Risk-Averse Stochastic Convex Bandit

Adrian Rivera Cardoso **Huan Xu**
Georgia Institute of Technology

Abstract

Motivated by applications in clinical trials and finance, we study the problem of online convex optimization (with bandit feedback) where the decision maker is risk-averse. We provide two algorithms to solve this problem. The first one is a descent-type algorithm which is easy to implement. The second algorithm, which combines the ellipsoid method and a center point device, achieves (almost) optimal regret bounds with respect to the number of rounds. To the best of our knowledge this is the first attempt to address risk-aversion in the online convex bandit problem.

1 Introduction

In this paper we study the problem of Online Risk-Averse Stochastic Optimization which generalizes Online Convex Optimization (OCO) when the loss functions are sampled i.i.d from an unknown distribution. During the last decade OCO has received a lot of attention due to its many applications and tight relations with problems such as Universal Portfolios [7, 17, 18], Online Shortest Path [29], Online Submodular Minimization [14], Convex Optimization [4, 13], Game Theory [6] and many others. Along with OCO came Online Bandit Optimization (OBO) a similar but more challenging line of research, perhaps more realistic in some applications, where the feedback is limited to observing only the function *values* of the chosen actions (bandit feedback) instead of the whole functions [9]. The standard goal of OCO and OBO is to develop algorithms such that the standard average regret

$$\frac{1}{T}\sum_{t=1}^{T} f_t(x_t) - \frac{1}{T}\min_{x\in X}\sum_{t=1}^{T} f_t(x)$$

vanishes as quickly as possible. In other words, we want our average loss to be as close as possible to the best loss if we had known all the functions in advance and committed to one action. Here the sequence of convex functions $\{f_t\}_{t=1}^T$ may be chosen by an adversary and the regret minimizing algorithm chooses action x_{t+1}, in some bounded convex set X by using only the information available at time t. This means that in the OCO setting the algorithm may use $\{x_1, ..., x_t\}$ and $\{f_1(\cdot), ...f_t(\cdot)\}$, and in the OBO setting it may only use $\{x_1, ..., x_t\}$ and $\{f_1(x_1), ...f_t(x_t)\}$. Due to recent breakthroughs [5, 15] we now have efficient algorithms (that meet lower bounds in terms of the number of rounds $\Omega(\frac{1}{\sqrt{T}})$ up to logarithmic factors) for both problems, OBO and OCO. While the set up of OCO and OBO is very powerful because it allows for the loss functions to be chosen adversarially, in some applications such as medicine and finance this may not be enough.

Let us consider an example in clinical trials. Suppose there are T patients with some rare disease and we have at our disposal a new drug that has the potential to cure the disease if we prescribe the right dose. Since we do not know what the right dose is, we must learn it as we treat each patient. In other words, we will choose a dose, observe the reaction of a patient and chose a new dose for the next patient. The previous problem can of course be be abstracted as an OBO problem, where each function $f_t(\cdot)$ encodes how patient t will react to the dose we prescribe x_t. Here, the assumption that f_t is chosen adversarially may not be very realistic and perhaps it makes more sense to assume that f_t is drawn randomly from some family of functions. An algorithm that guarantees that the standard average regret vanishes can be seen as an algorithm that is choosing the optimal dose for the average patient, something that is non-trivial to do. Unfortunately, such guarantee completely ignores what may happen to patients that do not look like the average patient. It could be that the optimal dose for the average patient has really negative effects on 5% of the patients. In this case, a dose that is slightly less effective on the average patient but does not harm the unlucky 5% may be more desirable. Thus, the goal of

Proceedings of the 22[nd] International Conference on Artificial Intelligence and Statistics (AISTATS) 2019, Naha, Okinawa, Japan. PMLR: Volume 89.

this paper is to provide algorithms for OCO and OBO that *explicitly incorporate risk*. By "risk" we mean the possibility of really negative outcomes, as it is used in the Economics and Operations Research communities.

Another area where an explicit consideration of risk must be taken into account is finance. For example, in [8] the authors show that in the online portfolio problem, risk neutral guarantees such as performing as well as the best constant rebalanced portfolio (i.e. minimizing standard average regret) may not perform well in practice. They show through experiments on the S&P500 that the simple strategy that maintains uniform weights on all the stocks outperforms that which seeks to perform as well as the best stock (regardless of its theoretical guarantees). To explicitly incorporate risk into the setting of OCO and OBO we will use a coherent risk measure called Conditional Value at Risk ($CVaR$) [24], sometimes also called Expected Shortfall, which is widely used in the financial industry. After the financial crisis of 2008, the Basel Committee on Banking Supervision created the Third Basel Accord (Basel III), a set of regulatory measures to strengthen the regulation, supervision and risk management of the banking sector [10]. In this accord one of the main points was to migrate from quantitative risk measures such as Value at Risk to Conditional Value at Risk since it better captures tail risk.

It should be clear from the previous examples that generally speaking, human decision makers are risk-averse. They prefer consistent sequences of rewards instead of highly variable sequences with slightly better rewards. Because of the previous, we want to develop algorithms that explicitly incorporate risk which have strong theoretical guarantees.

Our main contributions are the following. First, we develop and analyze two algorithms for the online stochastic convex bandit problem that explicitly incorporate the risk aversion of the decision maker (as measured by the $CVaR$). On our way we develop a finite-time concentration result for the $CVaR$. Second, we extend our results to the case where the decision maker uses more general risk measures to measure risk by using the Kusuoka representation theorem.

2 Related Work

Risk aversion has received very little attention in the online learning setting. The few existing work all focuses on the case where *the number of actions is finite*. For the stochastic multi-armed bandit problem, [25] provide algorithms that ensure the mean-variance of the sequence of rewards generated by the algorithm is not too far from the mean-variance of the rewards generated by the best arm. In [30] the same problem is studied and the authors provide tighter upper and lower bounds. In [19] the author considers a different risk measure, the cumulant generative function, and provide similar guarantees for a slightly modified definition of regret. In [11] the authors consider the $CVaR$ as measure of risk aversion and provide algorithms that achieve sublinear regret. The notion of regret they use is different from the one we will use as they do not look at the risk of the sequence of rewards obtained by the algorithms, but instead they seek to perform as well as the arm that minimizes $CVaR$ (i.e., "pseudo regret" as we called). The pseudo regret bound they prove, although optimal with respect to T scales linearly in the number of arms. By using a discretization approach in our setting together with their algorithm would yield an algorithm with pseudo regret that depends exponentially in the dimension of the problem with exponential running time. The previous is of course undesirable, therefore different tools must be used. In [31] the authors study the related problem of best arm identification where the goal is to identify the arm with the best risk measure. They consider Value at Risk, $CVaR$, and Mean-Variance as risk measures. In [8] the authors consider risk aversion in the experts problem. This setting is similar to the multi-armed bandit problem with the difference that the rewards are assigned adversarially, and at each time step all the rewards are visible to the player. In particular they seek to build algorithms such that the mean variance (or Sharpe ratio) of the sequence of rewards generated by the algorithm are as close as possible to that of the best expert. They show negative results for this problem however they provide algorithms that perform well for "localized" versions of the risk measures they consider.

To the best of our knowledge, all existing work that explicitly incorporates risk aversion under the assumptions of stochastic rewards and bandit feedback is restricted to the multi-armed bandit model. This paper is the first to consider an infinite number of arms and incorporate risk aversion under bandit feedback. In [8], where risk aversion in the experts problem is studied, one can think of instead of choosing an expert at every round one chooses a probability distribution over the experts. While the set of probability distributions over the experts is a convex set, this is a very specialized case (linear functional and simplex feasible set). Moreover, the authors assume full information feedback and adversarial rewards, which are very different from our setup.

3 Preliminaries

This section is devoted to preliminaries. In particular we review relevant concepts and technical results

essential to develop the proposed algorithms.

3.1 Notation

Let $||\cdot||$ be the l_2 norm unless otherwise stated. By default all vectors are column vectors, a vector with entries $x_1, ..., x_n$ is written as $x = [x_1; ...; x_n] = [x_1, ..., x_n]^\top$ where $\top$ denotes the transpose. For a random variable X, $X \sim P$ means that X is distributed according to distribution P. We let $\nabla g(x)$ be any element in the subdifferential of g at x. Whenever we write $\nabla f(x, \xi)$ we mean $\nabla_x f(x, \xi)$. Throughout the paper we will use O notation to hide constant factors. We use $\tilde{O}$ notation to hide constant factors and polylogarithmic factors of the number of rounds T, the inverse risk level $\frac{1}{\alpha}$ and the dimension of the problem d.

3.2 One-Point Gradient Estimation

Consider function $f : \mathbb{R}^d \to \mathbb{R}$ which is G-Lipschitz continuous. Define its smoothened version

$$\hat{f}^\delta(x) := \mathbb{E}_{v\sim\mathbb{B}}[f(x + \delta v)]$$

where $\mathbb{B}$ is the uniform distribution over the unit ball of appropriate dimension. From now on we omit superscript δ and write $\hat{f}(x)$. Define random quantity

$$g = \frac{d}{\delta} f(x + \delta u)u \tag{1}$$

with $u \sim \mathbb{S}$ where $\mathbb{S}$ is the uniform distribution over the unit sphere. We have the following

Lemma 1. *[16][Ch.2] $\hat{f}$ satisfies the following:*

1. *If f is α-strongly convex then so is $\hat{f}$*
2. $|f(x) - \hat{f}(x)| \leq \delta G$
3. $\mathbb{E}[g] = \nabla \hat{f}(x)$

That is, the smoothened version of f is convex as well, it is not too far from f, and by sampling from the unit sphere we can obtain an unbiased estimate of its gradient.

3.3 Conditional Value at Risk

In [24] the authors define the α-Value at Risk of random variable X as

$$VaR_\alpha[X] := \inf\{t : P(X \leq t) \geq 1 - \alpha\}.$$

Using the above definition they define Conditional Value at Risk ($CVaR$, sometimes also called Expected Shortfall) as

$$C_\alpha[X] := CVaR_\alpha[X] := \frac{1}{\alpha}\int_{1-\alpha}^{\alpha} VaR_{1-\tau}[X]d\tau. \tag{2}$$

Moreover, when the random variable has c.d.f. $H(x)$ continuous at $x = VaR_\alpha[X]$ it holds that

$$C_\alpha[X] = \mathbb{E}[X | X \geq VaR_\alpha[X]]. \tag{3}$$

We make use of the following notation. Let $\{a_t\}_{t=1}^T$ be an arbitrary sequence of real numbers, we let $C_\alpha[\{a_t\}_{t=1}^T]$ be the Conditional Value at Risk of the discrete random variable that takes each value a_t with probability $1/T$ for all $t = 1, ..., T$.

Below we state some well known results that will be used later. The proofs for the next two lemmas can be found in [28].

Lemma 2.

$$C_\alpha[X] = \min_{z\in\mathbb{R}} z + \frac{1}{\alpha}\mathbb{E}[X - z]_+, \tag{4}$$

where $[a]_+ := \max\{a, 0\}$. In fact, if $0 \leq X \leq 1$ with probability 1, the condition $z \in \mathbb{R}$ can be replaced with $z \in [0, 1]$.

Lemma 3. *Let ξ be a random variable supported in Ξ with distribution P, let $X \subset \mathbb{R}$ be a convex and compact and let $f : X \times \Xi \to \mathbb{R}$ be convex in x for every ξ. Define $F = f(x, \xi)$. Then*

$$C_\alpha[F](x) := CVaR_\alpha[F](x) = \min_z z + \frac{1}{\alpha}\mathbb{E}_\xi[f(x, \xi) - z]_+$$

and $C_\alpha[F](x)$ is a convex function of x. In fact, if $f(\cdot, \xi)$ is β-strongly convex for every $\xi \in \Xi$, then so is $C_\alpha[F](x)$.

4 Problem Setup

In this section we formally define the setup of our problem. Let ξ be a random variable supported in Ξ with unknown distribution P. Let $X \subset \mathbb{R}^d$ be a convex and compact set with diameter D_X that contains the origin. Let $f : X \times \Xi \to \mathbb{R}$ be a convex function in the first argument for every $\xi \in \Xi$. Let f satisfy $||\nabla f(x, \xi)|| \leq G$ for every $x \in X$ and every $\xi \in \Xi$. We define random function $F(x) = f(x, \xi)$ in the sense that for every $x \in X$, $F(x)$ is a random variable. We also assume that for every $x \in X$, $0 \leq F(x) \leq 1$ with probability 1.

A risk-averse player will make decisions in a *stochastic environment* for T time steps. In every time step $t = 1, ..., T$ the player chooses action $\tilde{x}_t \in X$, and nature obtains sample ξ_t from P. Then, the player incurs and observes only the loss incurred by its action $f(\tilde{x}_t, \xi_t)$ (when convenient we also make use of the following notation $f_t(x) \triangleq f(x, \xi_t)$). If the player were risk neutral then a reasonable goal would be to design an algorithm that obtains (in expectation) vanishing

standard average regret, that is

$$\mathbb{E}[\frac{1}{T}\sum_{t=1}^{T} f(\tilde{x}_t,\xi_t) - \frac{1}{T}\min_{x\in X}\sum_{t=1}^{T} f(x,\xi_t)] = o(1).$$

Where the expectation is taken with respect to the random draw of functions and the internal randomization of the algorithm. Such is the standard goal of OCO and OBO, and as mentioned in the introduction, there already exist polynomial time algorithms that achieve the optimal lower bound of $\Omega(1/\sqrt{T})$ (up to logarithmic factors) even when the functions f are chosen by an adversary instead of from some distribution.

In our setting, since the player is risk averse, the notion of average regret is not appropriate. In this section we assume that the player uses the Conditional Value at Risk $C_\alpha[\cdot] = CVaR_\alpha[\cdot]$ for some $\alpha \in (0,1]$ to measure risk (when $\alpha = 1$, $C_\alpha[\cdot] = \mathbb{E}[\cdot]$ i.e. the player becomes risk neutral). With this in mind, the following two quantities become interesting, namely pseudo-$CVaR$-regret defined as

$$\bar{\mathcal{R}}_T := \frac{1}{T}\sum_{t=1}^{T} C_\alpha[F](\tilde{x}_t) - \frac{1}{T}\min_{x\in X}\sum_{t=1}^{T} C_\alpha[F](x) \quad (5)$$

and $CVaR$-regret defined as

$$\mathcal{R}_T := C_\alpha[\{f_t(\tilde{x}_t)\}_{t=1}^T] - \min_{x\in X} C_\alpha[\{f_t(x)\}_{t=1}^T],$$

where we make more explicit what we mean by $C_\alpha[\{f_t(x_t)\}_{t=1}^T]$ in the next paragraph. In this setup, a risk averse player may be concerned with two types of risk, the risk of the individual losses it incurs and the overall risk of playing the game. The player that is concerned about the risk of the individual losses, should be pleased with an algorithm that obtains vanishing $\bar{\mathcal{R}}_T$, this would ensure that the average risk of the losses it incurs is not too far from that of the best point in the set.

On the other hand, the player that is concerned about the overall risk of playing the game may desire a different guarantee. Notice that the sequence of losses that the player incurs $\{f_t(\tilde{x}_t)\}_{t=1}^T$ defines an empirical distribution where every realization $f_t(\tilde{x}_t)$ occurs with probability $\frac{1}{T}$ and as such we can compute its risk $C_\alpha[\{f_t(\tilde{x}_t)\}_{t=1}^T]$. It is then natural for the player to desire a sequence of losses that has risk as close as possible to the minimum risk sequence of losses (where the sequence is generated by playing only one action). The quantity $\mathcal{R}_T$ makes the previous statement precise.

A reader familiar with the OBO literature may notice that (5) already looks like a quantity for which running Online Gradient Descent without a Gradient may yield vanishing regret. Unfortunately, at every step all we observe is $f_t(\tilde{x}_t)$ and not $C_\alpha[F](\tilde{x}_t)$. To obtain a reasonable (not too noisy) evaluation of $C_\alpha[F](\cdot)$ the same x must be played for several rounds. It is possible to design algorithms that follow this idea, however, since we were able to develop better algorithms for the same problem we do not further discuss the details of this somewhat naive approach.

5 A Finite-Time Concentration Result for the $CVaR$

Before we present the algorithms we must derive a finite-time concentration result for the $CVaR$. This result will be heavily used to prove sublinear regret bounds for both algorithms. In [28] the authors present an asymptotic result. Unfortunately, since our goal is to achieve finite-time bounds we could not use it and had to prove our own result. To the best of our knowledge this is the first finite time concentration result for the $CVaR$.

Theorem 1. *Suppose $0 \le f(x,\xi) \le 1$ for every $x \in X$ and every $\xi \in \Xi$. For any $x \in X$, let the N-sample estimate of $CVaR_\alpha[F](x)$ be $\widehat{CVaR_\alpha[F]}(x) := \min_{z\in Z} z + \frac{1}{\alpha N}\sum_{n=1}^{N}[f(x,\xi_n)-z]_+$. Where $Z := [0,1]$. It holds that with probability at least $1-\delta$,*

$$|CVaR_\alpha[F](x) - \widehat{CVaR_\alpha[F]}(x)| \le O(\sqrt{\frac{\ln(N/\delta)}{\alpha^2 N}}).$$

While the previous result holds with high probability it is also possible to derive from it a result that holds in expectation.

To prove such a result we had to use a finite time concentration result for Lipschitz functions from [26] applied to the sequence of functions $\{z + \frac{1}{\alpha}[f(x,\xi_t) - z]_+\}_{t=1}^T$. After this, some extra work had to be done transform this guarantee into one that holds for the $CVAR$. A formal proof of the theorem can be found in the appendix.

6 Algorithm 1

In this section we provide an algorithm that obtains vanishing regret while playing an action only once. The key to the algorithm is to look at functions $\mathcal{L}_t(x,z) := z + \frac{1}{\alpha}[f(x,\xi_t) - z]_+$ which by Lemma 3 are closely related to $C_\alpha[F](x)$. Although with one sample we can not evaluate (accurately enough) $C_\alpha[F](\cdot)$, we can evaluate $\mathcal{L}_t$. This observation is important because it will allow us to build one-point gradient estimators of the smoothened function $\hat{\mathcal{L}}_t$ as it is done in [9]. These one-point gradient estimators will allow us to perform a descent step. This idea allows us to

obtain sublinear pseudo-regret. The rest of the analysis consists of using the bound on the pseudo-regret to bound the regret.

Algorithm 1

Input: $X \subset \mathbb{R}^d$, $x_1 \in X$, $z_1 \in Z := [0,1]$ step size η, δ
for $t = 1, ..., T$ **do**
Sample $u \sim \mathbb{S}^{d+1}$
Let $u^1 = [u_1; ...; u_d]$ and $u^2 = u_{d+1}$
Play $\tilde{x}_t := x_t + \delta u^1$, incur and observe loss $f_t(\tilde{x}_t)$

Let $\tilde{z}_t = z_t + \delta u^2$
Let $g_t^1 := \frac{(d+1)}{\delta}(\tilde{z}_t + \alpha^{-1}[f_t(\tilde{x}_t) - \tilde{z}_t]_+)u^1$
Let $g_t^2 := \frac{(d+1)}{\delta}(\tilde{z}_t + \alpha^{-1}[f_t(\tilde{x}_t) - \tilde{z}_t]_+)u^2$
Update $x_{t+1} \leftarrow \Pi_{X_\delta}(x_t - \eta g_t^1)$
Update $z_{t+1} \leftarrow \Pi_{Z_\delta}(z_t - \eta g_t^2)$
end for

Here $\mathbb{S}^d$ denotes the uniform distribution over the d-dimensional unit sphere, $X_\delta := \{x : \frac{1}{1-\delta}x \in X\}$ and $\Pi_X[\cdot]$ denotes the $||\cdot||_2$ projection onto convex set X.

We have the following two main results.

Theorem 2. *Using $\eta = \frac{\alpha D_{\mathcal{L}}}{(d+1)T^{3/4}}$ and $\delta = \frac{1}{T^{1/4}}$ Algorithm 1 guarantees:*

$$\mathbb{E}[\bar{\mathcal{R}}_T] \leq O(\frac{d}{\alpha T^{1/4}}).$$

Where the expectation is taken over the random draw of functions and the internal randomization of the algorithm. $D_{\mathcal{L}}$ is specified in the appendix.

Theorem 3. *Let $f(x, \xi)$ be strongly convex with parameter $\beta > 0$. Algorithm 1 guarantees*

$$\mathbb{E}[\mathcal{R}_T] \leq \tilde{O}(\frac{d^{1/2}}{\alpha^{3/2}\beta^{1/2}T^{1/8}}).$$

Where the expectation is taken over the random draw of functions and the internal randomization of the algorithm.

The proofs of these theorems can be found in the appendix.

7 Algorithm 2

Algorithm 1, while it is intuitive and easy to implement, does not achieve the optimal pseudo-regret bound of $\frac{1}{\sqrt{T}}$. In this section, we adapt an algorithm from [2] that achieves the optimal regret bound (up to logarithmic factors), unfortunately its dependency on d is less than ideal. We consider the cases $d = 1$ and $d > 1$ separately.

7.1 The 1-Dimensional Case

For simplicity, in this section we assume that $X = [0,1]$ and that $f(\cdot, \xi)$ is 1-Lipschitz continuous for every $\xi \in \Xi$. This implies that $C_\alpha[F](\cdot)$ is also 1-Lipschitz continuous (see Lemma 10 in the appendix). We let $LB_{\gamma_i}(x)$ and $UB_{\gamma_i}(x)$ denote the $C_\alpha[F](\cdot)$ lower and upper bounds of the confidence intervals (CI's) of width γ_i at point x. That is, sample point x $\frac{\ln(T/(\alpha\gamma))}{\gamma_i^2\alpha^2}$ times, compute the empirical $CVaR_\alpha$, $\hat{C}_\alpha[F](x)$ and let $UB_{\gamma_i}(x) := \hat{C}_\alpha[F](x) + \gamma_i$ and $LB_{\gamma_i}(x) := \hat{C}_\alpha[F](x) - \gamma_i$.

The algorithm proceeds in epochs and rounds until we have played a total of T times. In epoch τ the algorithm works with region $[l_\tau, r_\tau]$. In this region we will be playing three points x_l, x_c, x_r (x_c is the center point) for several rounds $i = 1, 2, ...$. In each round i the algorithm will play $\frac{\ln(T/(\alpha\gamma))}{\alpha^2\gamma_i^2}$ times the aforementioned points and build CI's for $C_\alpha[F]$. Roughly speaking, the reason why the algorithm works is because in every round we are 1) either playing points such that we are not suffering too much pseudo-regret or 2) we are quickly identifying a subregion of the working region which only contains "bad points" and discarding it. Every time 2) occurs we are shrinking the working region by a constant factor, this will guarantee that after not too many rounds we are only working with a small feasible region.

For convenience we denote $h(x) := C_\alpha[F](x)$ and $x^* := \text{argmin}_{x \in X} h(x)$. Notice that the minimizer need not be unique in which case we choose one arbitrarily. At the end of a round one of the following occurs:

Case 1. The CI's around $h(x_l)$ and $h(x_r)$ are sufficiently separated. If this is the case, then by convexity we can discard one fourth of the working feasible region: either the one to the left of x_l or the one to the right of x_r .

Case 2. If Case 1 does not occur, the algorithm checks if the CI around $h(x_c)$ is sufficiently below at least one of the CI's around $h(x_l)$ or $h(x_r)$. If this is the case then we can discard one fourth of the working feasible region.

Case 3. If neither Case 1 or Case 2 occurs then we can be sure that the function is flat in the working feasible region (as measured by γ) and thus we are not incurring a very high pseudo-regret.

The main results of this section are the following.

Theorem 4. *With probability at least $1 - \frac{1}{T}$, Algorithm 2 (1-D) guarantees*

$$\bar{\mathcal{R}}_T \leq O(\frac{\ln(T)}{\sqrt{T}\alpha} \ln(\frac{\alpha T}{\ln(T)})).$$

Algorithm 2 ($d = 1$)

Input: Input: $X \in [0,1]$, total number of time-steps T
Let $l_1 := 0, r_1 := 1$
for epoch $\tau = 1, 2, ...$ **do**
  Let $w_\tau := r_\tau - l_\tau$
  Let $x_l := l_\tau + w_\tau/4, x_c = l_\tau + w_\tau/2, x_r := l_\tau + 3w_\tau/4$
  for round $i = 1, 2, ...$: **do**
    Let $\gamma_i = 2^{-i}$
    For each $x \in \{x_l, x_c, x_r\}$ play x $\frac{\ln(T/(\alpha\gamma))}{\gamma_i^2\alpha^2}$ times and build CI's: $[\hat{C}_\alpha[F](x_k)] - \gamma_i, \hat{C}_\alpha[F](x_k) + \gamma_i]$ for $k \in \{l, c, r\}$
    if $\max\{LB_{\gamma_i}(x_l), LB_{\gamma_i}(x_r)\} \geq \min\{UB_{\gamma_i}(x_l), UB_{\gamma_i}(x_r)\} + \gamma_i$ (Case 1) **then**
      if $LB_{\gamma_i}(x_l) \geq LB_{\gamma_i}(x_r)$ **then**
        set $l_{\tau+1} := x_l$ and $r_{\tau+1} := r_\tau$
      else
        set $l_{\tau+1} := l_\tau$ and $r_{\tau+1} := x_r$
      end if
      Continue to epoch $\tau + 1$
    else if $\max\{LB_{\gamma_i}(x_l), LB_{\gamma_i}(x_r)\} \geq UB_{\gamma_i}(x_c) + \gamma_i$ (Case 2) **then**
      if $LB_{\gamma_i}(x_l) \geq LB_{\gamma_i}(x_r)$ **then**
        set $l_{\tau+1} := x_l$ and $r_{\tau+1} := r_\tau$
      else
        set $l_{\tau+1} := l_\tau$ and $r_{\tau+1} := x_r$
      end if
      Continue to epoch $\tau + 1$
    end if (Case 3)
  end for
end for

Theorem 5. *Let $f(\cdot, \xi)$ be strongly convex with parameter $\beta > 0$ for all $\xi \in \Xi$. With probability at least $1 - \frac{3}{T}$, Algorithm 2 (1-D) guarantees*

$$\mathcal{R}_T \leq \tilde{O}(\frac{1}{\alpha^{3/2}\beta^{1/2}T^{1/4}}).$$

We follow [2] for the analysis of the algorithm. The main difference in the analysis is that we must build estimates of the $CVaR$ of the random loss at every point instead of building them for the expected loss. Because of this, we have to use our concentration result from Section 5. This directly affects how many times we must choose an action.The detailed analysis of the algorithm and the proofs of the theorems in this section can be found in the appendix.

7.2 The d-Dimensional Case

Let us first consider the problem of minimizing a convex function over a bounded set with a first-order oracle (i.e. a gradient and function value oracle). For simplicity let us assume that the convex set is a ball. An ellipsoid-type method would work really well in this setup because of the following. By querying the first order oracle at any point (due to convexity) we could identify a subregion of the current feasible region where the function value is worse than the function value at the point we made the query. If we could somehow discard that bad portion of the feasible set, and the size of this bad region is big enough, by iterating the procedure (assuming this can be done) we should end up with a set that only has points close to optimal.

Let us now consider a similar but harder problem of minimizing a convex function over a a bounded set (say a ball) with a zeroth-order oracle (i.e. a function value oracle). In this setup, with one query, we can no longer identify a subregion of the current feasible region where the function values are worse than the function value at the point we made the query. A first approach to tackle this problem is the following. Build a small regular simplex centered at the origin of the ball and query the function at its vertices. Assume the maximal function value occurs at vertex y', then by convexity of the function one can conclude that the cone generated by reflecting the simplex around y' is a region where the function values are bad. Since we have identified a bad region of the feasible set we would like to discard it and keep iterating our method, unfortunately what remains of the ball when we discard

the cone is a non-convex set we can not keep iterating the method. To try to fix the previous one could try to find the minimum volume enclosing ellipsoid of the non-convex set and keep iterating. Unfortunately this does not work since the minimum volume enclosing ellipsoid will not have sufficiently small volume [20]. The reason this occurs is that the angle of the cone generated by reflecting the simplex around y' is not wide enough. In [20] the authors fix the previous by constructing a pyramid (with wide enough angle) with y' as its apex and sample the vertices of the pyramid. If we are lucky enough and y' has the maximal function value among all the vertices of the pyramid, we can then discard the cone generated by reflecting the pyramid around y' and enclose that region in the minimum volume ellipsoid. However, if we were not lucky enough and y' did not have the maximal function value then, Nemirovski and Yudin [20], show that by repeatedly building a new pyramid with apex at the point with maximal function value we will identify a bad region after building not too many pyramids. It is not to hard to see that the previous approach may work even if we have a noisy-zeroth-order oracle, as long as the noise is not too large. The previous approach describes an optimization procedure but by itself it does not guarantee low regret. However, by incorporating center points as done in [2], sublinear regret can be achieved. Due to a lack of space the algorithm and its analysis can be found in the appendix. The main results from this section are the following.

Theorem 6. *Algorithm 2 (d-D) run with parameters* $c_1 \geq 64, c_2 \leq 1/32$ *and*

$$\Delta_\tau(\gamma) = \big(\frac{6c_1 d^4}{c_2^2} + 3\big)\gamma, \quad \bar{\Delta}_\tau(\gamma) = \big(\frac{6c_1 d^4}{c_2^2} + 5\big)\gamma,$$

guarantees that with probability at least $1 - \frac{1}{T}$

$$\bar{\mathcal{R}}_T \leq \tilde{O}(\frac{d^{16}}{\alpha^2\sqrt{T}}).$$

Theorem 7. *Let* $f(\cdot, \xi)$ *be strongly convex with parameter* $\beta > 0$ *for any* $\xi \in \Xi$*, Algorithm 2 (d-D) run with the same parameters as in Theorem 6 guarantees that with probability at least* $1 - \frac{3}{T}$

$$\mathcal{R}_T \leq \tilde{O}(\frac{d^8}{\alpha^3\beta^{1/2}T^{1/4}}).$$

8 Extension to More General Risk Measures

In Sections 6 and 7 we developed regret minimization algorithms suitable for decision makers who are risk averse, where the notion of risk was measured using the $CVaR_\alpha$. In this section we extend our results to more general risk measures. We slightly modify the setup from Section 4. Now, we assume ξ is a discrete random variable supported in Ξ with $|\Xi| = N$. That is, there are N scenarios. Moreover we assume that each scenario has the same probability of occurring. Let $X \subset \mathbb{R}^d$ be a convex and compact set. Let $f : X \times \Xi \to \mathbb{R}$ be a convex function in the first argument for every $\xi \in \Xi$. Let f satisfy $||\nabla f(x, \xi)|| \leq G$ for every $\xi \in \Xi$ and every $x \in X$. Additionally, we assume $0 \leq f(x, \xi) \leq 1$ for every $x \in X$ and every $\xi \in \Xi$. We consider some law invariant, coherent and comonotone risk measure $\rho(\cdot)$ (see next subsection). Our goal now is to obtain vanishing pseudo-ρ-regret

$$\bar{\mathcal{R}}_T^\rho := \frac{1}{T}\sum_{t=1}^T \rho[F](x_t) - \frac{1}{T}\min_{x\in X}\sum_{t=1}^T \rho[F](x),$$

and ρ-regret

$$\mathcal{R}_T^\rho := \rho[\{f_t(x_t)\}_{t=1}^T] - \min_{x\in X}\rho[\{f_t(x_t)\}_{t=1}^T].$$

In this section we will show that by using the Kusuoka Representation Theorem along with the ideas we developed earlier we can obtain vanishing $\bar{\mathcal{R}}_T^\rho$ and $\mathcal{R}_T^\rho$.

8.1 Kusuoka Representation of Risk Measures

Before presenting the algorithms we present some necessary definitions and well known results.

Definition 1. *A risk measure* $\rho : \mathcal{X}(\Omega, 2^\Omega, P) \to \mathbb{R}$ *is coherent if for every* $X_1, X_2 \in \mathcal{X}$ *it is:*

- *Normalized,* $\rho(0) = 0$.
- *Monotone,* $X_1 \leq X_2 \implies \rho(X_1) \leq \rho(X_2)$.
- *Superadditive,* $\rho(X_1) + \rho(X_2) \leq \rho(X_1 + X_2)$.
- *Positive homogenous,* $\rho(\lambda X_1) = \lambda\rho(X_1), \forall \lambda > 0$.
- *Translation invariant,* $\rho(X_1 + c) = \rho(X_1) + c$.

Moreover, we say ρ is *law invariant* if $\rho(X_1)$ depends only on the distribution of X_1. Additionally, we say ρ is *comonotone additive* if $\rho(X_1 + X_2) = \rho(X_1) + \rho(X_2)$.

It is well known [1] that $CVaR$ is a coherent risk measure. Indeed many risk measures can be expressed as functions of $CVaR$ [23]. We present a special case of the Kusuoka representation theorem that will be useful later.

Lemma 4. *[22] Consider a finite probability space* $(\Omega, 2^\Omega, P)$*, with* $\Omega = \{\omega_1, ..., \omega_N\}$*, and* $P(\omega_n) = \frac{1}{N}$ *for all* $n = 1, ..., N$*. Then, a mapping* $\rho : \mathcal{X}(\Omega, 2^\Omega, P) \to \mathbb{R}$ *is a law invariant coherent and comonotone additive*

risk measure if and only if it has a Kusuoka representation of the form

$$\rho(X) = \sum_{n=1}^{N} \mu_n CVaR_{\frac{n}{N}}(X), \quad \forall X \in \mathcal{X} \tag{6}$$

where $\mu \in [0,1]^N$ *and* $||\mu||_1 = 1$.

[23] give examples on how the Kusuoka representation theorem can be used, in particular how to write the following risk measures as mixtures of $CVaR$'s. We refer the reader to their paper for the details.

- $\rho(Z) := \inf_{t\in\mathbb{R}}\{t + c||[Z-t]_+||_p\}, \quad \forall Z \in \mathcal{L}^p(\omega, \mathcal{F}, P)$ with $c > 1$ and $1 < p < \infty$.
- $\rho(Z) := \mathbb{E}[Z] + \lambda||[Z - \mathbb{E}[Z]]_+||$ for $p \geq 1$ and $0 \leq \lambda \leq 1$.

8.2 Algorithms

We define for every $t = 1, ..., T$, function $\mathcal{G}_t(x, z) : X \times Z \to \mathbb{R}$, with $Z := [0,1]^N$, as

$$\mathcal{G}_t(x,z) := \sum_{n=1}^{N} \mu_n (z_n + \frac{1}{n/N}[f_t(x) - z_n]_+)$$

for some $\mu \in [0,1]^N, \mu \geq 0, ||\mu||_1 = 1$. For convenience we write $\mathcal{L}_n^t(x,z) := z_n + \frac{1}{n/N}[f_t(x) - z_n]_+$ for $n = 1, ..., N$. Notice that for any $x \in X$, after taking expectation with respect to ξ and plugging the minimizer of every individual term $\mathcal{L}_n^t$ we end up with the Kusuoka representation of a law invariant, coherent and commonotone risk measure. Let μ be the vector corresponding to the Kusuoka representation of our risk measure of interest ρ (see Equation (6)). Algorithm 3, a generalization of Algorithm 1 that uses functions $\mathcal{G}_t$ instead of $\mathcal{L}_t$ can be found in the appendix. We have the following guarantees for Algorithm 3.

Theorem 8. *Algorithm 3 with* $\eta = O(\frac{1}{dN^{3/2}T^{3/4}})$ *and* $\delta = O(\frac{N^{1/2}}{T^{1/4}})$*guarantees*

$$\mathbb{E}[\bar{\mathcal{R}}_T^\rho] \leq O(\frac{dN^{3/2}}{T^{1/4}}),$$

where the expectation is taken over the random draw of functions and the internal randomization of the algorithm.

Theorem 9. *Let* $f(\cdot, \xi)$ *be strongly convex with parameter* $\beta > 0$ *for all* $\xi \in \Xi$. *Algorithm 3, run with the same parameters as in Theorem 8, guarantees*

$$\mathbb{E}[\mathcal{R}_T^\rho] \leq O(\frac{d^{1/2}N^{7/4}}{\beta^{1/2}T^{1/8}}),$$

where the expectation is taken over the random draw of functions and the internal randomization of the algorithm.

To obtain a better dependence on the number of rounds T, Algorithm 2 (in both cases, $d = 1$ and $d > 1$) can be modified to solve this more general problem. The only modification is that we will sample $\tilde{O}(\frac{N^2 \ln(\sqrt{N}T)}{\gamma})$ times a point to build a γ-CI for $\rho[F](x)$ for any $x \in X$. Let this modification of Algorithm 2 be Algorithm 4. We have the following guarantees.

Theorem 10. *Algorithm 4 run with the right parameters guarantees that with probability at least* $1 - \frac{1}{T}$

$$\bar{\mathcal{R}}_T^\rho \leq \tilde{O}(\frac{N^2 d^{16}}{\sqrt{T}}).$$

Theorem 11. *Let* $f(\cdot, \xi)$ *be strongly convex with parameter* $\beta > 0$ *for all* $\xi \in \Xi$, *Algorithm 4 run with the right parameters guarantees that with probability at least* $1 - \frac{3}{T}$

$$\mathcal{R}_T^\rho \leq \tilde{O}(\frac{N^3 d^8}{\beta^{1/2}T^{1/4}}).$$

The proofs of these theorems can be found in the appendix.

9 Acknowledgements

Adrian Rivera Cardoso was supported in part by a TRIAD-NSF grant (award 1740776).

References

[1] C. Acerbi and D. Tasche. On the coherence of expected shortfall. *Journal of Banking & Finance*, 26(7):1487–1503, 2002.

[2] A. Agarwal, D. P. Foster, D. J. Hsu, S. M. Kakade, and A. Rakhlin. Stochastic convex optimization with bandit feedback. In *Advances in Neural Information Processing Systems*, pages 1035–1043, 2011.

[3] K. Ball. An elementary introduction to modern convex geometry. *Flavors of geometry*, 31:1–58, 1997.

[4] A. Ben-Tal, E. Hazan, T. Koren, and S. Mannor. Oracle-based robust optimization via online learning. *Operations Research*, 63(3):628–638, 2015.

[5] S. Bubeck, R. Eldan, and Y. T. Lee. Kernel-based methods for bandit convex optimization. *arXiv preprint arXiv:1607.03084*, 2016.

[6] N. Cesa-Bianchi and G. Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[7] T. M. Cover. Universal portfolios. *Mathematical finance*, 1(1):1–29, 1991.

[8] E. Even-Dar, M. Kearns, and J. Wortman. Risk-sensitive online learning. In *ALT*, pages 199–213. Springer, 2006.

[9] A. D. Flaxman, A. T. Kalai, and H. B. McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 385–394. Society for Industrial and Applied Mathematics, 2005.

[10] B. for International Settlements. Basel iii: international regulatory framework for banks.

[11] N. Galichet, M. Sebag, and O. Teytaud. Exploration vs exploitation vs safety: Risk-aware multi-armed bandits. In *Asian Conference on Machine Learning*, pages 245–260, 2013.

[12] D. Goldfarb and M. J. Todd. Modifications and implementation of the ellipsoid algorithm for linear programming. *Mathematical Programming*, 23 (1):1–19, 1982.

[13] E. Hazan and S. Kale. An optimal algorithm for stochastic strongly-convex optimization. *arXiv preprint arXiv:1006.2425*, 2010.

[14] E. Hazan and S. Kale. Online submodular minimization. *Journal of Machine Learning Research*, 13(Oct):2903–2922, 2012.

[15] E. Hazan and Y. Li. An optimal algorithm for bandit convex optimization. *arXiv preprint arXiv:1603.04350*, 2016.

[16] E. Hazan et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

[17] D. P. Helmbold, R. E. Schapire, Y. Singer, and M. K. Warmuth. On-line portfolio selection using multiplicative updates. *Mathematical Finance*, 8 (4):325–347, 1998.

[18] A. Kalai and S. Vempala. Efficient algorithms for universal portfolios. *Journal of Machine Learning Research*, 3(Nov):423–440, 2002.

[19] O.-A. Maillard. Robust risk-averse stochastic multi-armed bandits. In *International Conference on Algorithmic Learning Theory*, pages 218–233. Springer, 2013.

[20] A. Nemirovskii, D. B. Yudin, and E. R. Dawson. Problem complexity and method efficiency in optimization. 1983.

[21] Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.

[22] N. Noyan and G. Rudolf. Kusuoka representations of coherent risk measures in general probability spaces. *Annals of Operations Research*, 229(1): 591–605, 2015.

[23] A. Pichler and A. Shapiro. Uniqueness of kusuoka representations. *arXiv preprint arXiv:1210.7257*, 2012.

[24] R. T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. In *Journal of Risk*. Citeseer, 2000.

[25] A. Sani, A. Lazaric, and R. Munos. Risk-aversion in multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 3275–3283, 2012.

[26] S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan. Stochastic convex optimization.

[27] S. Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

[28] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on stochastic programming: modeling and theory*. SIAM, 2009.

[29] E. Takimoto and M. K. Warmuth. Path kernels and multiplicative updates. *Journal of Machine Learning Research*, 4(Oct):773–818, 2003.

[30] S. Vakili, K. Liu, and Q. Zhao. Deterministic sequencing of exploration and exploitation for multi-armed bandit problems. *IEEE Journal of Selected Topics in Signal Processing*, 7(5):759–767, 2013.

[31] J. Y. Yu and E. Nikolova. Sample complexity of risk-averse bandit-arm selection. In *IJCAI*, pages 2576–2582, 2013.

[32] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 928–936, 2003.