. We can think of this as a dynamic two-player zero-sum game, where at each stage t , each player makes only one move (decision) $x_t \in X$ and $y_t \in Y$ as opposed to reaching to an approximate equilibrium. Then the goal of each player is to minimize their weighted regrets given the sequence of moves of the other player,² i.e.,

$$\sum_{t=1}^T \theta_t \phi_t(x_t, y_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t \phi_t(x, y_t) \quad \text{and} \quad \sup_{y \in Y} \sum_{t=1}^T \theta_t \phi_t(x_t, y) - \sum_{t=1}^T \theta_t \phi_t(x_t, y_t).$$

In this setup, we assume that at each period t , the decisions and actions (queries made to the function ϕ_t) of each player, i.e., x_t etc., are revealed to the other and vice versa *immediately after* they make their decision or action. This revealed information from period t can then be used by both players in their subsequent decisions and actions in the same period t or in future rounds $t+1$ and so on.

Let us now examine the affine regret associated with the monotone gradient operators of the functions ϕ_t , denoted by $F_t(x, y) = [\nabla_x \phi_t(x, y); -\nabla_y \phi_t(x, y)]$. More

² Note that the y -player receives a *concave reward* $\phi_t(x_t, y_t)$ at each time step, so their regret is written with the supremum.

precisely, let $z = [x; y]$ and $z_t = [x_t; y_t]$, and define $\xi_t = F_t(z_t)$. Then we have the following relation on the affine regret:

$$\begin{aligned}
& \sup_{z \in X \times Y} \sum_{t=1}^T \theta_t \langle \xi_t, z_t - z \rangle \\
&= \sup_{z \in X \times Y} \sum_{t=1}^T \theta_t (\langle \nabla_x \phi_t(x_t, y_t), x_t - x \rangle + \langle \nabla_y \phi_t(x_t, y_t), y - y_t \rangle) \\
&= \sup_{x \in X} \sum_{t=1}^T \theta_t \langle \nabla_x \phi_t(x_t, y_t), x_t - x \rangle + \sup_{y \in Y} \sum_{t=1}^T \theta_t \langle \nabla_y \phi_t(x_t, y_t), y - y_t \rangle \\
&\geq \sup_{x \in X} \sum_{t=1}^T \theta_t (\phi_t(x_t, y_t) - \phi_t(x, y_t)) + \sup_{y \in Y} \sum_{t=1}^T \theta_t (\phi_t(x_t, y) - \phi_t(x_t, y_t)) \\
&= \sum_{t=1}^T \theta_t \phi_t(x_t, y_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t \phi_t(x, y_t) + \sup_{y \in Y} \sum_{t=1}^T \theta_t \phi_t(x_t, y) - \sum_{t=1}^T \theta_t \phi_t(x_t, y_t) \\
&= \sup_{y \in Y} \sum_{t=1}^T \theta_t \phi_t(x_t, y) - \inf_{x \in X} \sum_{t=1}^T \theta_t \phi_t(x, y_t),
\end{aligned}$$

where the inequality follows from the convex-concave structure of $\phi_t(x, y)$ and the subgradient inequalities. Notice that the last line is simply the sum of both players' weighted regrets. Hence, minimizing the (weighted) affine regret of the gradient operators F_t results in minimizing the average social loss, i.e., the sum of the players' regrets. We refer to this sum as the *weighted online SP gap*, and call the problem of minimizing the weighted online SP gap the *online SP problem*. More precisely, the online SP gap problem seeks OCO algorithms to generate $[x_t; y_t]$ that minimize the weighted online SP gap

$$\sup_{y \in Y} \sum_{t=1}^T \theta_t \phi_t(x_t, y) - \inf_{x \in X} \sum_{t=1}^T \theta_t \phi_t(x, y_t) \leq r(T), \quad \lim_{T \rightarrow \infty} r(T) = 0. \quad (9)$$

When the functions ϕ_t remain the same throughout the time horizon, i.e., $\phi_t = \phi$ for all $t \in [T]$, and \bar{x}, \bar{y} are taken to be the weighted sums of $\{x_t\}_{t=1}^T, \{y_t\}_{t=1}^T$ respectively, the weighted online SP gap naturally bounds the standard SP gap for the underlying offline SP problem, i.e., $\epsilon_{\text{sad}}^\phi(\bar{x}, \bar{y})$ in (8).

An offline (online) SP problem can be solved by solving two related OCO problems, which can also be interpreted as two regret-minimizing players playing a static (dynamic) zero-sum game. Note that the reverse is not true in general: solving an offline (online) SP problem does not in general give us bounds on the individual regrets of each player.

The online SP gap interpretation of (9) is advantageous when we relax the non-anticipatory restriction. In an online setup where 1-lookahead decisions are allowed,

by examining specialized algorithms for minimizing the weighted online SP gap (9) rather than employing two separate regret-minimization algorithms for the players, we can exploit both the fact that our choices $[x_t; y_t]$ may utilize the current function ϕ_t and any favorable structural properties of the functions ϕ_t such as smoothness. In Sect. 3.3, we introduce algorithms that minimize the weighted online SP gap (9) directly. Our analysis demonstrates that exploiting favorable structural properties of functions ϕ_t plays a crucial role for obtaining better convergence rates for (9). See also Remark 12.

3 An algorithmic framework for online convex optimization

Many OCO algorithms are closely related to offline iterative FOMs. In this section, we first introduce some notation and key concepts related to the proximal setup for FOMs along with general properties of two classical FOMs, namely the *Mirror Descent* and *Mirror Prox* algorithms, that are crucial in our analysis for OCO. We then analyze the general versions of these FOMs to develop upper bounds on the weighted regret and weighted online SP gap. We follow the presentation and notation of the excellent survey [23, 24].

3.1 Proximal setup for the domains

Most FOMs capable of solving OCO and online SP problems are quite flexible in terms of adjusting to the geometry of the problem characterized by its domain \mathcal{Z} . In the case of SP problems, the domain is given by $\mathcal{Z} = X \times Y$ where (7) lives. The following components are standard in forming the setup for such FOMs and their convergence analysis:

- *Norm*: $\|\cdot\|$ on the Euclidean space \mathbb{E} where the domain \mathcal{Z} lives, along with its dual norm $\|\zeta\|_* := \max_{\|z\| \leq 1} \langle \zeta, z \rangle$.
- *Distance-Generating Function* (d.g.f.): A function $\omega(z) : \mathcal{Z} \rightarrow \mathbb{R}$, which is convex and continuous on \mathcal{Z} , and admits a selection of subgradients $\nabla\omega(z)$ that is continuous on the set $\mathcal{Z}^\circ := \{z \in \mathcal{Z} : \partial\omega(z) \neq \emptyset\}$ (here $\partial\omega(z)$ is a subdifferential of ω taken at z), and is strongly convex with modulus 1 with respect to $\|\cdot\|$:

$$\forall z', z'' \in \mathcal{Z}^\circ : \langle \nabla\omega(z') - \nabla\omega(z''), z' - z'' \rangle \geq \|z' - z''\|^2.$$

- *Bregman distance*: $V_z(z') := \omega(z') - \omega(z) - \langle \nabla\omega(z), z' - z \rangle$ for all $z \in \mathcal{Z}^\circ$ and $u \in \mathcal{Z}$.

Note that $V_z(z') \geq \frac{1}{2}\|z - z'\|^2 \geq 0$ for all $z \in \mathcal{Z}^\circ$ and $z' \in \mathcal{Z}$ follows from the strong convexity of ω .

- *Prox-mapping*: Given a *prox center* $z \in \mathcal{Z}^\circ$,

$$\text{Prox}_z(\xi) := \arg \min_{z' \in \mathcal{Z}} \{\langle \xi, z' \rangle + V_z(z')\} : \mathbb{E} \rightarrow \mathcal{Z}^\circ.$$

When the d.g.f. is taken as the squared ℓ_2 -norm, the prox mapping becomes the usual projection operation of the vector $z - \xi$ onto \mathcal{Z} .

- ω -center: $z_\omega := \arg \min_{z \in \mathcal{Z}} \omega(z)$.
- Set width: $\Omega = \Omega_z := \max_{z \in \mathcal{Z}} V_{z_\omega}(z) \leq \max_{z \in \mathcal{Z}} \omega(z) - \min_{z \in \mathcal{Z}} \omega(z)$.

For common domains \mathcal{Z} such as simplex, Euclidean ball, and spectahedron, standard proximal setups, i.e., selection of norm $\|\cdot\|$, d.g.f. $\omega(\cdot)$, the resulting Prox computations and set widths Ω are discussed in [23, Sect. 5.7].

When we have a decomposable domain $\mathcal{Z} = X \times Y$, we can build a proximal setup for \mathcal{Z} from the individual proximal setups on X and Y . Given a norm $\|\cdot\|_x$ and a d.g.f. $\omega_x(\cdot)$ for the domain X , similarly $\|\cdot\|_y$, $\omega_y(\cdot)$ for the domain Y , and two scalars $\beta_x, \beta_y > 0$, we build the d.g.f. $\omega(z)$ and ω -center z_ω for $\mathcal{Z} = X \times Y$ as

$$\omega(z) = \beta_x \omega_x(x) + \beta_y \omega_y(y) \quad \text{and} \quad z_\omega = [x_{\omega_x}; y_{\omega_y}],$$

where $\omega_x(\cdot)$ and $\omega_y(\cdot)$ as well as x_{ω_x} and y_{ω_y} are customized based on the geometry of the domains X and Y . In this construction, the flexibility in determining the scalars $\beta_x, \beta_y > 0$ is useful in optimizing the overall convergence rate. Moreover, by letting $\xi = [\xi_x; \xi_y]$ and $z = [x; y]$, the prox mapping becomes decomposable as

$$\text{Prox}_z(\xi) = \left[\text{Prox}_x^{\omega_x}\left(\frac{\xi_x}{\beta_x}\right); \text{Prox}_y^{\omega_y}\left(\frac{\xi_y}{\beta_y}\right) \right],$$

where $\text{Prox}_x^{\omega_x}(\cdot)$ and $\text{Prox}_y^{\omega_y}(\cdot)$ are respectively prox mappings with respect to $\omega_x(\cdot)$ in domain X and $\omega_y(\cdot)$ in domain Y . We refer the reader to the references [23, Sect. 5.7.2] and [24, Sect. 6.3.3] for further details on how to optimally choose the parameters β_x, β_y for SP problems.

3.2 Non-smooth convex functions

In the most basic setup, our functions f_t (resp. ϕ_t) are convex (resp. convex-concave) and non-smooth. In this case, we analyze a generalization of Mirror Descent, outlined in Algorithm 1 for bounding the weighted regret and weighted online SP gap.

Algorithm 1 Generalized Mirror Descent

```

input:  $\omega$ -center  $z_\omega$ , time horizon  $T$ , positive step sizes  $\{\gamma_t\}_{t=1}^T$ , and a sequence  $\{\xi_t\}_{t=1}^T$ .
output: sequence  $\{z_t\}_{t=1}^T$ .
 $z_1 := z_\omega$ .
for  $t = 1, \dots, T$  do
     $z_{t+1} = \text{Prox}_{z_t}(\gamma_t \xi_t)$ 
end for

```

Remark 2 In Algorithm 1, computation of z_t depends on only z_{t-1} and ξ_{t-1} . In the following we will examine Algorithm 1 by allowing ξ_{t-1} to depend on only the

past information on functions f_1, \dots, f_{t-1} (or $\phi_1, \dots, \phi_{t-1}$). Then the iterations in Algorithm 1 will be based on solely the past information allowing us to carry out a *non-anticipatory* analysis for Algorithm 1. \square

Proposition 1 describes a fundamental property exhibited by the Mirror Descent updates. Its proof can be found in [23, Proposition 5.1, Equation 5.13].

Proposition 1 Suppose that the sequence of vectors $\{z_t\}_{t=1}^T$ is generated by Algorithm 1 for a given sequence of vectors $\{\xi_t\}_{t=1}^T$ and step sizes $\gamma_t > 0$ for $t \in [T]$. Then for any $z \in \mathcal{Z}$ and $t \in [T]$, we have

$$\gamma_t \langle \xi_t, z_t - z \rangle \leq V_{z_t}(z) - V_{z_{t+1}}(z) + \frac{1}{2} \gamma_t^2 \|\xi_t\|_*^2. \quad (10)$$

3.2.1 Weighted regret

From Proposition 1, we may derive a bound on the weighted regret (6) in the most general case where our functions $f_t(x)$ need only satisfy convexity and Lipschitz continuity. More precisely, we will assume the following.

Assumption 1 A proximal setup of Sect. 3.1 exists for the domain $\mathcal{Z} = X$. Each function f_t is convex, and there exists $G \in (0, \infty)$ such that the subgradients of f_t are bounded, i.e., $\|\nabla f_t(x)\|_* \leq G$ for all $x \in X$ and $t \in [T]$.

Theorem 1 Suppose Assumption 1 holds, and we are given weights $\theta \in \Delta_T$. Then running Algorithm 1 with $z_t = x_t$, $\xi_t = \theta_t \nabla f_t(x_t)$, and step sizes $\gamma_t = \gamma := \sqrt{\frac{2\Omega}{\sup_{t \in [T]} \theta_t^2 G^2 T}}$ for all $t \in [T]$ results in

$$\sum_{t=1}^T \theta_t f_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t f_t(x) \leq \sqrt{2\Omega \left(\sup_{t \in [T]} \theta_t^2 \right) G^2 T}.$$

Note that Theorem 1 is a simple generalization of the fundamental result of [39]. We include its proof for completeness.

Proof By summing up (10) for $t \in [T]$ and writing $\gamma_t = \gamma$ as a constant we obtain

$$\begin{aligned} \sum_{t=1}^T \gamma_t \langle \xi_t, x_t - x \rangle &= \gamma \sum_{t=1}^T \theta_t \langle \nabla f_t(x_t), x_t - x \rangle \leq V_{x_1}(x) - V_{x_{T+1}}(x) \\ &\quad + \frac{\gamma^2}{2} \sum_{t=1}^T \theta_t^2 \|\nabla f_t(x_t)\|_*^2. \end{aligned}$$

Because $\|\theta_t \nabla f_t(x_t)\|_* \leq \theta_t G \leq (\sup_{t \in T} \theta_t) G$, $V_{x_1}(x) \leq \Omega$ by our choice of x_1 in Algorithm 1, $-V_{x_{T+1}}(x) \leq 0$, and dividing through by γ , we reach to

$$\sum_{t=1}^T \theta_t \langle \nabla f_t(x_t), x_t - x \rangle \leq \frac{\Omega}{\gamma} + \frac{\gamma}{2} \left(\sup_{t \in [T]} \theta_t^2 \right) G^2 T.$$

Optimizing the right hand side over $\gamma \geq 0$ gives us the desired upper bound of $\sqrt{2\Omega (\sup_{t \in [T]} \theta_t^2)} G^2 T$. The left hand side of inequality in the theorem follows from $\theta_t \geq 0$ for all $t \in [T]$ and the convexity of functions f_t implying for all $x \in X$

$$\langle \xi_t, x_t - x \rangle = \theta_t \langle \nabla f_t(x_t), x_t - x \rangle \geq \theta_t f_t(x_t) - \theta_t f_t(x).$$

□

The bound on weighted regret in Theorem 1 is optimized when the convex combination weights $\theta \in \Delta_T$ are set to be *uniform*, i.e., $\theta_t = 1/T$; in this case, the right hand side of the inequality becomes $O(1/\sqrt{T})$.

Remark 3 We would like to highlight the importance of customizing our proximal setup based on the geometry of the domain. In many cases, weighted regret or weighted online SP gap bounds have a dependence on the set width parameter Ω associated with the proximal setup; see e.g., Theorem 1. For example, when our domain $X = \Delta_n$, equipping X with a proximal setup based on negative entropy d.g.f. $\omega(x) = \sum_{j=1}^n x^{(j)} \log(x^{(j)})$ results in $\Omega = \log(n)$, which is almost dimension independent. Using the Euclidean d.g.f. $\omega(x) = \frac{1}{2}\langle x, x \rangle$ on $X = \Delta_n$ leads to a suboptimal (and dimension-dependent) set width of $\Omega = \sqrt{n}$. Moreover, certain domains admit d.g.f.s that lead to quite efficient Prox computations given either in closed form or by simple computations, taking only $O(n)$ arithmetic operations. Negative entropy d.g.f. over simplex and Euclidean d.g.f. over the Euclidean unit ball are such examples. A possible issue for equipping the simplex with a Euclidean proximal setup is that the prox-mapping (usual projection) no longer has a closed form, but it still can be done efficiently in $O(n \log(n))$ arithmetic operations. See [23] for a complete discussion.

□

3.2.2 Exploiting strong convexity

When our functions f_t admit further favorable structure in the form of strong convexity, it is possible to customize Algorithm 1 using specific *nonuniform* weights θ_t and achieve a bound of $O(1/T)$, which is significantly better than the standard $O(1/\sqrt{T})$ bound of Theorem 1 given by uniform weights. Our developments here are based on the following structural assumption.

Assumption 2

- A proximal setup of Sect. 3.1 exists for the domain $\mathcal{Z} = X$.
- The loss functions $f_t(x)$ for $t \in [T]$ have the property that the functions $f_t(x) - \alpha \omega(x)$ is convex for some $\alpha > 0$ independent of t , or equivalently

$$f_t(x) \leq f_t(x') + \langle \nabla f_t(x), x - x' \rangle - \alpha V_x(x'), \quad \forall x, x' \in X, t \in [T].$$

- The subgradients of the loss functions are bounded, i.e., there exists $G \in (0, \infty)$ such that $\|\nabla f_t(x)\|_* \leq G$ for all $x \in X$, $t \in [T]$.

Remark 4 When our proximal setup for X is based on a Euclidean d.g.f. $\omega(x) = \frac{1}{2}\langle x, x \rangle$ and Euclidean norm $\|x\|_2$, then Assumption 2 simply states that the functions f_t are α -strongly convex. In this paper, we will abuse terminology slightly and say that f_t is α -strongly convex when $f_t(x) - \alpha \omega(x)$ is convex, where the dependence on the d.g.f. ω will be clear from the context. \square

In Eq. (5), we demonstrated that the weighted regret of a sequence of functions and points $\{f_t, x_t\}_{t=1}^T$ can be upper bounded by a affine regret term with loss vectors $\xi_t = \nabla f_t(x_t)$. Under Assumption 2, we can improve this upper bound via the following lemma.

Lemma 1 Suppose that for $t \in [T]$, the loss functions f_t satisfy Assumption 2. Given a sequence $\{x_t\}_{t=1}^T$, define $q_t(x) = \langle \nabla f_t(x_t), x \rangle + \alpha V_{x_t}(x)$. Then the weighted regret of the sequence $\{x_t\}_{t=1}^T$ on the functions f_t can be bounded by the weighted regret of the same sequence on the functions q_t :

$$\begin{aligned} \sum_{t=1}^T \theta_t f_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t f_t(x) &\leq \sup_{x \in X} \sum_{t=1}^T \theta_t (\langle \nabla f_t(x_t), x_t - x \rangle - \alpha V_{x_t}(x)) \\ &= \sum_{t=1}^T \theta_t q_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t q_t(x). \end{aligned} \quad (11)$$

Proof Assumption 2 implies that $f_t(x) - \alpha \omega(x)$ is convex, and thus the inequality holds. The equality holds since $V_{x_t}(x_t) = 0$. \square

Notice that since $V_{x_t}(x) \geq 0$, (11) is an improvement on the affine regret bound of (5). By Lemma 1, in order to bound the weighted regret, it suffices to bound the right hand term of (11). By selecting the step sizes γ_t and weights θ_t in a clever fashion, we are able to exploit the extra $-\alpha V_{x_t}(x)$ terms to improve the regret bound. This result is a generalization of the offline stochastic gradient descent algorithm equipped with a Euclidean d.g.f. based proximal setup presented in Lacoste-Julien et al. [26] to the online setting with domain X admitting a general proximal setup.

Theorem 2 Suppose Assumption 2 holds. Fix a set of convex combination weights $\theta_t = \frac{2t}{T(T+1)}$ for $t \in [T]$. Then running Algorithm 1 with $z_t = x_t$, $\xi_t = \nabla f_t(x_t)$, and step sizes $\gamma_t = \frac{2}{\alpha(t+1)}$ for all $t \in [T]$ results in

$$\begin{aligned} \sum_{t=1}^T \theta_t f_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t f_t(x) &\leq \sup_{x \in X} \sum_{t=1}^T \theta_t (\langle \nabla f_t(x_t), x_t - x \rangle - \alpha V_{x_t}(x)) \\ &\leq \frac{2G^2}{\alpha(T+1)}. \end{aligned}$$

Proof By Lemma 1, the first inequality holds, so we focus on the second. Proposition 1 gives us the following inequality for all $x \in X$

$$\begin{aligned}\gamma_t \langle \xi_t, x_t - x \rangle &= \gamma_t \langle \nabla f_t(x_t), x_t - x \rangle \leq V_{x_t}(x) - V_{x_{t+1}}(x) + \frac{\gamma_t^2}{2} \|\xi_t\|_*^2 \\ &= V_{x_t}(x) - V_{x_{t+1}}(x) + \frac{\gamma_t^2}{2} \|\nabla f_t(x_t)\|_*^2.\end{aligned}$$

This, along with $\|\nabla f_t(x_t)\|_* \leq G$ implies

$$\langle \nabla f_t(x_t), x_t - x \rangle - \alpha V_{x_t}(x) \leq \frac{1}{\gamma_t} V_{x_t}(x) - \frac{1}{\gamma_t} V_{x_{t+1}}(x) - \alpha V_{x_t}(x) + \frac{\gamma_t G^2}{2}. \quad (12)$$

Multiplying (12) by θ_t and summing over $t \in [T]$ establishes the inequalities below.

$$\begin{aligned}&\sum_{t=1}^T \theta_t (\langle \nabla f_t(x_t), x_t - x \rangle - \alpha V_{x_t}(x)) \\ &\leq \sum_{t=1}^T \theta_t \left(\frac{1}{\gamma_t} V_{x_t}(x) - \frac{1}{\gamma_t} V_{x_{t+1}}(x) - \alpha V_{x_t}(x) + \frac{\gamma_t G^2}{2} \right).\end{aligned}$$

Now, when $\gamma_t = \frac{2}{\alpha(t+1)}$, we arrive at

$$\begin{aligned}\frac{1}{\gamma_t} V_{x_t}(x) - \frac{1}{\gamma_t} V_{x_{t+1}}(x) - \alpha V_{x_t}(x) + \frac{\gamma_t G^2}{2} &= \frac{\alpha(t-1)}{2} V_{x_t}(x) - \frac{\alpha(t+1)}{2} V_{x_{t+1}}(x) \\ &\quad + \frac{G^2}{\alpha(t+1)}.\end{aligned}$$

Multiplying this by t gives us

$$\begin{aligned}t \left(\frac{1}{\gamma_t} V_{x_t}(x) - \frac{1}{\gamma_t} V_{x_{t+1}}(x) - \alpha V_{x_t}(x) + \frac{\gamma_t G^2}{2} \right) &\leq \frac{\alpha(t-1)t}{2} V_{x_t}(x) \\ &\quad - \frac{\alpha t(t+1)}{2} V_{x_{t+1}}(x) + \frac{G^2}{\alpha}.\end{aligned}$$

After summing this over $t \in [T]$ and noting that the first two terms telescope, the coefficient in front of $V_{x_1}(x)$ is zero, and $V_{x_{T+1}}(x) \geq 0$, we deduce

$$\begin{aligned}\sum_{t=1}^T t \left(\frac{1}{\gamma_t} V_{x_t}(x) - \frac{1}{\gamma_t} V_{x_{t+1}}(x) - \alpha V_{x_t}(x) + \frac{\gamma_t G^2}{2} \right) &\leq \frac{G^2 T}{\alpha} - \frac{\alpha T(T+1)}{2} V_{x_{T+1}}(x) \\ &\leq \frac{G^2 T}{\alpha}.\end{aligned}$$

Dividing both sides of this inequality by $\frac{T(T+1)}{2}$ leads to

$$\sum_{t=1}^T \theta_t \left(\frac{1}{\gamma_t} V_{x_t}(x) - \frac{1}{\gamma_t} V_{x_{t+1}}(x) - \alpha V_{(x_t)}(x) + \frac{\gamma_t G^2}{2} \right) \leq \frac{2G^2}{\alpha(T+1)},$$

which establishes the second inequality. \square

Let us revisit Remark 3 on customizing the proximal setup based on the geometry of the domain.

Remark 5 In contrast to Theorem 1, the bound of Theorem 2 has no dependence on set width Ω . Nevertheless, customization of the proximal setup, in particular selection of d.g.f. ω plays an important role in Theorem 2 through Assumption 2. In many cases, it is much more likely to encounter functions f_t that are α -strongly convex in the usual sense, i.e., $f_t(x) - \alpha \|x\|_2^2/2$ is convex, but it may not be possible to ensure the convexity of $f_t(x) - \alpha \omega(x)$ with respect to a different d.g.f. ω . In such cases, it is possible (and more desirable) to select a d.g.f. ω that will ensure that the strong convexity requirement of Assumption 2 is satisfied. Because the bound of Theorem 2 has no dependence on Ω , such a selection of ω will not adversely affect overall the weighted regret bound of Theorem 2. \square

Remark 6 For strongly convex losses, Theorem 2 establishes an upper bound of $O(1/T)$ on weighted regret. In contrast to this, Hazan and Kale [18] established a lower bound of $O(\log(T)/T)$ for minimizing standard regret in OCO with strongly convex loss functions. The main distinguishing feature of [18] and our result in Theorem 2 is that while [18] considers the case of using uniform weights $\theta_t = 1/T$ only, we are allowed to use nonuniform (in fact increasing) weights $\theta_t = 2t/(T^2 + T)$. The faster rate of $O(1/T)$ in Theorem 2 is a result of this flexibility in our setup due to the weighted regret concept that lets us choose nonuniform weights. \square

3.2.3 Weighted online SP gap

Algorithm 1 can also be utilized in bounding the weighted online SP gap (9). In this case, in addition to a convex-concave structure assumption on functions $\phi_t(x, y)$, we assume boundedness of specific monotone gradient operators associated with $\phi_t(x, y)$.

Assumption 3 A proximal setup of Sect. 3.1 exists for the domain $\mathcal{Z} = X \times Y$. Each function $\phi_t(x, y)$ is convex in x and concave in y , and there exists $G \in (0, \infty)$ such that $\|[\nabla_x \phi_t(x, y); -\nabla_y \phi_t(x, y)]\|_* \leq G$ for all $x \in X, y \in Y$ and $t \in [T]$.

Theorem 3 Suppose Assumption 3 holds, and we are given convex combination weights $\theta \in \Delta_T$. Then running Algorithm 1 with $z_t = [x_t; y_t]$, $\xi_t = \theta_t [\nabla_x \phi_t(x_t, y_t); -\nabla_y \phi_t(x_t, y_t)]$, and step sizes $\gamma_t = \sqrt{\frac{2\Omega}{\sup_{t \in [T]} \theta_t^2 G^2 T}}$ for all $t \in [T]$ gives us

$$\sup_{y \in Y} \sum_{t=1}^T \theta_t \phi_t(x_t, y) - \inf_{x \in X} \sum_{t=1}^T \theta_t \phi_t(x, y_t) \leq \sqrt{2\Omega \left(\sup_{t \in [T]} \theta_t^2 \right) G^2 T}$$

Proof The proof proceeds exactly as the proof of Theorem 1 to arrive at

$$\sum_{t=1}^T \langle \xi_t, z_t - z \rangle \leq \sqrt{2\Omega \left(\sup_{t \in [T]} \theta_t^2 \right) G^2 T}$$

for all $z = [x; y] \in X \times Y$. Then, from the convex-concave structure of the function ϕ_t , we have for all $z = [x; y] \in X \times Y$ and all $t \in [T]$,

$$\begin{aligned} \langle \xi_t, z_t - z \rangle &= \theta_t \langle \nabla_x \phi_t(x_t, y_t), x_t - x \rangle + \theta_t \langle \nabla_y \phi_t(x_t, y_t), y - y_t \rangle \\ &\geq \theta_t (\phi_t(x_t, y_t) - \phi_t(x, y_t)) + \theta_t (\phi_t(x_t, y) - \phi_t(x_t, y_t)) \\ &= \theta_t \phi_t(x_t, y) - \theta_t \phi_t(x, y_t). \end{aligned}$$

The result then follows by combining the inequality above with the inequality that provides the upper bound on the term $\sum_{t=1}^T \langle \xi_t, z_t - z \rangle$. \square

Remark 7 Uniform weights $\theta_t = 1/T$ minimize $\sup_{t \in [T]} \theta_t$ and result in a regret (online SP gap) bound of $O(1/\sqrt{T})$ in Theorem 1 (Theorem 3). Moreover, Theorems 1 and 3 can accommodate a variety of convex combination weights $\theta \in \Delta_T$ via adapting their step sizes γ_t and still achieve bounds of form $O(1/\sqrt{T})$. For example, this is the case when the nonuniform weights $\theta_t = 2t/(T^2 + T)$ from Theorem 2 are used in these results. Employing nonuniform weights becomes more consequential when we have to run several OCO or online SP algorithms in conjunction with each other using the *same* weights θ_t in all of them. Such a situation arises in solving robust feasibility problems, which we discuss in Sect. 4. \square

3.3 Exploiting lookahead and smoothness

In offline convex optimization when minimizing a *smooth* convex function over a convex domain, the Mirror Prox algorithm of [29] admits a better convergence rate than Mirror Descent and is thus preferable. In this section we demonstrate that the same improvement is also attainable in an online setting when our functions exhibit a smooth structure and our setting allows for *1-lookahead*—that is, we are allowed to a *limited* query access to our current function f_t at time period t before we make our decision z_t . In fact, we query f_t only once in each period t .

As discussed in the Introduction, 1-lookahead setting may prevent it being applicable in certain online settings. In addition, if at iteration t we are given multiple query access to f_t (or ϕ_t), we can *guarantee* that the weighted regret (online SP gap) will be non-positive by directly minimizing f_t (solving for the SP of ϕ_t). However, solving a complete optimization problem at each iteration may be expensive, and hence even in the situations where we have multiple query access to f_t at iteration t , it may be

preferable to use our more efficient methods to bound the weighted regret (online SP gap). We present an example of such a situation, solving robust feasibility problems, in the next section.

Our analysis is based on the generalization of Mirror Prox outlined in Algorithm 2.

Algorithm 2 Generalized Mirror Prox

```

input:  $\omega$ -center  $z_\omega$ , time horizon  $T$ , positive step sizes  $\{\gamma_t\}_{t=1}^T$ , and sequences  $\{\eta_t, \xi_t\}_{t=1}^T$ .
output: sequence  $\{z_t\}_{t=1}^T$ .
 $v_1 := z_\omega$ 
for  $t = 1, \dots, T$  do
     $z_t = \text{Prox}_{v_t}(\gamma_t \eta_t)$ .
     $v_{t+1} = \text{Prox}_{v_t}(\gamma_t \xi_t)$ .
end for
```

Proposition 2 states a fundamental property of Mirror Prox updates which is instrumental in the derivation of our bounds; see [24, Lemma 6.2] for its proof.

Proposition 2 Suppose that the sequences of vectors $\{v_t, z_t\}_{t=1}^T$ are generated by Algorithm 2 for the given sequences $\{\eta_t, \xi_t\}_{t=1}^T$ and step sizes $\gamma_t > 0$ for $t \in [T]$. Then for any $z \in \mathcal{Z}$ and $t \in [T]$, we have

$$\gamma_t \langle \xi_t, z_t - z \rangle \leq V_{v_t}(z) - V_{v_{t+1}}(z) + \frac{1}{2} \left(\gamma_t^2 \|\xi_t - \eta_t\|_*^2 - \|z_t - v_t\|^2 \right).$$

We analyze Algorithm 2 under the following smoothness assumption and derive an improved rate of convergence for minimizing weighted regret.

Assumption 4 A proximal setup of Sect. 3.1 exists for the domain $\mathcal{Z} = X$. Each function $f_t(x)$ is convex in x , and there exists $L \in (0, \infty)$ such that $\|\nabla f_t(x) - \nabla f_t(v)\|_* \leq L \|x - v\|$ holds for all $x, v \in X$ and all $t \in [T]$.

Theorem 4 Suppose Assumption 4 holds, and we are given weights $\theta \in \Delta_T$. Then running Algorithm 2 with $z_t = x_t$, $\eta_t = \theta_t \nabla f_t(v_t)$, $\xi_t = \theta_t \nabla f_t(z_t)$, and step sizes $\gamma_t = \frac{1}{(L \sup_{t \in T} \theta_t)}$ for all $t \in [T]$ leads to

$$\sum_{t=1}^T \theta_t f_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t f_t(x) \leq \mathcal{Q} L \sup_{t \in [T]} \theta_t.$$

Proof From Assumption 4, we have for all $t \in [T]$

$$\|\xi_t - \eta_t\|_* = \theta_t \|\nabla f_t(x_t) - \nabla f_t(v_t)\|_* \leq L \theta_t \|x_t - v_t\| \leq L \sup_{t \in [T]} \theta_t \|x_t - v_t\|.$$

Thus, by setting $\gamma_t = \frac{1}{(L \sup_{t \in [T]} \theta_t)}$, we deduce $\gamma_t^2 \|\xi_t - \eta_t\|_*^2 - \|x_t - v_t\|^2 \leq 0$ for all $t \in [T]$. Then from Proposition 2 we obtain for all $x \in X$ and $t \in [T]$

$$\langle \xi_t, x_t - x \rangle = \theta_t \langle \nabla f_t(x_t), x_t - x \rangle \leq (V_{v_t}(x) - V_{v_{t+1}}(x)) L \sup_{t \in [T]} \theta_t.$$

Summing this inequality over $t \in [T]$ and using $V_{v_1}(x) \leq \Omega$, $V_{v_{T+1}}(x) \geq 0$, we get

$$\sum_{t=1}^T \langle \xi_t, x_t - x \rangle = \sum_{t=1}^T \theta_t \langle \nabla f_t(x_t), x_t - x \rangle \leq \Omega L \sup_{t \in [T]} \theta_t.$$

The result then follows from convexity of f_t and using the subgradient inequality $\langle \nabla f_t(x_t), x_t - x \rangle \geq f_t(x_t) - f_t(x)$. \square

A similar result holds for the online SP gap under the following analogous smoothness assumption.

Assumption 5 A proximal setup of Sect. 3.1 exists for the domain $\mathcal{Z} = X \times Y$, and we denote $z = [x; y]$. Each function $\phi_t(x, y)$ is convex in x and concave in y . Denoting $F_t(z) = [\nabla_x \phi_t(x, y); -\nabla_y \phi_t(x, y)]$, there exists $L \in (0, \infty)$ such that for all $v, z \in \mathcal{Z}$ and all $t \in [T]$, we have

$$\|F_t(z) - F_t(v)\|_* \leq L \|z - v\|.$$

Remark 8 A sufficient condition for the Lipschitz continuity of monotone gradient operators F_t of Assumption 5 is Lipschitz continuity of their partial subgradients. For brevity, we omit the proof of this; see [24, 29] for further details. \square

Theorem 5 Suppose Assumption 5 holds, and we are given weights $\theta \in \Delta_T$. Then running Algorithm 2 with $z_t = [x_t; y_t]$, $\eta_t = \theta_t F_t(v_t)$, $\xi_t = \theta_t F_t(z_t)$, and step sizes $\gamma_t = \frac{1}{(L \sup_{t \in T} \theta_t)}$ for all $t \in [T]$ leads to

$$\sup_{y \in Y} \sum_{t=1}^T \theta_t \phi_t(x_t, y) - \inf_{x \in X} \sum_{t=1}^T \theta_t \phi_t(x, y_t) \leq \Omega L \sup_{t \in [T]} \theta_t.$$

Proof Following the outline of the proof of Theorem 4, we obtain

$$\sum_{t=1}^T \langle \xi_t, z_t - z \rangle = \sum_{t=1}^T \theta_t \langle F_t(z_t), z_t - z \rangle \leq \Omega L \sup_{t \in [T]} \theta_t$$

for all $z = [x; y] \in X \times Y$. As in the proof of Theorem 3, using the convex-concave structure of the functions ϕ_t , we arrive at

$$\theta_t \langle F_t(z_t), z_t - z \rangle \geq \theta_t \phi_t(x_t, y) - \theta_t \phi_t(x, y_t),$$

which establishes the result. \square

Remark 9 As discussed in Remark 7, when the convex combination weights θ_t are set to be either uniform weights $\theta_t = 1/T$ or nonuniform weights $\theta_t = 2t/(T^2 + T)$ from Theorem 2, we have $\sup_{t \in [T]} \theta_t = O(1/T)$, and thus we achieve a better weighted regret (online SP gap) bound of $O(1/T)$ in Theorem 4 (Theorem 5) than the $O(1/\sqrt{T})$ bound of Theorem 1 (Theorem 3). \square

There is a fundamental distinction between Algorithms 1 and 2 in terms of their anticipatory/non-anticipatory behavior. This distinction between anticipatory/non-anticipatory behavior is important in the context of using these algorithms for coupled optimization problems. We discuss this next.

Remark 10 When Algorithm 2 is utilized in Theorems 4 and 5, at step t , in order to compute the decision $z_t = \text{Prox}_{v_t}(\gamma_t \eta_t)$, where $v_t \in \mathcal{Z}$ is a point computed in the previous step, we utilize the knowledge of the current function f_t or ϕ_t because $\eta_t = \theta_t \nabla f_t(v_t)$ or $\eta_t = \theta_t F_t(v_t)$. Therefore, Algorithm 2 is categorized as *1-lookahead* or *anticipatory*. This is in contrast to the non-anticipatory nature of Algorithm 1 analyzed in Theorems 1, 2, and 3, where computing $z_t = \text{Prox}_{z_{t-1}}(\gamma_{t-1} \xi_{t-1})$ only required knowledge of the previous step $t-1$ because ξ_{t-1} was determined based on only $\nabla f_{t-1}(z_{t-1})$ or $F_{t-1}(z_{t-1})$. \square

Remark 11 Rakhlin and Sridharan [33, 34] also explore OCO with anticipatory decisions through the lens of *predictable sequences* $\{M_t\}_{t=1}^T$. More precisely, they also examine how regret bounds are affected when the player is allowed to utilize side information M_t before choosing x_t at time t . They propose the Optimistic Mirror Descent (OpMD) algorithm, which is a special case of Algorithm 2 for $\eta_t = M_t$, $\xi_t = \nabla f_t(z_t)$ and $\theta_t = 1/T$, and are able to recover the offline Mirror Prox algorithm from [29] for smooth offline convex optimization and smooth offline SP problems. In fact, our results in Theorem 4 and Theorem 5 can be derived from [34, Lemma 1] by specifying the predictable sequences $M_t = \theta_t \nabla f_t(v_t)$ and $M_t = \theta_t F_t(v_t)$ respectively. Here, we allow the player to have access *only to gradient information* of f_t or ϕ_t at time t . Because the focus of [33, 34] was different, the observation that the OpMD algorithm can obtain faster $O(1/T)$ convergence rates in the 1-lookahead setting was not made before. \square

Remark 12 It is known that the OCO regret bounds with general smooth loss functions have a lower bound complexity of at least $O(1/\sqrt{T})$ (this holds even for the case of linear loss functions [1, Theorem 5]). This is in contrast to the faster rate of $O(1/T)$ established in Theorem 4. The lookahead nature of our analysis of Algorithm 2 discussed in Remark 10 plays a crucial role for achieving the speedup established in Theorem 4. \square

4 Application: robust optimization

In this section, we apply our developments on OCO to solving the robust optimization (RO) problem (2). Instead of solving (2) directly, we examine the associated *robust feasibility problem*: given desired accuracy $\epsilon > 0$,

$$\begin{cases} \text{Either : find } x \in X \text{ s.t. } \sup_{u^i \in U^i} f^i(x, u^i) \leq \epsilon \quad \forall i \in [m]; \\ \text{or : declare infeasibility, } \forall x \in X, \exists i \in [m] \text{ s.t. } \sup_{u^i \in U^i} f^i(x, u^i) > 0. \end{cases} \quad (13)$$

We note that optimizing an objective function $f(x)$ via feasibility oracle (13) will incur only an extra $\log(1/\epsilon)$ multiplicative factor in the number of iterations. This approximate ϵ -feasibility problem is motivated by how most convex optimization solvers certify their solutions. We are interested in the number of iterations needed to solve (13), which will depend on the accuracy parameter ϵ .

It was established in [19] that under the basic convexity assumptions, (13) can be solved by standard OCO algorithms achieving $O(1/\epsilon^2)$ convergence rate and requiring only basic arithmetic operations and subgradient computations in each iteration. In this section, we examine how our regret bounds from Sect. 3 can improve the $O(1/\epsilon^2)$ convergence rate for (13) under certain structural assumptions on the constraint functions f^i . We first define some notation. We denote $u := [u^1; \dots; u^m]$, $U = U^1 \times \dots \times U^m$ and $Y := \Delta_m$. Given sequences $x_t \in X$, $u_t \in U$, $y_t \in Y$ for $t \in [T]$ and weights $\theta \in \Delta_T$, we define

$$\begin{aligned} \epsilon^\circ(\{x_t, u_t, \theta_t\}_{t=1}^T) &:= \max_{i \in [m]} \left\{ \sup_{u^i \in U^i} \sum_{t=1}^T \theta_t f^i(x_t, u^i) - \sum_{t=1}^T \theta_t f^i(x_t, u_t^i) \right\}, \\ \epsilon^\bullet(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T) &:= \max_{i \in [m]} \sum_{t=1}^T \theta_t f^i(x_t, u_t^i) - \inf_{x \in X} \sum_{t=1}^T \theta_t \sum_{i=1}^m y_t^{(i)} f^i(x, u_t^i), \quad \text{and} \\ \epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T) &:= \sum_{t=1}^T \theta_t \max_{i \in [m]} f^i(x_t, u_t^i) - \inf_{x \in X} \sum_{t=1}^T \theta_t \max_{i \in [m]} f^i(x, u_t^i). \end{aligned}$$

The following results from [19] states how (13) can be verified in an iterative fashion.

Theorem 6 ([19, Theorem 3.2, Corollary 3.1]) *Let $x_t \in X$, $u_t \in U$, $y_t \in \Delta_m$ for $t \in [T]$, $\theta \in \Delta_T$, and $\tau \in (0, 1)$. If $\epsilon^\circ(\{x_t, u_t, \theta_t\}_{t=1}^T) \leq \tau\epsilon$ and $\max_{i \in [m]} \sum_{t=1}^T \theta_t f^i(x_t, u_t^i) \leq (1 - \tau)\epsilon$, then the solution $\bar{x}_T := \sum_{t=1}^T \theta_t x_t$ is ϵ -feasible with respect to (13). If $\epsilon^\bullet(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T) \leq (1 - \tau)\epsilon$ and $\max_{i \in [m]} \sum_{t=1}^T \theta_t f^i(x_t, u_t^i) > (1 - \tau)\epsilon$, then (13) is infeasible. When all but $\{y_t\}_{t=1}^T$ is given, there exists an appropriate choice of $y_t \in \Delta_m$ such that $\epsilon^\bullet(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T) \leq \epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T)$. Thus, if $\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T) \leq (1 - \tau)\epsilon$ and $\max_{i \in [m]} \sum_{t=1}^T \theta_t f^i(x_t, u_t^i) > (1 - \tau)\epsilon$, then (13) is infeasible.*

Thus, solving the robust feasibility problem (13) reduces to bounding $\epsilon^\circ(\{x_t, u_t, \theta_t\}_{t=1}^T)$ and $\epsilon^\bullet(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T)$ (or $\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T)$), and then evaluating $\max_{i \in [m]} \sum_{t=1}^T \theta_t f^i(x_t, u_t^i)$. Intuitively, robust feasibility can be seen as a two-player zero sum game, where one player chooses the $\{u_t\}_{t=1}^T$ and the other player chooses $\{x_t\}_{t=1}^T$. We can think of $\epsilon^\circ(\{x_t, u_t, \theta_t\}_{t=1}^T)$ and $\epsilon^\bullet(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T)$ (or $\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T)$) as approximations to the regret of each player.

We first discuss how to bound these terms *individually*. After that, we discuss how to combine these bounds properly to solve (13) by taking into account the common weights $\theta \in \Delta_T$ and any non-anticipatory/lookahead properties of the algorithms.

Observation 1 Given a sequence $\{x_t\}_{t=1}^T$, define the functions $f_t^i(u^i) := -f^i(x_t, u^i)$. Then the term $\epsilon^\circ(\{x_t, u_t, \theta_t\}_{t=1}^T)$ can be written as the maximum of weighted regret terms (6) with the functions f_t^i and weights $\theta \in \Delta_T$ over the sequences $\{u_t^i\}_{t=1}^T$:

$$\epsilon^\circ(\{x_t, u_t, \theta_t\}_{t=1}^T) = \max_{i \in [m]} \left\{ \sum_{t=1}^T \theta_t f_t^i(u_t^i) - \inf_{u^i \in U^i} \sum_{t=1}^T \theta_t f_t^i(u^i) \right\}.$$

Given a sequence $\{u_t\}_{t=1}^T$, define the functions $\phi_t(x, y) := \sum_{i=1}^m y^{(i)} f^i(x, u_t^i)$. Then $\epsilon^\bullet(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T)$ can be written as a weighted online saddle point gap term (8) with functions ϕ_t and weights $\theta \in \Delta_T$ over the sequence $\{x_t, y_t\}_{t=1}^T$:

$$\epsilon^\bullet(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T) = \max_{y \in Y} \sum_{t=1}^T \theta_t \phi_t(x_t, y) - \inf_{x \in X} \sum_{t=1}^T \theta_t \phi_t(x, y_t).$$

Furthermore, let $h_t(x) := \max_{i \in [m]} f^i(x, u_t^i)$. Then $\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T)$ can be written as a weighted regret term (6) with functions h_t and weights $\theta \in \Delta_T$ over the sequence $\{x_t\}_{t=1}^T$:

$$\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T) = \sum_{t=1}^T \theta_t h_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t h_t(x).$$

Observation 1 states that we may bound the terms $\epsilon^\circ(\{x_t, u_t, \theta_t\}_{t=1}^T)$, $\epsilon^\bullet(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T)$ and $\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T)$ using OCO results from Sect. 3.

We have the following basic setup assumptions.

Assumption 6

- The domain X is convex and admits a proximal setup with norm $\|\cdot\|_X$ and set width Ω_X as in Sect. 3.1.
- For $i \in [m]$, the uncertainty sets U^i are convex and admit proximal setups with norms $\|\cdot\|_{(i)}$ and set widths $\Omega_U < \infty$ as in Sect. 3.1.

Assumption 7 For each $i \in [m]$, the functions $f^i(x, u^i)$ are convex in x , concave in u^i , and are Lipschitz continuous in each variable, i.e., the subgradients are bounded: for all $u^i \in U^i$, $\|\nabla_x f^i(x, u^i)\|_{X,*} \leq G_X < \infty$, and for all $x \in X$, $\|\nabla_u f^i(x, u^i)\|_{(i),*} \leq G_U < \infty$.

Under Assumption 7, the functions $f_t^i(u^i)$ and $h_t(x)$ defined in Observation 1 are convex in u^i and x respectively, and the functions $\phi_t(x, y)$ are convex-concave in x and y . In [19, Sect. 4.1], it is shown that we can bound $\epsilon^\circ(\{x_t, u_t, \theta_t\}_{t=1}^T)$ and $\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T)$ by $O(1/\sqrt{T})$, which then allows us to solve (13) in $T = O(1/\epsilon^2)$

iterations. We will now examine how to improve these bounds under strong convexity and smoothness assumptions on the constraint functions f^i , which will then allow us to improve the rate for solving (13).

We first examine the bounds under strong convexity assumptions.

Assumption 8 For each $i \in [m]$ and any fixed $x \in X$, the functions $f^i(x, u^i)$ are α_U^i -strongly concave in u^i : there exists $\alpha_U^i > 0$ such that $-f^i(x, u^i) - \alpha_U^i \omega(u^i)$ is convex in u^i , where ω is the d.g.f. from the proximal setup for U^i . Furthermore, let $\alpha_U := \min_{i \in [m]} \alpha_U^i$.

Proposition 3 Suppose that Assumptions 6, 7 and 8 hold. Fix any $i \in [m]$ and the set of convex combination weights $\theta_t = \frac{2t}{T(T+1)}$. For any sequence $\{x_t\}_{t=1}^T$, running Algorithm 1 with d.g.f. ω^i , $z_t = u_t^i$, $\xi_t = -\nabla_u f^i(x_t, u_t^i)$ and $\gamma_t = \frac{2}{\alpha_U(t+1)}$ guarantees that

$$\sup_{u^i \in U^i} \sum_{t=1}^T \theta_t f^i(x_t, u^i) - \sum_{t=1}^T \theta_t f^i(x_t, u_t^i) \leq \frac{2G_U^2}{\alpha_U(T+1)}.$$

In particular, for $\theta_t = \frac{2t}{T(T+1)}$, we can choose a sequence $\{u_t\}_{t=1}^T$ such that for any sequence $\{x_t\}_{t=1}^T$, we guarantee $\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T) \leq O(1/T)$.

Proof Assumption 2 holds since Assumptions 6, 7 and 8 hold. Theorem 2 then applies to obtain the upper bounds on the regret terms. \square

Assumption 9 There exists $\alpha_X^i > 0$ such that for each $i \in [m]$ and each fixed $u^i \in U^i$, the function $f^i(x, u^i)$ is α_X^i -strongly convex in x , that is, $f^i(x, u^i) - \alpha_X^i \omega(x)$ is convex, where ω is the d.g.f. from the proximal setup for X . Furthermore, define $\alpha_X := \min_{i \in [m]} \alpha_X^i$.

Proposition 4 Suppose that Assumptions 6, 7 and 9 hold. Fix the set of convex combination weights $\theta_t = \frac{2t}{T(T+1)}$. For any sequence $\{u_t\}_{t=1}^T$, running Algorithm 1 with $z_t = x_t$, $\xi_t = \nabla_x f^{i(t)}(x_t, u_t^{i(t)})$ where $i(t) = \operatorname{argmax}_{i \in [m]} f^i(x_t, u_t^i)$, and $\gamma_t = \frac{2}{\alpha_X(t+1)}$ guarantees that

$$\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T) \leq \frac{2G_X^2}{\alpha_X(T+1)} = O\left(\frac{1}{T}\right).$$

Proof Assumption 2 holds since Assumptions 6, 7 and 9 hold, and for any u , the function $h_u(x) = \max_{i \in m} f^i(x, u^i)$ is strongly convex in x with parameter $\alpha_X = \min_{i \in [m]} \alpha_X^i$. Theorem 2 then applies to obtain the upper bound on the regret term. \square

We now examine the bounds under smoothness assumptions.

Assumption 10 For each $i \in [m]$ and any fixed $x \in X$, the functions $f^i(x, u^i)$ are L_U -smooth in u^i : there exists $L_U < \infty$ such that for any $u^i, (u^i)' \in U^i$,

$$\|\nabla_u f^i(x, u^i) - \nabla_u f^i(x, (u^i)')\|_{i,*} \leq L_U \|u^i - (u^i)'\|_i.$$

Proposition 5 Suppose that Assumptions 6, 7 and 10 hold. Fix any $i \in [m]$. For any sequence $\{x_t\}_{t=1}^T$, running Algorithm 2 with $z_t = u_t^i$, $\eta_t = -\theta_t \nabla_u f^i(x_t, v_t^i)$, $\xi_t = -\theta_t \nabla_u f^i(x_t, u_t^i)$ and $\gamma_t = \frac{1}{L_U \sup_{t \in [T]} \theta_t}$ guarantees that

$$\sup_{u^i \in U^i} \sum_{t=1}^T \theta_t f^i(x_t, u^i) - \sum_{t=1}^T \theta_t f^i(x_t, u_t^i) \leq \Omega_U L_U \sup_{t \in [T]} \theta_t.$$

In particular, for uniform weights $\theta_t = 1/T$ or increasing weights $\theta_t = \frac{2t}{T(T+1)}$, we can choose a sequence $\{u_t\}_{t=1}^T$ such that for any sequence $\{x_t\}_{t=1}^T$, we guarantee $\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T) \leq O(1/T)$.

Proof Assumption 4 holds since Assumptions 6, 7 and 10 hold. Theorem 4 then applies to obtain the upper bounds on the regret terms. \square

Before continuing, we note that the functions $\max_{i \in [m]} f^i(x, u^i)$ are non-smooth in x in general, so we will not examine the term $\epsilon^\bullet(\{x_t, u_t, \theta_t\}_{t=1}^T)$. Instead, we examine the ‘smoothed’ term $\epsilon^\bullet(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T)$, where the functions $\sum_{i=1}^m y^{(i)} f^i(x, u^i)$ are convex-concave and smooth in $[x; y]$. That is, we will bound the online saddle point gap (9) from Observation 1.

Assumption 11 For each $i \in [m]$ and any fixed $u^i \in U^i$, the functions $f^i(x, u^i)$ are L_X -smooth in x^i : there exists $L_X < \infty$ such that for any $x, x' \in X$,

$$\|\nabla_x f^i(x, u^i) - \nabla_x f^i(x', u^i)\|_{X,*} \leq L_X \|x - x'\|_X.$$

Observation 2 Our domain is now $X \times Y$, since we add the variables $y \in Y = \Delta_m$. For the simplex Y , there exists a proximal setup with ℓ_1 -norm and set width $\Omega_Y = \log(m)$. As mentioned in Sect. 3.1, we can construct a norm and proximal setup for $X \times Y$ according to [23, Sect. 5.7.2] and [24, Sect. 6.3.3]. Then the set width of this hybrid setup is $\Omega_{X,Y} = 1$, and under Assumption 11, the smoothness parameter for the function $\phi_u(x, y) = \sum_{i=1}^m y^{(i)} f^i(x, u^i)$ with the constructed norm will be

$$L_{X,Y} := L_X \Omega_X + 2G_X \sqrt{\Omega_X \log(m)}, \quad (14)$$

where G_X is the bound on $\|\nabla_x f^i(x, u^i)\|_{X,*}$. We refer to [29, Sect. 5] and [24, Sect. 6.3.3] for further details.

Proposition 6 Suppose that Assumptions 6, 7 and 11 hold. Fix any $i \in [m]$. For any sequence $\{u_t\}_{t=1}^T$, denote $\phi_t(x, y) = \sum_{i=1}^m y^{(i)} f^i(x, u_t^i)$. Running Algorithm 2 with $z_t = [x_t; y_t]$, $\eta_t = \theta_t [\nabla_x \phi_t(v_t); -\nabla_y \phi_t(v_t)]$, $\xi_t = \theta_t [\nabla_x \phi_t(z_t); -\nabla_y \phi_t(z_t)]$ and $\gamma_t = \frac{1}{L_{X,Y} \sup_{t \in [T]} \theta_t}$ guarantees that

$$\epsilon^\bullet(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T) \leq \Omega_{X,Y} L_{X,Y} \sup_{t \in [T]} \theta_t = \left(L_X \Omega_X + 2G_X \sqrt{\Omega_X \log(m)} \right) \sup_{t \in [T]} \theta_t.$$

In particular, for uniform weights $\theta_t = 1/T$, or for increasing weights $\theta_t = \frac{2t}{T(T+1)}$, we can choose a sequence $\{u_t\}_{t=1}^T$ such that for any sequence $\{x_t\}_{t=1}^T$, we guarantee $\epsilon^*(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T) \leq O\left(\frac{\sqrt{\log(m)}}{T}\right)$.

Proof Assumptions 6, 7 and 11 along with Observation 2 imply that Assumption 5 holds. Then from Theorem 5, we obtain the upper bounds on the regret terms. \square

We now examine how to combine our results to solve the robust feasibility problem (13). To solve (13), we must choose weights $\theta \in \Delta_T$ and generate sequences $\{x_t, u_t, y_t\}_{t=1}^T$ to simultaneously bound $\epsilon^*(\{x_t, u_t, \theta_t\}_{t=1}^T)$ and one of $\epsilon^*(\{x_t, u_t, y_t, \theta_t\}_{t=1}^T)$ or $\epsilon^*(\{x_t, u_t, \theta_t\}_{t=1}^T)$. Depending on the structural assumptions, we would like to combine Propositions 3, 5 and Propositions 4, 6 in a valid fashion to achieve the best possible rate. Every combination is valid, except for Propositions 5 and 6 because of the 1-lookahead (anticipatory) nature of Algorithm 2. We discuss this below.

Remark 13 Note that the sequences $\{u_t\}_{t=1}^T$ and $\{x_t, y_t\}_{t=1}^T$ (or just $\{x_t\}_{t=1}^T$) are generated by two different processes which use inter-related information. Hence, we have to ensure that the information available to each process is sufficient to generate the next step. For example, suppose that we use Proposition 5 to generate the sequence $\{u_t\}_{t=1}^T$. By Remark 10, at iteration t , for each $i \in [m]$ we require the knowledge of the function $f_t^i(u^i) = -f^i(x_t, u^i)$ to compute u_t^i . In other words, we need x_t to compute u_t . As a consequence, we must compute x_t using only knowledge of previous iterations $\{u_s\}_{s=1}^{t-1}$. Therefore, by Remark 2, we cannot use Proposition 6; only Proposition 4 can be utilized. \square

In the light of Remark 13, we can combine these propositions in three different ways under various structural assumptions. We state three results which improve on the $O(1/\epsilon^2)$ convergence from [19]. The proofs of these are straightforward applications of the relevant propositions and hence are omitted.

Theorem 7 Suppose that Assumptions 6, 7, 8 and 9 hold. Then we can solve (13) to within ϵ -approximation in $T = O(1/\epsilon)$ iterations by employing Proposition 3 to generate $\{u_t\}_{t=1}^T$ and Proposition 4 to generate $\{x_t\}_{t=1}^T$ using increasing weights $\theta_t = \frac{2t}{T(T+1)}$. Here, both x_t and u_t are computed with the knowledge of only past iterates x_{t-1}, u_{t-1} .

Theorem 8 Suppose that Assumptions 6, 7, 8 and 11 hold. Then we can solve (13) to within ϵ -approximation in $T = O(\sqrt{\log(m)}/\epsilon)$ iterations by employing Proposition 3 to generate $\{u_t\}_{t=1}^T$ and Proposition 6 to generate $\{x_t, y_t\}_{t=1}^T$ using increasing weights $\theta_t = \frac{2t}{T(T+1)}$. Here, u_t is computed with knowledge of x_{t-1}, u_{t-1} , while $[x_t; y_t]$ is computed with the knowledge of u_{t-1}, u_t and $[x_{t-1}; y_{t-1}]$.

Theorem 9 Suppose that Assumptions 6, 7, 10 and 9 hold. Then we can solve (13) to within ϵ -approximation in $T = O(1/\epsilon)$ iterations by employing Proposition 5 to generate $\{u_t\}_{t=1}^T$ and Proposition 4 to generate $\{x_t\}_{t=1}^T$ using increasing weights $\theta_t = \frac{2t}{T(T+1)}$. Here, x_t is computed with knowledge of x_{t-1}, u_{t-1} , while u_t is computed with the knowledge of u_{t-1}, x_{t-1}, x_t .

Remark 14 As shown in [19, Sect. 4.2, 4.3], OCO algorithms are not the only ways to bound the terms $\epsilon^{\circ}(\{x_t, u_t, \theta_t\}_{t=1}^T)$ and $\epsilon^{\bullet}(\{x_t, u_t, \theta_t\}_{t=1}^T)$. Instead, we can use pessimization oracles from [28] to bound $\epsilon^{\circ}(\{x_t, u_t, \theta_t\}_{t=1}^T)$, or nominal feasibility oracles from [5] to bound $\epsilon^{\bullet}(\{x_t, u_t, \theta_t\}_{t=1}^T)$. A reasonable idea is to combine these oracles with Propositions 3, 4, 5, 6 to obtain improved rates. However, we meet a challenge similar to Remark 13. In iteration t , the pessimization oracles of [28] need knowledge of x_t to compute u_t (see [19, Remark 4.1]), while the nominal feasibility oracle of [5] needs knowledge of u_t to compute x_t (see [19, Remark 4.2]). Therefore, only Propositions 3 and 4 may be used to improve the oracle-based rates. Nevertheless, this still allows us to partially answer the following open question from [5, Sect. 5]: is it possible to improve the $O(1/\epsilon^2)$ oracle calls required to solve (13)? Our results imply the following partial affirmative answer: if every $f^i(x, u^i)$ is *strongly concave* in u^i , then Proposition 3 can be employed to generate $\{u_t\}_{t=1}^T$, which guarantees a solution to (13) in $T = O(1/\epsilon)$ iterations. It remains open whether a provable lower bound on the number of iterations exists with or without additional favorable structure such as strong concavity. \square

5 Application: joint estimation–optimization

In this section, we examine the joint estimation-optimization (JEO) problems **(Opt(u^*))**–**(Est)**. We first establish a relation between iterative methods for JEO problem and regret minimization in OCO. We then show that our results from Sect. 3 can recover most of the results from [2], e.g., when f is smooth or non-smooth and is not strongly convex, and immediately extend these to proximal setups. In addition, we cover the case when f is strongly convex but non-smooth, which as stated in the introduction, is not examined in the prior literature [2, 21, 22]. We first state our basic setup assumptions on the domains and the function f .

Assumption 12

- The domain X is convex and admits a proximal setup as in Sect. 3.1 with set width \mathcal{Q} . Furthermore, it is compact, with $\max_{x, u \in X} \|x - u\| \leq D < \infty$.
- For all $u \in U$, the function $f(\cdot, u)$ is convex in $x \in X$, and is Lipschitz continuous, i.e., the gradients $\nabla_x f(x, u)$ are bounded by a constant $G_{f,X} > 0$ independent of u .

Assumption 13 For any fixed $u \in U$, strong convexity, i.e., Assumption 2 holds for any $f(\cdot, u)$ with uniform strong convexity parameter $\alpha_{f,X} > 0$ independent of u .

Assumption 14 For any fixed $u \in U$, smoothness, i.e., Assumption 4 holds for the function $f(\cdot, u)$ with uniform smoothness parameter $L_{f,X} \geq 0$ independent of u .

As in [2], we also assume access to a sequence of points $\{u_t\}_{t=1}^T$ which approximate the correct data u^* in **(Est)**. Whenever a new approximation u_{t-1} is revealed, we generate a point x_t based on this new data. After T iterations, we build the point $\bar{x}_T = \sum_{t=1}^T \theta_t x_t \in X$ through averaging. Using this scheme, we bound the approximation quality of \bar{x}_T by two terms: an affine regret term based on the sequences $\{x_t, u_t\}_{t=1}^T$

and the function f , and a penalty term for our inability to work with the correct data u^* . We start with a simple lemma, which establishes the link between JEO and OCO.

Lemma 2 Suppose that Assumption 12 holds. Given sequences $\{x_t, u_t\}_{t=1}^T$ and weights $\theta_t \in \Delta_T$, define $q_t(x) := \langle \nabla_x f(x_t, u_t), x \rangle$ and $\bar{x}_T := \sum_{t=1}^T \theta_t x_t \in X$. Then

$$\begin{aligned} f(\bar{x}_T, u^*) - \min_{x \in X} f(x, u^*) &\leq \sum_{t=1}^T \theta_t q_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t q_t(x) \\ &\quad + D \sum_{t=1}^T \theta_t \|\nabla_x f(x_t, u_t) - \nabla_x f(x_t, u^*)\|_*. \end{aligned}$$

If, in addition, Assumption 13 holds, then the same holds with $q_t(x) := \langle \nabla_x f(x_t, u_t), x \rangle + \alpha_{f,X} V_{x_t}(x)$. Furthermore, for either definition of the function q_t ,

$$\begin{aligned} \left| f(\bar{x}_T, u_T) - \min_{x \in X} f(x, u^*) \right| &\leq \sum_{t=1}^T \theta_t q_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t q_t(x) + |f(\bar{x}_T, u_T) \\ &\quad - f(\bar{x}_T, u^*)| + D \sum_{t=1}^T \theta_t \|\nabla_x f(x_t, u_t) - \nabla_x f(x_t, u^*)\|_*. \end{aligned}$$

Proof We will first consider the case when Assumption 13 holds and work with $q_t(x) := \langle \nabla_x f(x_t, u_t), x \rangle + \alpha_{f,X} V_{x_t}(x)$. If Assumption 13 does not hold, the same proof holds by setting $\alpha_{f,X} = 0$. Assumption 13 implies that for any $x \in X$, $f(x_t, u^*) - f(x, u^*) \leq \langle \nabla_x f(x_t, u^*), x_t - x \rangle - \alpha_{f,X} V_{x_t}(x)$. In addition, for any $t \in [T]$,

$$\begin{aligned} \langle \nabla_x f(x_t, u^*), x_t - x \rangle &= \langle \nabla_x f(x_t, u_t), x_t - x \rangle + \langle \nabla_x f(x_t, u^*) - \nabla_x f(x_t, u_t), x_t - x \rangle, \\ &\leq \langle \nabla_x f(x_t, u_t), x_t - x \rangle + D \|\nabla_x f(x_t, u_t) - \nabla_x f(x_t, u^*)\|_*, \end{aligned}$$

where the inequality follows from Cauchy-Schwarz applied to $\langle \nabla_x f(x_t, u^*) - \nabla_x f(x_t, u_t), x_t - x \rangle$ and recognizing $\|x_t - x\| \leq D$ from Assumption 12. After subtracting $\alpha_{f,X} V_{x_t}(x)$ from both sides of this inequality for t , multiplying the resulting inequalities with θ_t , summing them over $t \in [T]$ and using strong convexity of $f(\cdot, u^*)$, we arrive at:

$$\begin{aligned} f(\bar{x}_T, u^*) - f(x, u^*) &\leq \sum_{t=1}^T \theta_t (\langle \nabla_x f(x_t, u_t), x_t - x \rangle - \alpha_{f,X} V_{x_t}(x)) \\ &\quad + D \sum_{t=1}^T \theta_t \|\nabla_x f(x_t, u_t) - \nabla_x f(x_t, u^*)\|_*. \end{aligned}$$

Then the first result follows from $\langle \nabla_x f(x_t, u_t), x_t - x \rangle - \alpha_{f,X} V_{x_t}(x) = q_t(x_t) - q_t(x)$, and taking the maximum of both sides over $x \in X$. The last result follows from the

triangle inequality

$$\begin{aligned} \left| f(\bar{x}_T, u_T) - \min_{x \in X} f(x, u^*) \right| &\leq |f(\bar{x}_T, u_T) - f(\bar{x}_T, u^*)| \\ &\quad + \left| f(\bar{x}_T, u^*) - \min_{x \in X} f(x, u^*) \right|. \end{aligned}$$

□

The last result of Lemma 2 provides a bound on the gap between a computable quantity $f(\bar{x}_T, u_T)$ and the true optimum defined by the correct data u^* . This bound incurs additional penalty terms, $D \sum_{t=1}^T \theta_t \|\nabla_x f(x_t, u_t) - \nabla_x f(x_t, u^*)\|_*$ and $|f(\bar{x}_T, u_T) - f(\bar{x}_T, u^*)|$, which disappear when $u_t = u^*$ for all $t \in [T]$. Hence, these penalty terms can be interpreted as the ‘cost’ of not working with the correct data u^* .

In order to ensure high quality solutions to the JEO problem, we need to bound the gap $|f(\bar{x}_T, u_T) - \min_{x \in X} f(x, u^*)|$, and by Lemma 2 this entails bounding three quantities: the regret term associated with the functions q_t and the two penalty terms. We next demonstrate how the results from [2] on bounding the penalty terms can be recovered from our OCO based analysis. We work under the common assumption of [2] that g is smooth and strongly convex, which assures the existence of algorithms with linear convergence $\|u_t - u^*\| = O(\beta^t)$ for our sequence u_t , and some mild Lipschitz continuity assumptions on $f(x, \cdot)$ and $\nabla_x f(x, \cdot)$. Note that essentially the same results are achievable even if we assume g is non-smooth and strongly convex. In such a case we can quarantine $\|u_t - u^*\| = O(1/t)$, and using this the modification of the other parts of Fact 1 below with the replacement of little- o notation with big- O notation is immediate.

Assumption 15

- The function g in (Est) is strongly convex and smooth in u .
- There exists $G_{f,U} > 0$ such that for all $u, u' \in U$ and $x \in X$, it holds that $|f(x, u) - f(x, u')| \leq G_{f,U} \|u - u'\|$.
- There exists $L_{f,U} > 0$ such that for all $u, u' \in U$ and $x \in X$, we have

$$\|\nabla_x f(x, u) - \nabla_x f(x, u')\|_* \leq L_{f,U} \|u - u'\|.$$

Under Assumption 15, we can bound the two penalty terms in terms of the norms $\|u_t - u^*\|$ as:

$$\begin{aligned} |f(\bar{x}_T, u_T) - f(\bar{x}_T, u^*)| &\leq G_{f,U} \|u_T - u^*\| \\ D \sum_{t=1}^T \theta_t \|\nabla_x f(x_t, u_t) - \nabla_x f(x_t, u^*)\|_* &\leq D L_{f,U} \sum_{t=1}^T \theta_t \|u_t - u^*\|. \end{aligned}$$

Since we assume that $\|u_t - u^*\| = O(\beta^t)$, we can further bound the penalty terms using the following fact.

Fact 1 Consider a sequence $\{u_t\}_{t=1}^T$ such that $\|u_t - u^*\| = O(\beta^t)$ for some $0 < \beta < 1$. Then

- (i) $\|u_T - u^*\| = o(1/T)$.
- (ii) For $\theta_t = 1/T$, we have $\sum_{t=1}^T \theta_t \|u_t - u^*\| = O(1/T) = o(1/\sqrt{T})$.
- (iii) For $\theta_t = \frac{2t}{T(T+1)}$, we have $\sum_{t=1}^T \theta_t \|u_t - u^*\| = O(1/T^2) = o(1/T)$.

Proof Because $\|u_t - u^*\| \leq O(\beta^t) \leq o\left(\frac{1}{T}\right)$, item (i) follows immediately. For item (ii), when $\theta_t = \frac{1}{T}$, we note that

$$\sum_{t=1}^T \theta_t \|u_t - u^*\| \leq \frac{1}{T} O\left(\sum_{t=1}^T \beta^t\right) = O\left(\frac{1}{T}\right).$$

For item (iii), when $\theta_t = \frac{2t}{T(T+1)}$, we observe that

$$\begin{aligned} \sum_{t=1}^T \theta_t \|u_t - u^*\| &\leq \frac{2}{T(T+1)} O\left(\sum_{t=1}^T t\beta^t\right) \\ &\leq \frac{2}{T(T+1)} O\left(\frac{\beta(1-(T+1)\beta^T + T\beta^{T+1})}{(1-\beta)^2}\right) \\ &= O\left(\frac{1}{T^2}\right) = o\left(\frac{1}{T}\right). \end{aligned}$$

□

To complete our bound of the gap $|f(\bar{x}_T, u_T) - \min_{x \in X} f(x, u^*)|$, it remains to bound the weighted regret term associated with the functions q_t in Lemma 2. We can do so by using our results from Sect. 3. We summarize the cases when $f(\cdot, u)$ is not strongly convex in the following remark. Note that these cases are covered by [2, Propositions 4 and 6].

Remark 15 Suppose that Assumption 12 holds, and that we are given a sequence $\{u_t\}_{t=1}^T$ of points from U . Given x_t , define $q_t(x) = \langle \nabla_x f(x_t, u_t), x \rangle$. By applying Theorem 1 appropriately with uniform weights $\theta_t = 1/T$, we obtain the regret bound

$$\sum_{t=1}^T \theta_t q_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t q_t(x) \leq \sqrt{\frac{2\Omega G_{f,X}^2}{T}} = O\left(\frac{1}{\sqrt{T}}\right).$$

By Fact 1, in this case the penalty terms in Lemma 2 are asymptotically negligible $o(1/\sqrt{T})$ compared to the regret bound. This then recovers the overall convergence rate of $O(1/\sqrt{T})$ for solving JEO under the basic Assumption 12, see [2, Proposition 6].

If, in addition, Assumption 14 holds, then by applying Theorem 4 appropriately with uniform weights $\theta_t = 1/T$, the regret associated with q_t is bounded by

$$\sum_{t=1}^T \theta_t q_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t q_t(x) \leq \frac{L_{f,X}\Omega}{T} = O\left(\frac{1}{T}\right).$$

By Fact 1, the penalty terms in this case are asymptotically equivalent $O(1/T)$ to the regret bound. Hence, we recover the overall convergence rate of $O(1/T)$ for solving JEO under Assumptions 12 and 14, see [2, Proposition 4].

Notably, these rates achieved in the JEO framework are the *same rates* for FOMs for solving $(\text{Opt}(u^*))$ for the corresponding classes of functions f when the correct data u^* is available. \square

We now study the case where f is non-smooth and strongly convex; this case was not covered in [2].

Theorem 10 Suppose that Assumptions 12 and 13 hold, and that we are given a sequence $\{u_t\}_{t=1}^T$ of points from U . Given $x_t \in X$, define $q_t(x) := \langle \nabla_x f(x_t, u_t), x \rangle + \alpha_{f,X} V_{x_t}(x)$. Running Algorithm 1 with $z_t = x_t$, $\xi_t = \theta_t \nabla_x f(x_t, u_t)$, weights $\theta_t = \frac{2t}{T(T+1)}$ and step sizes $\gamma_t = \frac{2}{\alpha(t+1)}$ for $t \in [T]$ results in the bound

$$\sum_{t=1}^T \theta_t q_t(x_t) - \inf_{x \in X} \sum_{t=1}^T \theta_t q_t(x) \leq \frac{2G_{f,X}^2}{\alpha_{f,X}(T+1)} = O\left(\frac{1}{T}\right).$$

Furthermore, suppose that Assumption 15 holds, and that $\|u_t - u^*\| = O(\beta^t)$. Define $\bar{x}_T = \sum_{t=1}^T \theta_t x_t$. Then

$$\left| f(\bar{x}_T, u_T) - \min_{x \in X} f(x, u^*) \right| = O\left(\frac{1}{T}\right) + o\left(\frac{1}{T}\right).$$

Proof Assumptions 12 and 13 ensure that the assumptions of Theorem 2 are met, which gives us the regret bound on q_t (note also the equation in (11)). Then we use Lemma 2 to decompose the bound on $|f(\bar{x}_T, u_T) - \min_{x \in X} f(x, u^*)|$ into the regret term and the penalty terms. Also, from Assumption 15 and Fact 1, the penalty terms satisfy

$$\begin{aligned} |f(\bar{x}_T, u_T) - f(\bar{x}_T, u^*)| &\leq G_{f,U} \|u_T - u^*\| = O(\beta^T) = o\left(\frac{1}{T}\right), \\ D \sum_{t=1}^T \theta_t \|\nabla_x f(x_t, u_t) - \nabla_x f(x_t, u^*)\|_* &\leq D L_{f,U} \sum_{t=1}^T \theta_t \|u_t - u^*\| \\ &= O\left(\frac{1}{T^2}\right) = o\left(\frac{1}{T}\right). \end{aligned}$$

The result then follows. \square

Notice that both penalty terms in Theorem 10 are $o(1/T)$, that is, asymptotically negligible compared to the $O(1/T)$ error. Thus, when the data generation process (Est) involves minimizing a smooth and strongly convex function g , the simultaneous JEO approach in Theorem 10 achieves the optimal offline rate of $O(1/T)$ for minimizing non-smooth strongly convex functions [11, Theorem 3.13], plus some asymptotically negligible $o(1/T)$ penalty for not using the correct data. The analysis

presented above depends crucially on the regret bound for the sequence of functions $\{q_t(x) := \langle \nabla_x f(x_t, u_t), x \rangle + \alpha_{f,X} V_{x_t}(x)\}_{t=1}^T$. Therefore, by Remark 6, if we restricted ourselves to standard regret, we would only be able to get a bound of $O(\log(T)/T)$. Thus, our developments and analysis of weighted regret are fundamental in achieving the rate $O(1/T)$.

6 Conclusion

In this paper, we examine iterative solution techniques for RO and JEO through the lens of OCO and study their structure-based acceleration. For this purpose, we advance the line of research in OCO by introducing the concepts of weighted regret, online SP problems, and studying their implications when the decisions are restricted to be made in either non-anticipatory or 1-lookahead fashion. Our analyses demonstrate that when structural information such as smoothness or strong convexity of the loss functions is present, the additional flexibility introduced to the OCO framework by allowing weighted regret and/or 1-lookahead decisions can lead to significant improvements in the convergence rates. These then have immediate consequences on the convergence rates of iterative methods for solving RO problems studied in [5, 19]; in particular Theorem 2 helps in partially resolving an open question from [5] for the lower bound on the number of iterations/calls needed in these iterative frameworks for RO. Moreover, our results also have immediate application in the simultaneous JEO approach studied in [2, 21, 22]. We establish that, in certain cases, our convergence rates for JEO, despite working with only estimates u_t approximating the correct data u^* , match the optimum lower bounds established for offline FOMs solving problems supplied with the correct data u^* .

There are a number of compelling avenues for future research. We believe our results may be further applicable to solve problems with uncertain data in the same spirit of Sects. 4 and 5 and may open up possibilities for more principled solution approaches in other application domains. An important extension of particular interest is the case where the learning problem ([Est](#)) in JEO is no longer static, but it dynamically evolves over time. Lower complexity bounds have been previously established for offline FOMs for problems over simple domains as well as some specific OCO problems. Nevertheless, the flexibilities we have introduced here point out that some of these lower bounds are no longer valid in the new setups (see Remarks 6 and 12). Thus, establishing lower bounds matching our weighted regret (online SP gap) bounds in these setups are of interest. In particular, establishing the tightness of $O(1/T)$ bounds for weighted regret of strongly convex loss functions has a major consequence in determining the worst-case complexity of iterative approaches for solving RO problems. From a practical perspective, in certain applications and/or OCO contexts, it may be reasonable to assume that the players are not presented with exact feedback in the form of gradient/subgradient information but with only their unbiased estimates. Then deriving online stochastic iterative algorithms and studying the impact of several choices such as weighted regret, lookahead decisions, etc., on their behavior is of practical and theoretical interest. In this paper, we have worked under the assumption that our domain is convex; however both RO and JEO have many applications with

nonconvex domains, e.g., involving discrete decision variables. A few online learning algorithms do not rely on such convexity assumption. It is appealing to study the implications of weighted regret and lookahead decisions for such algorithms and their potential use in solving online SP problems as well.

Acknowledgements The authors wish to thank the review team for their constructive feedback that improved the presentation of the material in this paper. This research is supported in part by NSF Grant CMMI 1454548.

References

1. Abernethy, J., Bartlett, P.L., Rakhlin, A., Tewari, A.: Optimal strategies and minimax lower bounds for online convex games. In: Proceedings of the 19th Annual Conference on Computational Learning Theory (2008)
2. Ahmadi, H., Shanbhag, U.V.: Data-driven first-order methods for misspecified convex optimization problems: global convergence and rate estimates. In: 53rd IEEE Conference on Decision and Control, pp. 4228–4233 (2014)
3. Andrew, L.L., Barman, S., Ligett, K., Lin, M., Meyerson, A., Roytman, A., Wierman, A.: A tale of two metrics: simultaneous bounds on competitiveness and regret. *J. Mach. Learn. Res.: Workshop Conf. Proc.* **30**, 741–763 (2013)
4. Ben-Tal, A., Ghaoui, L., Nemirovski, A.: Robust Optimization. Princeton Series in Applied Mathematics. Princeton University Press, Princeton (2009)
5. Ben-Tal, A., Hazan, E., Koren, T., Mannor, S.: Oracle-based robust optimization via online learning. *Oper. Res.* **63**(3), 628–638 (2015)
6. Ben-Tal, A., Nemirovski, A.: Robust convex optimization. *Math. Oper. Res.* **23**(4), 769–805 (1998)
7. Ben-Tal, A., Nemirovski, A.: Robust optimization—methodology and applications. *Math. Program.* **92**(3), 453–480 (2002)
8. Ben-Tal, A., Nemirovski, A.: Selected topics in robust convex optimization. *Math. Program.* **112**(1), 125–158 (2008)
9. Bertsimas, D., Brown, D.B., Caramanis, C.: Theory and applications of robust optimization. *SIAM Rev.* **53**(3), 464–501 (2011)
10. Borodin, A., Linial, N., Saks, M.E.: An optimal on-line algorithm for metrical task system. *J. ACM* **39**(4), 745–763 (1992)
11. Bubeck, S.: Convex optimization: algorithms and complexity. *Found. Trends Mach. Learn.* **8**(3–4), 231–357 (2015)
12. Buchbinder, N., Chen, S., Naor, J., Shamir, O.: Unified algorithms for online learning and competitive analysis. *J. Mach. Learn. Res.: Workshop Conf. Proc.* **23**, 5.1–5.18 (2012)
13. Caramanis, C., Mannor, S., Xu, H.: Robust optimization in machine learning. In: Sra, S., Nowozin, S., Wright, S. (eds.) Optimization for Machine Learning. MIT Press, Cambridge (2012)
14. Cesa-Bianchi, N., Lugosi, G.: Prediction, Learning, and Games. Cambridge University Press, Cambridge (2006)
15. Chiang, C.-K., Yang, T., Lee, C.-J., Mahdavi, M., Lu, C.-J., Jin, R., Zhu, S.: Online optimization with gradual variations. In: Conference on Learning Theory, pp. 6–1 (2012)
16. Goldfarb, D., Iyengar, G.: Robust portfolio selection problems. *Math. Oper. Res.* **28**(1), 1–38 (2003)
17. Hazan, E.: Introduction to online convex optimization. *Found. Trends Optim.* **2**(3–4), 157–325 (2016)
18. Hazan, E., Kale, S.: Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *J. Mach. Learn. Res.* **15**, 2489–2512 (2014)
19. Ho-Nguyen, N., Kılınç-Karzan, F.: Online First-order Framework for Robust Convex Optimization. *Oper. Res. (to appear)*
20. Jenatton, R., Huang, J., Csiba, D., Archambeau, C.: Online Optimization and Regret Guarantees for Non-additive Long-Term Constraints. Technical report (2016). [arXiv:1602.05394](https://arxiv.org/abs/1602.05394)
21. Jiang, H., Shanbhag, U.V.: On the solution of stochastic optimization problems in imperfect information regimes. In: 2013 Winter Simulations Conference, pp. 821–832 (2013)
22. Jiang, H., Shanbhag, U.V.: On the solution of stochastic optimization and variational problems in imperfect information regimes. *SIAM J. Optim.* **26**(4), 2394–2429 (2016)

23. Juditsky, A., Nemirovski, A.: First-order methods for nonsmooth convex large-scale optimization, I: general purpose methods. In: Sra, S., Nowozin, S., Wright, S. (eds.) Optimization for Machine Learning. MIT Press, Cambridge (2012)
24. Juditsky, A., Nemirovski, A.: First-order methods for nonsmooth convex large-scale optimization, II: utilizing problem's structure. In: Sra, S., Nowozin, S., Wright, S. (eds.) Optimization for Machine Learning. MIT Press, Cambridge (2012)
25. Koppel, A., Jakubiec, F.Y., Ribeiro, A.: A saddle point algorithm for networked online convex optimization. *IEEE Trans. Signal Process.* **63**(19), 5149–5164 (2015)
26. Lacoste-Julien, S., Schmidt, M.W., Bach, F.R.: A Simpler Approach to Obtaining an $O(1/t)$ Convergence Rate for the Projected Stochastic Subgradient Method. Technical report (2012). [arXiv:1212.2002](https://arxiv.org/abs/1212.2002)
27. Mahdavi, M., Jin, R., Yang, T.: Trading regret for efficiency: online convex optimization with long term constraints. *J. Mach. Learn. Res.* **13**, 2503–2528 (2012)
28. Mutapcic, A., Boyd, S.: Cutting-set methods for robust convex optimization with pessimizing oracles. *Optim. Methods Softw.* **24**(3), 381–406 (2009)
29. Nemirovski, A.: 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 J. Optim.* **15**(1), 229–251 (2004)
30. Nesterov, Y.: Smooth minimization of non-smooth functions. *Math. Program.* **103**(1), 127–152 (2005)
31. Nedić, A., Lee, S.: On stochastic subgradient mirror-descent algorithm with weighted averaging. *SIAM J. Optim.* **24**(1), 84–107 (2014)
32. Rakhlin, A., Shamir, O., Sridharan, K.: Making gradient descent optimal for strongly convex stochastic optimization. In: Langford, J., Pineau, J. (eds.) Proceedings of the 29th International Conference on Machine Learning (ICML-12) (ICML'12), pp. 449–456. Omnipress, New York, NY, USA (2012)
33. Rakhlin, A., Sridharan, K.: Online learning with predictable sequences. In: Conference on Learning Theory, pp. 993–1019 (2013)
34. Rakhlin, A., Sridharan, K.: Optimization, learning, and games with predictable sequences. In: Advances in Neural Information Processing Systems, pp. 3066–3074 (2013)
35. Robbins, H.: Asymptotically subminimax solutions of compound statistical decision problems. In: In Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, pp. 131–149 (1950)
36. Shalev-Shwartz, S.: Online learning and online convex optimization. *Found. Trends Mach. Learn.* **4**(2), 107–194 (2012)
37. Sion, M.: On general minimax theorems. *Pac. J. Math.* **8**(1), 171–176 (1958)
38. Yang, T., Mahdavi, M., Jin, R., Zhu, S.: Regret bounded by gradual variation for online convex optimization. *Mach. Learn.* **95**(2), 183–223 (2014)
39. Zinkevich, M.: Online convex programming and generalized infinitesimal gradient ascent. In: Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21–24, 2003, Washington, DC, USA, pp. 928–936 (2003)