Online Auctions and Multi-scale Online Learning [*]

Sébastien Bubeck[†], Nikhil R. Devanur[†], Zhiyi Huang[‡] and Rad Niazadeh[§]

[†]Microsoft Research, Redmond,
[‡]Department of Computer Science, The University of Hong Kong,
[§]Department of Computer Science, Cornell University.

May 30, 2017

Abstract

We consider revenue maximization in online auctions and pricing. A seller sells an identical item in each period to a new buyer, or a new set of buyers. For the online posted pricing problem, we show regret bounds that scale with the *best fixed price*, rather than the range of the values. We also show regret bounds that are *almost scale free*, and match the offline sample complexity, when comparing to a benchmark that requires a *lower bound on the market share*. These results are obtained by generalizing the classical learning from experts and multi-armed bandit problems to their *multi-scale* versions. In this version, the reward of each action is in a *different range*, and the regret w.r.t. a given action scales with its *own range*, rather than the maximum range.

1 Introduction

Consider the following revenue maximization problem in a repeated setting, called the *online posted pricing* problem. In each period, the seller has a single item to sell, and a new prospective buyer. The seller offers to sell the item to the buyer at a given price; the buyer buys the item if and only if the price is below his private valuation for the item. The private valuation of the buyer itself is never revealed to the seller. How should a monopolistic seller iteratively set the prices if he wishes to maximize his revenue? What if he also cares about market share?

Estimating price sensitivities and demand models in order to optimize revenue and market share is the bedrock of econometrics. The emergence of online marketplaces has enabled sellers to costlessly change prices, as well as collect huge amounts of data. This has renewed the interest in understanding best practices for data driven pricing. The extreme case of this when the price is updated for each buyer is the online pricing problem described above; one can always use this for less frequent price updates. Moreover this problem is intimately related to classical experimentation and estimation procedures.

This problem has been studied from an *online learning* perspective, as a variant of the *multi-armed bandit* problem. The revenue of a pricing algorithm is compared to the revenue of the best fixed posted price, in hindsight, and the difference between the two, called the *regret*, is analyzed. No assumption is made on the distribution of values; the regret bounds are required to hold for the *worst case* sequence of values. Blum et al. (2004) assume that the buyer valuations are in $[1, h]$, and show

[*]Zhiyi Huang is supported by a RGC grant HKU17202115E. Rad Niazadeh is supported by a NSF grant CCF-1512964 and by a Google Ph.D. Fellowship. Part of this work were completed while Zhiyi was visiting and Rad was doing an internship at Microsoft Research Redmond.

the following multiplicative + additive bound on the regret: for any $\epsilon \in (0,1)$, the regret is at most ϵ times the revenue of the optimal price, $+ \; O(\epsilon^{-2} h \log h \log\log h)$. Blum and Hartline (2005) show that the additive factor can be made to be $O(\epsilon^{-3} h \log\log h)$, trading off a $\log h$ factor for an extra ϵ^{-1} factor.

An undesirable aspect of these bounds is that they scale *linearly with* h; this is particularly problematic when h is an estimate and we might set it to be a generous upper bound on the range of prices we wish to consider. A typical use case is when the same algorithm is used for many different products, with widely varying price ranges. We may not be able to manually tune the range for each product separately.

This dependency on h seems unavoidable, as is reflected by the lower bounds for the problem. (Lower bounds are discussed later in the introduction.) Yet, somewhat surprisingly, **our first contribution in this paper is to show that we can replace h by the best fixed price**[1] (that is used in the definition of the benchmark). In particular, we show that the additive bound can be made to be $O(\epsilon^{-2} p^* \log h)$, where p^* is the best fixed price in hindsight. This allows us to use a very generous estimate for h; we only lose a $\log h$ factor. The algorithm balances exploration probabilities of different prices carefully and automatically zooms in on the relevant price range. This does not violate known lower bounds, since in those instances p^* is close to h.

Bar-Yossef et al. (2002), Blum et al. (2004), and Blum and Hartline (2005) also consider the "full information" version of the problem, or what we call the *online auction* problem, where the valuations of the buyers are revealed to the algorithm after the buyer has made a decision. Such information may be available in a context where the buyers have to bid for the items, and are awarded the item if their bid is above a hidden price. In this case, the additive term can be improved to $O(\epsilon^{-1} h \log(\epsilon^{-1}))$, which is tight. Once again, *we show that h can be replaced with p^**; in particular, we show that the additive term can be made to be $O(\epsilon^{-1} p^* \log(h\epsilon^{-1}))$.

1.1 Purely multiplicative bounds and sample complexity

The regret bounds mentioned above can be turned into a purely multiplicative factor in the following way: for any $\epsilon > 0$, the algorithm is guaranteed to get a $1 - O(\epsilon)$ fraction of the best fixed price revenue, provided the number of periods $T \geq E/\epsilon$, where E is the additive term in the regret bounds above. This follows from the observation that a revenue of T is a lower bound on the best fixed price revenue. Call the number of periods required to get a $1-\epsilon$ multiplicative approximation (as a function of ϵ) as the *convergence rate* of the algorithm.

A $1-\epsilon$ multiplicative factor is also the target in the recent line of work on the *sample complexity* of auctions started by Dhangwatnotai et al. (2014); Cole and Roughgarden (2014). (We give a more comprehensive discussion of this line of work in Section 1.3.) Here, *i.i.d.* samples of the valuations are given from a *fixed but unknown distribution*, and the goal is to find a price such that its revenue w.r.t. the hidden distribution is a $1-\epsilon$ fraction of the optimum revenue for this distribution. The sample complexity is the minimum number of samples needed to guarantee this (as a function of ϵ).

The sample complexity and the convergence rate (for the full information setting) are closely related to each other. The sample complexity is always smaller than the convergence rate: the problem is easier because of the following.

1. The valuations are i.i.d. in case of sample complexity whereas they can be arbitrary (worst case) in case of convergence rate.

[1] Standard bounds allow regret to depend on the loss of the best action instead of the worst case loss. However, even such bounds still depend linearly on the range of the losses, and thus they would not allow to replace h by the best fixed price.

2. Sample complexity corresponds to an offline problem: you get all the samples at once. Convergence rate corresponds to an online problem: you need to decide what to do on a given valuation without knowing what valuations arrive in the future.

This is formalized in terms of an *online to offline reduction* [folklore] which shows that a convergence rate upper bound can be automatically translated to a sample complexity upper bound. This lets us convert sample complexity lower bounds into lower bounds on the convergence rate, and in turn into lower bounds on the additive error E in an additive + multiplicative regret bound. E.g., the additive error for the online auction problem (and hence also for the posted pricing problem[2]) cannot be $o(h\epsilon^{-1})$ (Huang et al., 2015). Moreover, it is insightful to compare convergence rates we show with *the best known sample complexity upper bound; proving better convergence rates would mean improving these bounds as well.*

A natural target convergence rate for a problem is therefore the corresponding sample complexity, but achieving this is not always trivial. An interesting version of the sample complexity bound for auctions did not have an analogous convergence rate bound. This version takes into account *both revenue and market share*, and surprisingly, gets sample complexity bounds that are *scale free*; there is no dependence on h, which means it works for unbounded valuations! For any $\delta \in (0, 1)$, the best fixed price benchmark is relaxed to ignore those prices whose market share (or equivalently probability of sale) is below a δ fraction; as δ increases the benchmark is lower. This is a meaningful benchmark since in many cases revenue is not the only goal, even if you are a monopolist. A more reasonable goal is to maximize revenue subject to the constraint that the market share is above a certain threshold. What more, this gives a sample complexity of $O(\epsilon^{-2}\delta^{-1}\log(\delta^{-1}\epsilon^{-1}))$ (Huang et al., 2015). In fact δ can be set to h^{-1} without loss of generality, when the values are in $[1, h]$,[3] and the above bound then matches the sample complexity w.r.t. the best fixed price revenue. In addition, this bound gives a precise interpolation: as the target market share δ increase, the number of samples needed decreases almost linearly.

The second contribution of this paper is to show a convergence rate that almost matches the above sample complexity, for the full information setting. We have a mild dependence on h; the rate is proportional to $\log\log h$. Further, we also show a near optimal convergence rate for the posted pricing problem.[4]

Multiple buyers: All of our results in the full information (online auction) setting extend to the multiple buyer model. In this model, in each time period, a new set of n buyers compete for a single item. The seller runs a truthful auction that determines the winning buyer and his payment. The benchmark here is the set of all "Myerson-type" mechanisms. These are mechanisms that are optimal when each period has n buyers of potentially different types, and the value of each buyer is drawn independently from a type dependent distribution. In fact, our convergence rates also imply new sample complexity bounds for these problems (except that they are not computationally efficient).

The various bounds and comparison to previous work are summarized in Tables 1 and 2.

[2] We conjecture that the lower bound for the posted pricing problem should be worse by a factor of ϵ^{-1}, since one needs to explore about ϵ^{-1} differnet prices.

[3] When the values are in $[1, h]$, we can guarantee a revenue of T by posting a price of 1, and to beat this, any other price (and in particular a price of h) would have to sell at least T/h times.

[4] Unfortunately, we cannot yet guarantee that our online algorithm itself gets a market share of δ, although we strongly believe that it does. Showing such bounds on the market share of the algorithm is an important avenue for future research.

	Lower bound	Upper bound		
		Best known (Sample complexity)	Best known (Convergence rate)	This paper (Thm. 2.5)
Online single buyer auction	$\Omega\big(\frac{h}{\epsilon^2}\big)$ *	$\tilde{O}\big(\frac{h}{\epsilon^2}\big)$ †	$\tilde{O}\big(\frac{h}{\epsilon^2}\big)$ †	$\tilde{O}\big(\frac{p^*}{\epsilon^2}\big)$
Online posted pricing	$\Omega\big(\max\{\frac{h}{\epsilon^2}, \frac{1}{\epsilon^3}\}\big)$ *§	-	$\tilde{O}\big(\frac{h}{\epsilon^3}\big)$ †	$\tilde{O}\big(\frac{p^*}{\epsilon^3}\big)$
Online multi buyer auction	$\Omega\big(\frac{h}{\epsilon^2}\big)$ *	$O\big(\frac{nh}{\epsilon^3}\big)$ ‡	-	$\tilde{O}\big(\frac{nh}{\epsilon^3}\big)$

* Huang et al. (2015); † Blum et al. (2004); ‡ Devanur et al. (2016); Gonczarowski and Nisan (2017); § Kleinberg and Leighton (2003).

Table 1: Number of rounds/samples needed to get a $1-\epsilon$ approximation to the best offline price/mechanism. Sample complexity is for the offline case with i.i.d. samples from an unknown distribution. Convergence rate is for the online case with a worst case sequence. Sample complexity is always no larger than the convergence rate. Lower bounds hold for sample complexity too, except for the online posted pricing problem for which there is no sample complexity version. The additive + multiplicative regret bounds are converted to convergence rates by dividing the additive error by ϵ. In the last row, n is the number of buyers. In the last column, p^* denotes the optimal price.

	Lower bound (Sample complexity)	Upper bound	
		Best known (Sample complexity)	This paper (Thm. 2.6)
Online single buyer auction	$\Omega\big(\frac{1}{\epsilon^2\delta}\big)$ *	$\tilde{O}\big(\frac{1}{\epsilon^2\delta}\big)$ *	$\tilde{O}\big(\frac{1}{\epsilon^2\delta}\big)$
Online posted pricing	$\Omega\big(\max\{\frac{1}{\epsilon^2\delta}, \frac{1}{\epsilon^3}\}\big)$ *†	-	$\tilde{O}\big(\frac{1}{\epsilon^4\delta}\big)$
Online multi buyer auction	$\Omega\big(\frac{1}{\epsilon^2\delta}\big)$ *	-	$\tilde{O}\big(\frac{n}{\epsilon^3\delta}\big)$

* Huang et al. (2015); † Kleinberg and Leighton (2003).

Table 2: Sample complexity & convergence rate w.r.t. the opt mechanism/price with market share $\geq \delta$.

1.2 Multi-scale online learning

The main technical ingredients in our results are variants of the classical problems of learning from expert advice and multi-armed bandit. We introduce the multi-scale versions of these problems, where each action has its reward bounded in a different range. **Our third contribution is to give an algorithm for this problem whose regret w.r.t. a certain action scales with the range of rewards for that particular action.** To contrast, the regret bounds in the standard versions scale with the maximum range. We expect such bounds to be of independent interest.

The multi-scale versions of these problems exhibit subtle variations that don't appear in the standard versions. First of all, our applications to auctions and pricing has non-negative rewards, and this actually makes a difference. For both the expert and the bandit versions, the minimax regret bounds for non-negative rewards are *provably better* than those when rewards could be negative. Further, for the bandit version, we can prove a better bound if we only require the bound to hold w.r.t. the *best* action, rather than *all* actions (for non-negative rewards). The various regret bounds and comparison to standard bounds are summarized in Tables 3.

We use algorithms based on online (stochastic) mirror descent (OSMD) (Bubeck, 2011), with a weighted negative entropy as the Legendre function. This framework gives regret bounds in terms of a "local norm" as well as an "initial divergence", which we then bound differently for each version of the problem. In the technical sections we highlight how the subtle variations arise as a result of different

	Standard regret bound $O(\cdot)$	Multi-scale bound (this paper)	
		Upper bound $O(\cdot)$	Lower bound $\Omega(\cdot)$
Experts/non-negative	$c_{\max}\sqrt{T\log(k)}$ *	$c_i\sqrt{T\log(kT)}$	$c_i\sqrt{T\log(k)}$
Bandits/non-negative	$c_{\max}\sqrt{Tk\log(k)}$ †	$c_iT^{\frac{2}{3}}(k\log(kT))^{\frac{1}{3}}$	$c_i\sqrt{TK}$
		$c_{i^*}\sqrt{Tk\log(k)}$, i^* is the best action	-
Experts/symmetric	$c_{\max}\sqrt{T\log(k)}$ *	$c_i\sqrt{T\log(k\cdot\frac{c_{\max}}{c_{\min}})}$	$c_i\sqrt{T\log(k)}$
Bandits/symmetric	$c_{\max}\sqrt{Tk\log(k)}$ †	$c_i\sqrt{Tk\cdot\frac{c_{\max}}{c_{\min}}\log(kT\cdot\frac{c_{\max}}{c_{\min}})}$	$c_i\sqrt{Tk\cdot\frac{c_{\max}}{c_{\min}}}$

* Freund and Schapire (1995); † Auer et al. (1995).

Table 3: Pure-additive regret bounds for non-negative rewards, i.e. when reward of any action i at any time is in $[0, c_i]$, and symmetric range rewards, i.e. when reward of any action i at any time is in $[-c_i, c_i]$ (suppose T is the time horizon, A is the actions set, and k is the number of actions).

techniques used to bound these two terms.

Foster et al. (2017) also consider the multi-scale online learning problem motivated by a model selection problem. They consider additive bounds, for the symmetric case, for full information, but not bandit feedback. Their regret bounds are not comparable to ours in general; our bounds are better for the pricing/auction applications we consider, and their bounds are better for their application.

1.3 Other related work

The online pricing problem, also called *dynamic pricing*, is a much studied topic, across disciplines such as operations research and management science (Talluri and Van Ryzin, 2006), economics (Segal, 2003), marketing, and of course computer science. The multi-armed bandit approach to pricing is particularly popular. See den Boer (2015) for a recent survey on various approaches to the problem.

Kleinberg and Leighton (2003) consider the online pricing problem, under the assumption that the values are in $[0, 1]$, and considered purely additive factors. They showed that the minimax additive regret is $\tilde{\Theta}(T^{2/3})$, where T is the number of periods. This is similar in spirit to regret bounds that scale with h, since one has to normalize the values so that they are in $[0, 1]$. The finer distinction about the magnitude of the best fixed price is absent in this work. Recently, Syrgkanis (2017) also consider the online auction problem, with an emphasis on a notion of "oracle based" computational efficiency. They assume the values are all in $[0, 1]$ and don't consider the scaling issue that we do; this makes their contribution orthogonal to ours.

Starting with Dhangwatnotai et al. (2014), there has been a spate of recent results analyzing the sample complexity of pricing and auction problems. Cole and Roughgarden (2014) and Devanur et al. (2016) consider multiple buyer auctions with regular distributions (with unbounded valuations) and give sample complexity bounds that are polynomial in n and ϵ^{-1}, where n is the number of buyers. Morgenstern and Roughgarden (2015) consider arbitrary distributions with values bounded by h, and gave bounds that are polynomial in n, h, and ϵ^{-1}. Roughgarden and Schrijvers (2016); Huang et al. (2015) give further improvements on the single- and multi-buyer versions respectively; tables 1 and 2 give a comparison of these results with our bounds, for the problems we consider. The dynamic pricing problem has also been studied when there are a given number of copies of the item to sell (limited supply) (Agrawal and Devanur, 2014; Babaioff et al., 2015; Badanidiyuru et al., 2013; Besbes and Zeevi, 2009). There are also variants where the seller interacts with the same buyer repeatedly, and the buyer can strategize to influence his utility in the future periods (Amin et al., 2013; Devanur et al., 2014).

2 Model and Main Results

We consider a variety of online algorithmic problems that are all parts of the *multiscale online learning* framework. We start by defining this framework and expressing our results in terms of *action-specific* regret bounds for this general problem. Next, we investigate different auction design problems that are covered by this framework, and show how to get multiplicative cum additive approximations for these problems by the help of the multi-scale learning framework. We then consider competing with δ*-guarded benchmarks* and show how our algorithms get pure multiplicative approximations with respect to these benchmarks. We can then translate the convergence rate of our online algorithms to sample complexity of auctions to 1) generalize many sample complexity upper-bounds to the online adversarial auction settings, 2) compare our bounds with the known sample complexity lower-bounds, and 3) design new algorithms achieving near-optimal sample complexity bounds for the offline Bayesian auction problem.

2.1 Multi-scale online learning framework

Our multi-scale online learning framework is basically the classical learning from expert advice problem (under full-information) or multi-armed bandit problem (under partial-information). The main difference is that the *range* of different experts/arms could be different. Suppose there is a set of actions A. The problem proceeds in T rounds, and in each round $t \in [T]$:[5]

- The algorithm picks an action $i_t \in A$
- The adversary picks a reward function $\mathbf{g}(t)$ simultaneously, where action i has reward $g_i(t)$.
- The algorithm gets the reward $g_{i_t}(t)$.
- In the *full information* setting, the algorithm sees the entire reward function $\mathbf{g}(t)$. In the *bandit* setting, the algorithm sees only its own reward, $g_{i_t}(t)$.

The total reward of the algorithm is denoted by

$$G_{\text{ALG}} := \textstyle\sum_{t=1}^{T} g_{i_t}(t).$$

The standard "best fixed action" benchmark is

$$G_{\text{MAX}} := \max_{i\in A} \textstyle\sum_{t=1}^{T} g_i(t).$$

We consider both full-information and the bandit setting:

- **Multi-scale experts:** The action set is countable. If the action set is finite of size k, we identify $A = [k]$. The reward $\mathbf{g}(t)$ is such that for all $i \in A$, $g_i(t) \in [0, c_i]$. The entire reward function $\mathbf{g}(t)$ is revealed to the algorithm after round t.
- **Multi-scale bandit learning:** The same as before, in the bandit setting.

We prove *action-specific regret* bounds, which we call also *multi-scale regret guarantees.* Towards this end, we define the following quantities.

$$G_i \quad := \quad \textstyle\sum_{t\in[T]} g_i(t) \ , \tag{1}$$

$$\text{REGRET}_i \quad := \quad G_i - G_{\text{ALG}} \ . \tag{2}$$

[5] We use the notation $[n] := \{1, 2, \ldots, n\}$, for any $n \in \mathbb{N}$.

The regret bound w.r.t. action i, i.e., an upper bound on $\mathbb{E}\left[\textsc{regret}_i\right]$, depends on the range c_i, as well as any *prior* distribution $\boldsymbol{\pi}$ over the action set A; this way, we can handle countably many actions. Let $c_{\min} = \inf_{i\in A} c_i$ and $c_{\max} = \sup_{i\in A} c_i$ (if applicable) be the minimum and the maximum range. We first state a version of the regret bound which is parameterized by $\epsilon > 0$; such bounds are stronger than $\sqrt{T}$ type bounds which are more standard.

Theorem 2.1. *There exists an algorithm for the multi-scale experts problem that takes as input any distribution $\boldsymbol{\pi}$ over A, the ranges c_i, $\forall\ i \in A$, and a parameter $0 < \epsilon \leq 1$, and satisfies:*

$$\forall i \in A: \quad \mathbb{E}\left[\textsc{regret}_i\right] \leq \epsilon \cdot G_i + O\left(\frac{1}{\epsilon}\log\left(\frac{1}{\epsilon\pi_i}\right)\cdot c_i\right) \tag{3}$$

Compare this to what you get by using the standard analysis for the experts problem (Arora et al., 2012), where the second term in the regret bound is $O\big(\frac{1}{\epsilon}\log(k)\cdot c_{\max}\big)$. Choosing $\boldsymbol{\pi}$ to be the uniform distribution in the above theorem gives $O\big(\frac{1}{\epsilon}\log\left(\frac{k}{\epsilon}\right)\cdot c_i\big)$. Also, one can compare the pure-additive version of this bound with the classic pure-additive regret bound $O\big(c_{\max}\cdot\sqrt{T\log(k)}\big)$ for the experts problem by setting $\epsilon = \sqrt{\frac{\log(kT)}{T}}$ (Corollary 2.2).

Corollary 2.2. *There exists an algorithm for the multi-scale experts problem that takes as input the ranges c_i, $\forall\ i \in A$, and satisfies:*

$$\forall i \in A: \quad \mathbb{E}\left[\textsc{regret}_i\right] \leq O\left(c_i\cdot\sqrt{T\log(kT)}\right) \tag{4}$$

For the bandit version, we can get a similar regret guarantee, but only for the *best* action. If we require the regret bound to hold for all actions, then we can only get a weaker bound, where the second term has ϵ^{-2} instead of ϵ^{-1}. The difference between the bounds for the bandit and the full information setting is essentially a factor of k, which is unavoidable.

Theorem 2.3. *There exists an algorithm for the online multi-scale bandits problem that takes as input the ranges c_i, $\forall\ i \in A$, and a parameter $0 < \epsilon \leq 1$, and satisfies,*

- *for $i^* = \arg\max_{i\in A} G_i$,*

$$\mathbb{E}\left[\textsc{regret}_{i^*}\right] \leq \epsilon\cdot G_{i^*} + O\left(\tfrac{1}{\epsilon}k\log\left(\tfrac{k}{\epsilon}\right)\cdot c_{i^*}\right). \tag{5}$$

- *for all $i \in A$,*

$$\mathbb{E}\left[\textsc{regret}_i\right] \leq \epsilon\cdot G_i + O\left(\tfrac{1}{\epsilon^2}k\log\left(\tfrac{k}{\epsilon}\right)\cdot c_i\right). \tag{6}$$

Also, one can compute the pure-additive versions of the bounds in Theorems 2.3 by setting $\epsilon = \sqrt{\frac{k\log(kT)}{T}}$ and $\epsilon = \left(\frac{k\log(kT)}{T}\right)^{\frac{1}{3}}$ resepctively (Corollary 2.4), and compare with the pure-additive regret bound $O\big(c_{\max}\cdot\sqrt{Tk\log k}\big)$ for the adversarial multi-armed bandit problem (Auer et al., 1995).

Corollary 2.4. *There exist algorithms for the online multi-scale bandits problem that satisfies,*

- *For $i^* = \arg\max_{i\in A} G_i$,*

$$\mathbb{E}\left[\textsc{regret}_{i^*}\right] \leq O\left(c_{i^*}\cdot\sqrt{Tk\log(kT)}\right) \tag{7}$$

- *For all $i \in A$,*

$$\mathbb{E}\left[\textsc{regret}_i\right] \leq O\left(c_i\cdot T^{\frac{2}{3}}(k\log(kT))^{\frac{1}{3}}\right) \tag{8}$$

2.2 Online auction design

The auction design problems that we consider are as follows.

- **Online single buyer auction:** The action set $A = [1, h]$. The reward function is such that the adversary picks a *value* $v(t) \in [1, h]$ and for any *price* $i \in A$, the reward $g_i(t) := p \cdot \mathbf{1}(v(t) \geq i)$. This is the full information setting, where the value $v(t)$ is revealed to the algorithm after round t.
- **Online posted pricing:** The same as above, in the bandit setting. The algorithm only learns the indicator function $\mathbf{1}(v(t) \geq i_t)$ where i_t is the price it picks in round t.
- **Online multi buyer auction:** The action set is the set of all "Myerson-type" mechanisms for n buyers, for some $n \in \mathbb{N}$. (See Definition 5.1.) The adversary picks a valuation vector $\mathbf{v}(t) \in [1, h]^n$ and the reward of a mechanism M is its revenue when the valuation of the buyers is given by $\mathbf{v}(t)$; this is denoted by $\textsc{rev}_M(\mathbf{v}(t))$. The algorithm sees the full vector of valuations $\mathbf{v}(t)$.

We show how to get a multiplicative cum additive approximations for these problems with $G_{\textsc{max}}$ as the benchmark, à la Blum et al. (2004); Blum and Hartline (2005). The main improvement over these results is that the additive term scales with the best price rather than h. Let p^* be the best fixed price on hindsight, which is the price that achieves $G_{\textsc{max}}$.

Theorem 2.5. *There are algorithms for the online single buyer auction, online posted price auction, and the online multi buyer auction problems that take as input a parameter $\epsilon > 0$, and satsify $G_{\textsc{alg}} \geq (1-\epsilon)G_{\textsc{max}} - O(E)$, where respectively (for the three problems mentioned above)*

$$E = \frac{p^* \log(\log h/\epsilon)}{\epsilon}, \quad \frac{p^* \log h \log(\log h/\epsilon)}{\epsilon^2}, \quad \textit{and} \quad \frac{hn \log h \log(n \log h/\epsilon)}{\epsilon^2} .$$

Even if h is not known upfront, we can still get the similar approximation guarantee for online single buyer auction and online multi buyer auction with:

$$E = \frac{p^* \log(p^*/\epsilon)}{\epsilon}, \quad \textit{and} \quad \frac{hn \log h \log(n \log h/\epsilon)}{\epsilon^2} .$$

Bounds on the *sample complexity* of auctions imply that the first bound in this theorem is tight up to log factors: the lower bound is $h\epsilon^{-1}$ in an instance where $p^* = h$, and the best upper bound known is $h\epsilon^{-1}\log(1/\epsilon)$. We conjecture that our bound for the online posted pricing problem is tight up to log factors, and leave resolving this as an open problem. The third bound is not comparable to the best sample complexity for the multi buyer auction problem by Roughgarden and Schrijvers (2016); it is better than theirs for large ϵ (when $1/\epsilon \leq o(nh)$), and is worse for smaller ϵ (when $1/\epsilon \geq \omega(nh)$). Also, compare these to the corresponding upper bounds for the first two problems by Blum et al. (2004); Blum and Hartline (2005), which are respectively

$$\frac{h \log(1/\epsilon)}{\epsilon}, \quad \text{and } \min\left\{\frac{h \log h \log\log h}{\epsilon^2}, \frac{h \log\log h}{\epsilon^3}\right\}.$$

2.3 Competing with δ-guarded benchmarks

For the single buyer auction/pricing problem, we define a δ-guarded benchmark, for any $\delta \in [0, 1]$. This benchmark is restricted to only those prices that sell the item in at least a δ fraction of the rounds.

$$G_{\textsc{max}}(\delta) := \max\left\{\textstyle\sum_{t=1}^T g_p(t) : p \in A, \sum_{t=1}^T \mathbf{1}(v_t \geq p) \geq \delta T\right\}.$$

As observed in Footnote 3, one can replace δ with $1/h$ and get the corresponding guarantees for $G_{\textsc{max}}$ rather than $G_{\textsc{max}}(\delta)$. However, the main point of these results is to show a graceful improvement of the bounds as δ is chosen to be larger.

Multiple buyers: For the multi buyer auction problem, we define the δ-guarded benchmark as follows. For any sequence of value vectors $\mathbf{v}(1), \mathbf{v}(2), \ldots, \mathbf{v}(T)$, let $\bar{V}$ denote the largest value such that there are at least δT distinct $\mathbf{v}(t)$'s with $\max_{i \in [n]} v_i(t) \geq \bar{V}$. Define the δ-guarded benchmark to be

$$G_{\text{MAX}}(\delta) = \max_M \sum_{t=1}^{T} Rev_M \left(\min(\bar{V}\vec{\mathbf{1}}, \mathbf{v}(t)) \right),$$

where the min is to be understood to be applied co-ordinate wise, and the max is over all Myerson-type mechanisms.

We focus on purely multiplicative approximation factors when competing with $G_{\text{MAX}}(\delta)$. In particular, for any given $\epsilon > 0$, we are interested in a $1 - \epsilon$ approximation. We state our results in terms of the *convergence rate*. We say that $T(\epsilon, \delta)$ is the convergence rate of an algorithm if for all time horizon $T \geq T(\epsilon, \delta)$, we are guaranteed that $G_{\text{ALG}} \geq (1 - \epsilon) G_{\text{MAX}}(\delta)$. Our main results are as follows.

Theorem 2.6. *There are algorithms for the online single buyer auction, online posted pricing, and the online multi buyer auction problems with convergence rates respectively of*

$$O\left(\frac{\log(\log h/\epsilon)}{\epsilon^2 \delta}\right), \quad O\left(\frac{\log h}{\epsilon^4 \delta}\right), \quad \textit{and } O\left(\frac{n \log(1/\epsilon\delta) \log(n \log(1/\epsilon\delta)/\epsilon)}{\epsilon^3 \delta} + \frac{\log(\log h/\epsilon)}{\epsilon^2 \delta}\right).$$

Even if h is not known upfront, we can still get the following similar convergence rates for online single buyer auction and online multi buyer auction respectively:

$$O\left(\frac{\log(p^*/\epsilon)}{\epsilon^2 \delta}\right), \quad \textit{and} \quad O\left(\frac{n \log(1/\epsilon\delta) \log(n \log(1/\epsilon\delta)/\epsilon)}{\epsilon^3 \delta} + \frac{\log(h/\epsilon)}{\epsilon^2 \delta}\right).$$

Once again, we compare to the sample compexity bounds: our first is within a $\log \log h$ factor of the best sample complexity upper bound in Huang et al. (2015). The lower bound for the online single buyer auction is $\Omega(\delta^{-1}\epsilon^{-2})$, which is also the best lower bound known for the pricing and the multi-buyer problem.[6] For the online posted pricing problem, we conjecture that the right dependence on ϵ should be ϵ^{-3}. No sample complexity bounds for the multi-buyer problem were known before; in fact we introduce the definition of a δ-guarded benchmark for this problem.

2.4 Multi-scale online learning with symmetric range

The standard analysis for the experts and the bandit problems holds even if the range of $g_i(t)$ is $[-c_i, c_i]$, rather than $[0, c_i]$ as we have assumed. In contrast, there are subtle differences on the best acheivable multi-scale regret bounds between the non-negative and the symmetric range. We first show the following upper bound for the full information setting when the range is symmetric. This bounds follows the same style of action-specific regret bounds as in Theorem 2.1. More detailed discussion on how the choice of initial distribution $\boldsymbol{\pi}$ affects the bound is deferred to the appendix, Section A.1.

Theorem 2.7. *There exists an algorithm for the multi-scale experts problem with symmetric range that takes as input any distribution $\boldsymbol{\pi}$ over A, the ranges c_i, $\forall\ i \in A$, and a parameter $0 < \epsilon \leq 1$, and satisfies:*

$$\forall i \in A: \quad \mathbb{E}\left[\text{REGRET}_i\right] \leq \epsilon \cdot \mathbb{E}\left[\sum_{t \in [T]} \left| g_t(i) \right|\right] + O\left(\frac{1}{\epsilon} \log\left(\frac{1}{\pi_i} \cdot \frac{c_i}{c_{\min}}\right) \cdot c_i\right). \tag{9}$$

Similar to Section 2.1, we can compute the pure-additive version of the bound in Theorem 2.7 by setting $\epsilon = \sqrt{\frac{\log(k \cdot \frac{c_{\max}}{c_{\min}})}{T}}$, as in Corollary 2.2.

[6] Cole and Roughgarden (2014) show that at least a linear dependence on n is necessary when the values are drawn from a regular distribution, but as is, their lower bound needs unbounded valuations. The lower bound probably holds for "large enough h" but it is not clear if it holds for all h.

Corollary 2.8. *There exists an algorithm for the online multi-scale experts problem with symmetric range that takes as input the ranges c_i, $\forall\ i \in A$, and satisfies:*

$$\forall i \in A: \quad \mathbb{E}\left[\text{REGRET}_i\right] \leq O\left(c_i \cdot \sqrt{T \log(k \cdot \tfrac{c_{\max}}{c_{\min}})}\right) \tag{10}$$

If we compare the above regret bound with the standard $O(c_{\max}\sqrt{T \log k})$ regret bound for the experts problem, we see that we replace the dependency on $c_{\max}$ in the standard bound with $c_i\sqrt{\log(\frac{c_{\max}}{c_{\min}})}$. It is natural to ask whether we could get rid of the dependence on $\log(c_i/c_{\min})$ and show regret bound of $O(c_i\sqrt{T \log k})$, like we did for non-negative rewards. However, the next theorem shows that this dependence on $\log(c_i/c_{\min})$ in the above bound is necessary, in a weak sense: where the constant in the $O(\cdot)$ is universal and does not depend on the ranges c_i. This is because the lower bound only holds for "small" values of the horizon T, which nonetheless grows with the $\{c_i\}$s.[7]

Theorem 2.9. *There exists an action set of size k, and ranges $c_i, \forall i \in [k]$, and time horizon T, such that for all algorithms for the online multi-scale experts problem with symmetric range, there is a sequence of T gain vectors such that*

$$\exists i \in A: \quad \mathbb{E}\left[\text{REGRET}_i\right] > \tfrac{c_i}{4} \cdot \sqrt{T \log(k \cdot \tfrac{c_{\max}}{c_{\min}})}$$

We then show the following upper bound for the bandit setting when the range is symmetric. This bound also follows the same style of action-specific regret bounds as in Theorem 2.3.

Theorem 2.10. *There exists an algorithm for the multi-scale bandits problem with symmetric range that takes as input the ranges c_i, $\forall\ i \in A$, and a parameter $0 < \epsilon \leq 1/2$, and satisfies:*

$$\forall i \in A: \quad \mathbb{E}\left[\text{REGRET}_i\right] \leq O\big(\epsilon T + \tfrac{k}{\epsilon}\tfrac{c_{\max}}{c_{\min}} \log\left(\tfrac{k}{\epsilon}\tfrac{c_{\max}}{c_{\min}}\right)\big) \cdot c_i. \tag{11}$$

Also, similar to Section 2.1, we can compute the pure-additive version of the bound in Theorem 2.10 by setting $\epsilon = \sqrt{\frac{k\frac{c_{\max}}{c_{\min}}\log(kT\cdot\frac{c_{\max}}{c_{\min}})}{T}}$, as in Corollary 2.2. This bound is comparable to the standard regret bound of $O(c_{\max}\sqrt{kT \log k})$ (Auer et al., 1995) for the adversarial multi-armed bandits problem.

Corollary 2.11. *There exists an algorithm for the online multi-scale bandits problem with symmetric range that satisfies:*

$$\forall i \in A: \quad \mathbb{E}\left[\text{REGRET}_i\right] \leq O\left(c_i \cdot \sqrt{Tk \cdot \tfrac{c_{\max}}{c_{\min}} \log(kT \cdot \tfrac{c_{\max}}{c_{\min}})}\right). \tag{12}$$

Once again, for the bandit problem, the following theorem shows that this bound cannot be improved beyond log factors (to get a guarantee like that of Theorem 2.3, for instance).

Theorem 2.12. *There exists an action set of size k, and ranges $c_i, \forall i \in [k]$, such that for all algorithms for the online multi-scale bandit problem with symmetric range, for all sufficiently large time horizon T, there is a sequence of T gain vectors such that*

$$\exists i \in A: \quad \mathbb{E}\left[\text{REGRET}_i\right] > \frac{c_i}{8\sqrt{2}} \cdot \sqrt{Tk \cdot \frac{c_{\max}}{c_{\min}}}.$$

[7] For this reason we chose not to include this bound in Table 3.

Organization We start in Section 3 by showing regret upper bounds for the multi-scale experts problem with non-negative rewards (Theorem 2.1). The corresponding upper bounds for the bandit version are in section 4 (Theorem 2.3). In Section 5 we show how the multi-scale regret bounds (Theorems 2.1 and 2.3) imply the corresponding bounds for the auction/pricing problems (Theorems 2.5 and 2.6). Finally, the regret (upper and lower) bounds for the symmetric range are discussed in Section 6 (Theorems 2.7, 2.9, 2.10, and 2.12).

3 Multi-Scale Online Learning with Full Information

In this section, we look at the full information multi-scale learning problem, in which different experts have different ranges. We exploit this structure to achieve expert-specific regret bounds.

Here is a map of this section. In Section 3.1 we propose an algorithm that exploits the aforementioned structure, and later in Section 3.2 we show how this algorithm is an online mirror descent with weighted negative entropy as the Legendre function. For reward-only instances, we prove the regret bound *without* dependency on $\log(c_i/c_{\max})$ in Section 3.3.

3.1 Multi-Scale Multiplicative-Weight (MSMW) algorithm

We propose the "*Multi-Scale Multiplicative-Weight*" (MSMW) algorithm as a multiplicative-weight update style learning for our problem. The algorithm is presented in Algorithm 1. The main idea behind this algorithm is taking into account different ranges for different experts, and therefore

1. Normalizing reward of each expert accordingly, i.e. dividing the reward of expert i by c_i.
2. Projecting the updated weights accordingly, by performing a *smooth multi-scale projection* into the simplex that will be described later.

Algorithm 1 MSMW

```
input  initial distribution $\boldsymbol{\mu}$ over $A$, learning rate $0 < \eta \leq 1$.
initialize  $\mathbf{p}(1)$ such that $p_i(1) = \mu_i$ for all $i \in A$.
for $t = 1, \dots, T$ do
    Randomly pick an action drawn from $\mathbf{p}(t)$, and observe $\mathbf{g}(t)$.
    $\forall i \in A: \quad w_i(t+1) \leftarrow p_i(t) \cdot \exp(\eta \cdot \frac{g_i(t)}{c_i})$.
    Find $\lambda^*$ (e.g., binary search) s.t. $\sum_{i \in A} w_i(t+1) \cdot \exp(-\frac{\lambda^*}{c_i}) = 1$.
    $\forall i \in A: \quad p_i(t+1) \leftarrow w_i(t+1) \cdot \exp(-\frac{\lambda^*}{c_i})$.
end for
```

3.2 Equivalence to Online Mirror Descent (OMD) with weighted negative entropy

While it is possible to analyze the regret of MSMW algorithm (Algorithm 1) by using first principles (Look at the proof of Lemma 3.4 in the appendix, Section A.4), we take a different approach . We show how this algorithm is indeed an instance of Online Mirror Descent (OMD) algorithm for a particular choice of *Legendre function*.

3.2.1 Preliminaries on online mirror descent.

Fix an open convex set $\mathcal{D}$ and its closure $\bar{\mathcal{D}}$, which in our case are $\mathbb{R}^A_{>0}$ and $\mathbb{R}^A_+$ respectively, and a closed-convex action set $\mathcal{A} \subset \bar{\mathcal{D}}$, which in our case is Δ_A, i.e. the set of all probability distributions over experts in A. At the heart of an OMD algorithm there is a *Legendre* function $F : \bar{\mathcal{D}} \to \mathbb{R}$, i.e. a strictly convex function that admits continuous first order partial derivatives on $\mathcal{D}$ and $\lim_{x \to \bar{\mathcal{D}} \setminus \mathcal{D}} \|\nabla F(x)\| = +\infty$, where $\nabla F(.)$ denotes the gradient map of F. One can think of OMD as a member of *projected gradient descent* algorithms, where the *gradient update* happens in the *dual space* $\nabla F(\mathcal{D})$ rather than in primal $\mathcal{D}$, and the *projection* is defined by using the *Bregman divergence* associated with F rather than ℓ_2-distance.

Definition 3.1 (Bregman Divergence (Bubeck, 2011))**.** Given a Legendre function F over Δ_A, the Bregman divergence associated with F, denoted as $D_F : \Delta_A \times \Delta_A \to \mathbb{R}$, is defined by

$$D_F(x, y) = F(x) - F(y) - (x - y)^T \nabla F(y)$$

Definition 3.2 (Online Mirror Descent Bubeck (2011))**.** Suppose F is a Legendre function. At every time $t \in [T]$, the online mirror descent algorithm with Legendre function F selects an expert drawn from distribution $\mathbf{p}(t)$, and then updates $\mathbf{w}(t)$ and $\mathbf{p}(t)$ given rewards $\mathbf{g}(t)$ by:

Gradient update:

$$\nabla F(\mathbf{w}(t+1)) = \nabla F(\mathbf{p}(t)) + \eta \cdot \mathbf{g}(t) \Rightarrow \mathbf{w}(t+1) = (\nabla F)^{-1} \left(\nabla F(\mathbf{p}(t)) + \eta \cdot \mathbf{g}(t)\right) \tag{13}$$

Bregman projection:

$$\mathbf{p}(t+1) = \operatorname*{argmin}_{\mathbf{p} \in \Delta_A} \left(D_F(\mathbf{p}, \mathbf{w}(t+1))\right) \tag{14}$$

where $\eta > 0$ is called the learning rate of OMD.

We use the following standard regret bound of OMD (Refer to Bubeck (2011) for a thorough discussion on OMD. For completeness, a proof is also provided in the appendix, Section A.5).

Lemma 3.1. *For any learning rate parameter $0 < \eta \leq 1$ and any benchmark distribution $\mathbf{q}$ over A, the OMD algorithm with Legendre function $F(.)$ admits the following:*

$$\textstyle\sum_{t \in [T]} \mathbf{g}(t) \cdot \left(\mathbf{q} - \mathbf{p}(t)\right) \leq \frac{1}{\eta} \sum_{t \in [T]} D_F(\mathbf{p}(t), \mathbf{w}(t+1)) + \frac{1}{\eta} D_F(\mathbf{q}, \mathbf{p}(1)) \tag{15}$$

3.2.2 MSMW algorithm as an OMD

For our application, we focus on a particular choice of Legendre function that captures different learning rates proportional to c_i^{-1} for different experts, as we saw earlier in Algorithm 1. We start by defining the *weighted negative entropy* function.

Definition 3.3. Given expert-ranges $\{c_i\}_{i \in A}$, the *weighted negative entropy* is defined by

$$F(x) = \textstyle\sum_{i \in A} c_i \cdot x_i \ln(x_i) \tag{16}$$

Corollary 3.2. *It is straightforward to see $F(x) = \sum_{i \in A} c_i \cdot x_i \ln(x_i)$ is a non-negative Legendre function over $\mathbb{R}^A_+$. Moreover, $\nabla F(x)_i = c_i(1 + \ln(x_i))$ and $D_F(x, y) = \sum_{i \in A} c_i \cdot (x_i \ln(\frac{x_i}{y_i}) - x_i + y_i)$.*

We now have the following lemma that shows Algorithm 1 is indeed an OMD algorithm.

Lemma 3.3. *The MSMW algorithm, i.e. Algorithm 1, is equivalent to an OMD algorithm associated with the weighted negative entropy $F(x) = \sum_{i \in A} c_i \cdot x_i \ln(x_i)$ as its Legendre function.*

Proof Look at the gradient update step of OMD, as in Equation (13), with Legendre transform $F(x) = \sum_{i\in A} c_i \cdot x_i \ln(x_i)$. By using Corollary 3.2 we have

$$\nabla F(\mathbf{w}(t+1)) = \nabla F(\mathbf{p}(t)) + \eta \cdot \mathbf{g}(t) \Rightarrow c_i(1 + ln(w_i(t+1))) = c_i(1 + ln(p_i(t))) + \eta \cdot g_i(t) \ ,$$

and therefore, $w_i(t+1) = p_i(t) \cdot \exp(\eta \cdot \frac{g_i(t)}{c_i})$. Moreover, for Bregman projection step we have

$$\mathbf{p}(t+1) = \underset{\mathbf{p}\in\Delta_A}{\operatorname{argmin}} \left(D_F(\mathbf{p}, \mathbf{w}(t+1))\right) = \underset{\mathbf{p}\in\Delta_A}{\operatorname{argmin}} \left(\sum_{i\in A} c_i \cdot (p_i \ln(\frac{p_i}{w_i(t+1)}) - p_i + w_i(t+1))\right) \tag{17}$$

This is a convex-minimization over a convex set. To find a closed form solution, we look at the Lagrangian dual function $\mathcal{L}(\mathbf{p}, \lambda) \triangleq \sum_{i\in A} c_i \cdot (p_i \ln(\frac{p_i}{w_i(t+1)}) - p_i + w_i(t+1)) + \lambda(\sum_{i\in A} p_i - 1)$ and the Karush-Kuhn-Tucker (KKT) conditions $\nabla\mathcal{L}(\mathbf{p}^*, \lambda^*) = \mathbf{0}$. We have

$$c_i \cdot \ln(\frac{p_i^*}{w_i(t+1)}) + \lambda^* = 0 \Rightarrow p_i^* = w_i(t+1) \cdot \exp(-\frac{\lambda^*}{c_i}) \tag{18}$$

As $\sum_{i\in A} p_i^* = 1$, λ^* should be unique number s.t. $\sum_{i\in A} w_i(t+1) \cdot \exp(-\frac{\lambda^*}{c_i}) = 1$, and then $p_i(t+1) = w_i(t+1) \cdot \exp(-\frac{\lambda^*}{c_i})$. So, Algorithm 1 is equivalent to OMD with weighted negative entropy as its Legendre transform.

Lemma 3.4. *For any initial distribution $\boldsymbol{\mu}$ over A, and any learning rate parameter $0 < \eta \leq 1$, and any benchmark distribution $\mathbf{q}$ over A, the MSMW algorithm satisfies that:*

$$\sum_{i\in A} q_i \cdot G_i - \mathbb{E}\left[G_{\text{ALG}}\right] \leq \eta \sum_{t\in[T]} \sum_{i\in A} p_i(t) \frac{(g_i(t))^2}{c_i} + \frac{1}{\eta} \cdot \sum_{i\in A} c_i \left(q_i \ln\left(\frac{q_i}{\mu_i}\right) - q_i + \mu_i\right) .$$

3.3 Regret bound for non-negative rewards - proof of Theorem 2.1

Proof of Theorem 2.1 Suppose $i_{\min}$ is an action with the minimum c_i. Let $\boldsymbol{\mu} = (1-\eta) \cdot \mathbf{1}_{i_{\min}} + \eta \cdot \boldsymbol{\pi}$, and let $\mathbf{q} = (1-\eta) \cdot \mathbf{1}_i + \eta \cdot \boldsymbol{\pi}$ in Lemma 3.4. If $i \neq i_{\min}$, we get that (note that $\mu_j = q_j$ for any $j \neq i, i_{\min}$):

$$\begin{aligned}(1-\eta) \cdot G_i + \eta \cdot \sum_{j\in A} \pi_j \cdot G_j - \mathbb{E}\left[G_{\text{ALG}}\right] \leq \eta \cdot \mathbb{E}\left[G_{\text{ALG}}\right] + \frac{1}{\eta} \cdot c_i \cdot \left(q_i \ln\left(\frac{q_i}{\mu_i}\right) - q_i + \mu_i\right) \\ + \frac{1}{\eta} \cdot c_{i_{\min}} \cdot \left(q_{i_{\min}} \ln\left(\frac{q_{i_{\min}}}{\mu_{i_{\min}}}\right) - q_{i_{\min}} + \mu_{i_{\min}}\right)\end{aligned}$$

By $1 \geq q_i > \mu_i \geq \eta\pi_i$, the 2nd term on the RHS is upper bounded as:

$$\frac{1}{\eta} \cdot c_i \cdot \left(q_i \ln\left(\frac{q_i}{\mu_i}\right) - q_i + \mu_i\right) \leq \frac{1}{\eta} \cdot c_i \cdot \ln\left(\frac{1}{\eta\pi_i}\right)$$

Similarly, by $1 \geq \mu_{i_{\min}} > q_{i_{\min}} \geq 0$, the 3rd term on the RHS is upper bounded as

$$\frac{1}{\eta} \cdot c_{i_{\min}} \cdot \left(q_{i_{\min}} \ln\left(\frac{q_{i_{\min}}}{\mu_{i_{\min}}}\right) - q_{i_{\min}} + \mu_{i_{\min}}\right) \leq \frac{1}{\eta} \cdot c_{i_{\min}} \leq \frac{1}{\eta} \cdot c_i$$

Finally, note that $G_j \geq 0$ for all $j \in A$ in reward-only instances. So the LHS is lower bounded by

$$(1-\eta) \cdot G_i - \mathbb{E}\left[G_{\text{ALG}}\right] = (1-\eta) \cdot \text{REGRET}_i - \eta \cdot \mathbb{E}\left[G_{\text{ALG}}\right] .$$

Putting together we get that

$$\mathbb{E}\left[\text{REGRET}_i\right] \leq \frac{2\eta}{1-\eta} \cdot \mathbb{E}\left[G_{\text{ALG}}\right] + O\left(\frac{1}{\eta} \ln\left(\frac{1}{\eta\pi_i}\right) \cdot c_i\right) \leq 3\eta \cdot \mathbb{E}\left[G_{\text{ALG}}\right] + O\left(\frac{1}{\eta} \ln\left(\frac{1}{\eta\pi_i}\right) \cdot c_i\right) .$$

The theorem then follows by choosing $\eta = \frac{\epsilon}{3}$ and rearranging terms.

4 Multi Scale Online Learning with Bandit Feedback

In this section, we look at the bandit feedback version of multi scale online learning. Inspired by online stochastic mirror descent algorithm, we introduce *Bandit-MSMW* algorithm. Our algorithm follows the standard bandit route of using unbiased estimators for the rewards in a full information strategy (in this case MSMW). We also mix the MSMW distribution with an extra uniform exploration, and use a tailored initial distribution for our multi-scale learning setting.

Here is a map of this section. In Section 4.1 we propose our bandit algorithm and prove its general regret guarantee for non-negative rewards. Then in Section 4.2 we show how to get a multi-scale style regret guarantee for the best arm c_{i^*}, and a weaker guarantee for all arms $\{c_i\}_{iA}$.

4.1 Bandit Multi-Scale Multiplicative Weight (Bandit-MSMW) algorithm

We present our Bandit algorithm (Algorithm 2) when the set of actions A is finite (with $|A| = k$). Let η be the learning rate and γ be the exploration probability. We show the following regret bound.

Algorithm 2 Bandit-MSMW

```
input  exploration parameter $\gamma > 0$, learning rate $\eta > 0$.
initialize  $\mathbf{p}(1) = (1-\gamma)\mathbf{1}_{i_{\min}} + \frac{\gamma}{k}\mathbf{1}$, where $i_{\min}$ is the arm with minimum range $c_{i_{\min}}$.
for $t = 1, \ldots, T$ do
    Let $\tilde{\mathbf{p}}(t) = (1-\gamma)\mathbf{p}(t) + \frac{\gamma}{k}\mathbf{1}$.
    Randomly pick an expert $i_t$ drawn from $\tilde{\mathbf{p}}(t)$, and observe $g_{i_t}(t)$.
    Let $\tilde{\mathbf{g}}(t)$ be such that
        $$\tilde{g}_i(t) = \begin{cases} \frac{g_i(t)}{\tilde{p}_i(t)} & \text{if } i = i_t; \\ 0 & \text{otherwise.} \end{cases}$$
    $\forall i \in A: \quad w_i(t+1) \leftarrow p_i(t) \cdot \exp(\frac{\eta}{c_i} \cdot \tilde{g}_i(t))$.
    Find $\lambda^*$ (e.g., binary search) s.t. $\sum_{i \in A} w_i(t+1) \cdot \exp(-\frac{\lambda^*}{c_i}) = 1$.
    $\forall i \in A: \quad p_i(t+1) \leftarrow w_i(t+1) \cdot \exp(-\frac{\lambda^*}{c_i})$.
end for
```

Lemma 4.1. *For any exploration probability $0 < \gamma \leq \frac{1}{2}$ and any learning rate parameter $0 < \eta \leq \frac{\gamma}{k}$, the Bandit-MSMW algorithm achieves the following regret bound when the gains are non-negative :*

$$\forall i \in A : \mathbb{E}\left[\text{REGRET}_i\right] \leq O\left(\frac{1}{\eta}\log\left(\frac{k}{\gamma}\right) \cdot c_i + \eta \sum_{j \in A} G_j + \gamma \cdot G_i\right)$$

Proof We further define:

$$\begin{aligned} \widetilde{G}_{\text{ALG}} &\triangleq \sum_{t \in [T]} g_{i_t}(t) = \sum_{t \in [T]} \tilde{\mathbf{p}}(t) \cdot \tilde{\mathbf{g}}(t) \ , \\ \widetilde{G}_j &\triangleq \sum_{t \in [T]} \tilde{g}_j(t) \ . \end{aligned}$$

In expectation over the randomness of the algorithm, we have:

1. $\mathbb{E}\left[G_{\text{ALG}}\right] = \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right]$; and

2. $G_j = \mathbb{E}\left[\widetilde{G}_j\right]$ for any $j \in A$.

Hence, to upper bound $\mathbb{E}\left[\text{REGRET}_i\right] = G_i - \mathbb{E}\left[G_{\text{ALG}}\right]$, it suffices to upper bound $\mathbb{E}\left[\widetilde{G}_i - \widetilde{G}_{\text{ALG}}\right]$. By the definition of the probability that the algorithm picks each arm, i.e., $\tilde{\mathbf{p}}(t)$, we have:

$$\mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \geq (1-\gamma) \textstyle\sum_{t \in [T]} \mathbf{p}(t) \cdot \tilde{\mathbf{g}}(t) \ .$$

Hence, we have that for any initial distribution $\mathbf{q}$ over A:

$$\begin{aligned}\sum_{j \in A} q_j \cdot \mathbb{E}\left[\widetilde{G}_j\right] - \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] &\leq \ \mathbb{E}\left[\textstyle\sum_{j \in A} q_j \cdot \widetilde{G}_j - \sum_{t \in [T]} \mathbf{p}(t) \cdot \tilde{\mathbf{g}}(t)\right] + \tfrac{\gamma}{1-\gamma} \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \\ &\leq \ \mathbb{E}\left[\textstyle\sum_{j \in A} q_j \cdot \widetilde{G}_j - \sum_{t \in [T]} \mathbf{p}(t) \cdot \tilde{\mathbf{g}}(t)\right] + 2\gamma \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \ . \end{aligned} \tag{19}$$

Next, we upper bound the 1st term on the RHS. Note that $\mathbf{p}(t)$'s are the probability of choosing experts by MSMW when the experts have rewards $\tilde{\mathbf{g}}(t)$'s. By Lemma 3.4, we have that for any benchmark distribution $\mathbf{q}$ over S, the Bandit-MSMW algorithm satisfies that:

$$\sum_{j \in A} q_j \cdot \widetilde{G}_j - \sum_{t \in [T]} \mathbf{p}(t) \cdot \tilde{\mathbf{g}}(t) \leq \eta \sum_{t \in [T]} \sum_{j \in A} \frac{p_j(t)}{c_j} \cdot \left(\tilde{g}_j(t)\right)^2 + \frac{1}{\eta} \sum_{j \in A} c_j \left(q_j \ln\left(\frac{q_j}{p_j(1)}\right) - q_j + p_j(1)\right) \ . \tag{20}$$

For any $t \in [T]$ and any $j \in A$, by the definition of $\tilde{g}_j(t)$, it equals $\frac{g_j(t)}{\tilde{p}_j(t)}$ with probability $\tilde{p}_j(t)$, and equals 0 otherwise. Thus, if we fix the random coin flips in the first $t-1$ rounds and, thus, fix $\tilde{\mathbf{p}}(t)$, and take expectation over the randomness in round t, we have that:

$$\mathbb{E}\left[\frac{p_j(t)}{c_j} \cdot \left(\tilde{g}_j(t)\right)^2\right] = \frac{p_j(t)}{c_j} \cdot \tilde{p}_j(t) \cdot \left(\frac{g_j(t)}{\tilde{p}_j(t)}\right)^2 = \frac{p_j(t)}{\tilde{p}_j(t)} \frac{(g_j(t))^2}{c_j} \ .$$

Further note that $\tilde{p}_j(t) \geq (1-\gamma) p_j(t)$, and $g_j(t) \leq c_j$, the above is upper bounded by $\frac{1}{1-\gamma} g_j(t) \leq 2 g_j(t)$. Putting together with (20), we have that for any $0 < \eta \leq \frac{\gamma}{n}$:

$$\begin{aligned}\mathbb{E}\left[\sum_{j \in A} q_j \cdot \widetilde{G}_j - \sum_{t \in [T]} \mathbf{p}(t) \cdot \tilde{\mathbf{g}}(t)\right] &\leq \ \eta \sum_{t \in [T]} \sum_{j \in A} 2 g_j(t) + \frac{1}{\eta} \sum_{j \in A} c_j \left(q_j \ln\left(\frac{q_j}{p_j(1)}\right) - q_j + p_j(1)\right) \\ &= \ 2\eta \sum_{j \in A} G_j + \frac{1}{\eta} \sum_{j \in A} c_j \left(q_j \ln\left(\frac{q_j}{p_j(1)}\right) - q_j + p_j(1)\right)\end{aligned}$$

Combining with (19), we have:

$$\sum_{j \in A} q_j \cdot \mathbb{E}\left[\widetilde{G}_j\right] - \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \leq 2\eta \sum_{j \in A} G_j + \frac{1}{\eta} \sum_{j \in A} c_j \left(q_j \ln\left(\frac{q_j}{p_j(1)}\right) - q_j + p_j(1)\right) + 2\gamma \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right]$$

Let $\mathbf{q} = (1-\gamma)\mathbf{1}_i + \frac{\gamma}{k}\mathbf{1}$. Recall that $\mathbf{p}(1) = (1-\gamma)\mathbf{1}_{i_{\min}} + \frac{\gamma}{k}\mathbf{1}$ (recall $i_{\min}$ is the arm with minimum range $c_{i_{\min}}$). Similar to the discussion for the expert problem in Section 3.3, the 2nd term on the RHS is upper bounded by $O\left(\frac{1}{\eta} \log\left(\frac{k}{\gamma}\right) \cdot c_i\right)$. Hence, we have:

$$\sum_{j \in A} q_j \cdot \mathbb{E}\left[\widetilde{G}_j\right] - \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \leq 2\eta \sum_{j \in A} G_j + O\left(\frac{1}{\eta} \log\left(\frac{k}{\gamma}\right) \cdot c_i\right) + 2\gamma \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \ . \tag{21}$$

Further, the LHS is lower bounded as:

$$(1-\gamma)\mathbb{E}\left[\widetilde{G}_i\right] + \frac{\gamma}{k} \sum_{j \in A} \mathbb{E}\left[\widetilde{G}_j\right] - \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \geq (1-\gamma)\mathbb{E}\left[\widetilde{G}_i\right] - \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \ .$$

The lemma then follows by putting it back to (21) and rearranging terms.

4.2 Regret bounds for non-negative rewards - proof of Theorem 2.3

Proof of Theorem 2.3 Let $\gamma = \epsilon$ and $\eta = \frac{\gamma}{k} = \frac{\epsilon}{k}$ in Lemma 4.1, we get that the expected regret w.r.t. an action $i \in A$ is bounded by:

$$O\left(\epsilon \cdot G_i + \tfrac{\epsilon}{k} \textstyle\sum_{j \in A} G_j + c_i \cdot \tfrac{k}{\epsilon} \ln\left(\tfrac{k}{\epsilon}\right)\right) \ .$$

When $i = i^*$ (best arm), regret is bounded by $O\left(\epsilon \cdot G_{i^*} + c_i^* \cdot \frac{k}{\epsilon} \ln\left(\frac{k}{\epsilon}\right)\right)$, as desired.

For the regret w.r.t. an arbitrary action, note that $\mathbb{E}\left[G_{\text{ALG}}\right] \geq \frac{\gamma}{k} \sum_{j \in A} G_j$. Thus, the regret bound w.r.t. an action $i \in A$ in Lemma 4.1 is further upper bounded by:

$$O\left(\tfrac{1}{\eta} \log\left(\tfrac{k}{\gamma}\right) \cdot c_i + \left(\tfrac{\eta k}{\gamma} + \gamma\right) \cdot \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right]\right)$$

The theorem then follows by letting $\gamma = \epsilon$ and $\eta = \frac{\gamma}{k} = \frac{\epsilon^2}{k}$.

5 Auctions and Pricing

5.1 Auctions and pricing as multi-scale online learning problems

Online single buyer auction and posted pricing Recall that in each round, the algorithm chooses an action, i.e., a price, $p_t \in [1, h]$; the adversary picks a value $v(t) \in [1, h]$; and the algorithm collects reward $g_{p_t}(t) = p_t \cdot \mathbf{1}(v(t) \geq p_t)$. In order to obtain a $1 - \epsilon$ approximation of the optimal revenue, it suffices to consider prices of the form $(1+\epsilon)^j$ for $0 \leq j \leq \lfloor \log_{1+\epsilon} h \rfloor = O(\frac{\log h}{\epsilon})$. As a result, we reduce the online single buyer auction problem and the online posted pricing problem to a multi-sclae online learning problem with full information and bandit feedback respectively with $k = O(\frac{\log h}{\epsilon})$ actions whose ranges form a geometric sequence $(1+\epsilon)^j$, $0 \leq j < k$.

Online multi buyer auction In multi buyer auctions, we consider the set of all discretized Myerson-type auctions as the action space. We start by defining Myerson-type auctions:

Definition 5.1 (Myerson-type auctions)**.** A *Myerson-type auction* is defined by n non-decreasing virtual value mappings $\phi_1, \ldots, \phi_n : [1, h] \mapsto [-\infty, h]$. Given a value profile $v_1, \ldots, v_n$, the item is given to the bidder j with the largest virtual value $\phi_j(v_j)$. Then, bidder j pays the minimum value that would keep him as the the winner.

Myerson (1981) shows that when the bidders' values are drawn from independent (but not necessarily identical) distributions, the revenue-optimal auction is a Myerson-type auction. Devanur et al. (2016, Lemma 5) observe that to obtain a $1 - \epsilon$ approximation, it suffices to consider the set of discretized Myerson-type auctions that treat each bidder's value as if it is equal to the closest power of $1 + \epsilon$ from below. As a result, it suffices to consider the set of discretized Myerson-type auctions, each of which is defined by the virtual values of $(1+\epsilon)^j$'s, i.e., by $O(n \log h/\epsilon)$ real numbers $\phi_\ell((1+\epsilon)^j)$, for $\ell \in [n]$, and $0 \leq j \leq \lfloor \log_{1+\epsilon} h \rfloor$. Devanur et al. (2016); Gonczarowski and Nisan (2017) further note that a discretized Myerson-type auction is in fact completely characterized by the total ordering of $\phi_\ell((1+\epsilon)^j)$'s; their actual values do not matter. Indeed, both the allocation rule and the payment rule are determined by the ordering of virtual values. As a result, our action space is a finite set with at most $O((n \log h/\epsilon)!)$ actions. The range of an action, i.e., a discretized Myerson-type auction, is the largest price ever charged by the auction, i.e., the largest value v of the form $(1+\epsilon)^j$ such that there exists $\ell \in [n]$, $\phi_\ell(v) > \phi_\ell((1+\epsilon)^{-1} v)$.

5.2 Proof of Theorem 2.5

Proof *Online single buyer auction.* Recall the above formulation of the problem as an online learning problem with full information. The case when h is known then follows by Theorem 2.1, letting $\boldsymbol{\pi}$ be the uniform distribution over the $k = O(\log h/\epsilon)$ actions, i.e., discretized prices.

When h is not known upfront, we consider a countably infinite action space comprised of all prices of the form $(1+\epsilon)^j$, for $j \geq 0$. Then, let the prior distribution $\boldsymbol{\pi}$ be such that for any price $p = (1+\epsilon)^j$, $\pi_p = \epsilon(1+\epsilon)^{-j-1} = \frac{\epsilon}{1+\epsilon} \cdot \frac{1}{p}$. The approximation guarantee then follows by Theorem 2.1.

Online posted pricing. Recall the above formulation of the problem as an online learning problem with bandit feedback. This part then follows by Theorem 2.3 with $k = O(\log h/\epsilon)$ actions.

Online multi buyer auction. Recall the above formulation of the problem as an online learning problem with full information. The case when h is known then follows by Theorem 2.1, where we let $\boldsymbol{\pi}$ be the uniform distribution over the $k = O((n \log h/\epsilon)!)$ actions, i.e., Myerson-type auctions.

When h is not known upfront, we consider a countably infinite action space A as follows. For any $p = (1+\epsilon)^j$, $j \geq 0$, let the $k_p = O((n \log p/\epsilon)!)$ Myerson-type auctions for values in $[1, p]$ be in A; we assume these auctions treat any values greater than p as if they were p. Further, we choose the prior distribution $\boldsymbol{\pi}$ such that the probability mass of each auction for range $[1, p]$ is equal to $\frac{\epsilon}{1+\epsilon} \cdot \frac{1}{p} \cdot \frac{1}{k_p}$. The approximation guarantee then follows by Theorem 2.1.

5.3 Proof of Theorem 2.6

Proof *Online single buyer auction.* When h is known, by Theorem 2.1, letting $\boldsymbol{\pi}$ be the uniform distribution over the $k = O(\log h/\epsilon)$ actions, i.e., discretized prices, we have that for any price p (recall that $c_p = p$):

$$G_{\text{ALG}} \geq (1-\epsilon) \cdot G_p - O\left(\tfrac{\log(\log h/\epsilon)}{\epsilon} \cdot p\right) \ .$$

For the δ-guarded optimal price p^* (i.e., subject to selling in at least δT rounds), we have $G_{p^*} \geq \delta T \cdot p^*$. Therefore, when $T \geq O\left(\log(\log h/\epsilon)/\epsilon^2 \delta\right)$, the additive term of the above approximation guarantee is at most $\epsilon \cdot G_{p^*}$. So the theorem holds.

The treatment for the case when h is not known upfront is essentially the same as in Theorem 2.5. We consider a countably infinite action space comprised of all prices of the form $(1+\epsilon)^j$, for $j \geq 0$. Then, let the prior distribution $\boldsymbol{\pi}$ be such that for any price $p = (1+\epsilon)^j$, $\pi_p = \epsilon(1+\epsilon)^{-j-1} = \frac{\epsilon}{1+\epsilon} \cdot \frac{1}{p}$.

Online posted pricing. Recall the above formulation of the problem as an online learning problem with bandit feedback. By Theorem 2.3 with $k = O(\log h/\epsilon)$ actions, we have that for any price p:

$$G_{\text{ALG}} \geq (1-\epsilon) \cdot G_p - O\left(\tfrac{\log h \log(\log h/\epsilon)}{\epsilon^3} \cdot p\right) \ .$$

Again, for the δ-guarded optimal price p^* (i.e., subject to selling in at least δT rounds), we have $G_{p^*} \geq \delta T \cdot p^*$. Therefore, when $T \geq O\left(\log h \log\left(\log h/\epsilon\right)/\epsilon^4 \delta\right)$, the additive term of the above approximation guarantee is at most $\epsilon \cdot G_{p^*}$. So the theorem holds.

Online multi buyer auction. Suppose i^* is the δ-guarded best Myerson-type auction. Recall that $\bar{V}$ is the largest value such that there are at least δT distinct $v(t)$'s with $\max_{\ell \in [n]} v_\ell(t) \geq \bar{V}$. So we may assume without loss of generality that i^* does not distinguish values greater than $\bar{V}$. Hence:

$$c_{i^*} \leq \bar{V} \ . \tag{22}$$

Further, note that running a 2nd-price auction with anonymous reserve $\bar{V}$ is a Myerson-type auction (e.g., mapping values less than $\bar{V}$ to virtual value $-\infty$ and values greater than or equal to $\bar{V}$ to virtual value $\bar{V}$), and it gets revenue at least $\delta T \cdot \bar{V}$. So we have that:

$$G_{p^*} \geq \delta T \cdot \bar{V} \ . \tag{23}$$

Finally, the above implies that to obtain a $1-\epsilon$ approximation, it suffices to consider prices that are at least $\epsilon\delta\bar{V}$. Hence, it suffices to consider Myerson-type auctions that, for a given $\bar{V}$, do not distinguish among values greater than $\bar{V}$, and do not distinguish among values smaller than $\epsilon\delta\bar{V}$. There are $O(\log h/\epsilon)$ different values of $\bar{V}$. Further, given $\bar{V}$, there are only $O(\log(1/\epsilon\delta)/\epsilon)$ distinct values to be considered and, thus, there are at most $O((n\log(1/\epsilon\delta)/\epsilon)!)$ distinct Myerson-type auctions of this kind. Hence, the total number of distinct Myerson-type actions that we need to consider is at most:

$$k = O\left(\frac{\log h}{\epsilon} \cdot \left(\frac{n\log(1/\epsilon\delta)}{\epsilon}\right)!\right) \ .$$

When h is known, letting $\boldsymbol{\pi}$ be the uniform distribution over the k actions in Theorem 2.1, we have that (recall Eqn. (22)):

$$G_{\text{ALG}} \geq (1-\epsilon)\cdot G_{i^*} - O\left(\frac{n\log(1/\epsilon\delta)\log(n\log(1/\epsilon\delta)/\epsilon)}{\epsilon^2} + \frac{\log(\log h/\epsilon)}{\epsilon}\right)\cdot \bar{V} \ .$$

When $T \geq O\left(\frac{n\log(1/\epsilon\delta)\log(n\log(1/\epsilon\delta)/\epsilon)}{\epsilon^3\delta} + \frac{\log(\log h/\epsilon)}{\epsilon^2\delta}\right)$, the additive term of the above approximation guarantee is at most $\epsilon \cdot G_{i^*}$ due to Eqn. (23). So the theorem holds.

Again, the treatment for the case when h is not known upfront is similar to that in Theorem 2.5. When h is not known upfront, we consider a countably infinite action space A as follows. For any $\bar{V} = (1+\epsilon)^j$, $j \geq 0$, let the $k' = O((n\log(1/\epsilon\delta)/\epsilon)!)$ Myerson-type auctions that do not distinguish among values greater than $\bar{V}$, and do not distinguish among values smaller than $\epsilon\delta\bar{V}$ be in A. Further, we choose the prior distribution $\boldsymbol{\pi}$ such that the probability mass of each Myerson-type auction for a given $\bar{V}$ is equal to $\frac{\epsilon}{1+\epsilon}\cdot\frac{1}{\bar{V}}\cdot\frac{1}{k'}$. The approximation guarantee then follows by Theorem 2.1 and essentially the same argument as the known h case.

Remark Devanur et al. (2016) show that when the values are drawn from independent regular distributions, the ϵ-guarded optimal is a $1-\epsilon$ approximation of the unguarded optimal. So our convergence rate for the online multi buyer auction problem in Theorem 2.1 implies a $\tilde{O}(n\epsilon^{-4})$ sample complexity modulo a mild $\log\log h$ dependency on the range, almost matching the best known sample complexity upper bound for regular distributions.

6 Multi-scale Online Learning with Symmetric Range

In this section, we consider multi-scale online learning when the rewards are in a symmetric range, i.e. for all $i \in A$ and $t \in [T]$, $g_i(t) \in [-c_i, c_i]$. We look at both full information and bandit setting, and prove action-specific regret upper bounds. We defer the regret lower bound proofs to the appendix, Sections A.2 and A.3.

6.1 Multi-scale expert problem with symmetric range

Recall the proof of Lemma 3.4. The proof only requires $g_i(t) \in [-c_i, c_i]$ for all $i \in A, t \in [T]$. Choosing q to be $\mathbf{1}_i$, a vector with a 1-entry in i^{th} coordinate and 0-entries elsewhere for an action $i \in A$, and noting that

$$\sum_{t\in[T]}\sum_{i\in A} p_i(t)\frac{(g_i(t))^2}{c_i} \le \sum_{t\in[T]}\sum_{i\in A} p_i(t)\cdot \left|g_i(t)\right| \ ,$$

we get the following regret bound as a corollary of Lemma 3.4.

Corollary 6.1. *For any initial distribution μ over A, and any learning rate parameter $0<\eta\le 1$, the MSMW algorithm achieves the following regret bound:*

$$\forall i\in A: \quad \mathbb{E}\left[\text{REGRET}_i\right] \le \eta\cdot\mathbb{E}\left[\sum_{t\in[T]}\left|g_i(t)\right|\right] + \frac{1}{\eta}c_i\cdot\log\left(\frac{1}{\mu_i}\right) + \frac{1}{\eta}\sum_{j\in A}\mu_j c_j \tag{24}$$

Now, we can prove the multi-scale regret upper bound in Theorem 2.7 using Corollary 6.1.

Proof of Theorem 2.7 The proof follows by choosing an appropriate initial distribution μ in Corollary 6.1. By Corollary 6.1, we have:

$$\mathbb{E}\left[\text{REGRET}_i\right] \le \eta\cdot\mathbb{E}\left[\sum_{t\in[T]}\left|g_i(t)\right|\right] + \frac{1}{\eta}c_i\cdot\log(\frac{1}{\mu_i}) + \frac{1}{\eta}\sum_{j\in A}\mu_j c_j$$

Let $i_{\min}$ be an action with the minimum range $c_{i_{\min}} = c_{\min}$. Consider an initial distribution $\mu_j = \pi_j\frac{c_{\min}}{c_j}$ for all $j\neq i_{\min}$, and $\mu_{i_{\min}} = 1-\sum_{j\neq i_{\min}}\mu_j$, i.e., putting all remaining probability mass on action $i_{\min}$. Then, the 3rd term on the RHS is upper bounded by:

$$\sum_{j\in A}\mu_j c_j = \sum_{j\neq i_{\min}}\mu_j c_j + \mu_{i_{\min}}c_{i_{\min}} = \sum_{j\neq i_{\min}}\pi_j c_{\min} + \mu_{i_{\min}}c_{\min} \le 2c_{\min} \le 2c_i \ .$$

For $i\neq i_{\min}$, by the definition of μ_i, we have:

$$\begin{aligned}\mathbb{E}\left[\text{REGRET}_i\right] &\le \eta\cdot\mathbb{E}\left[\sum_{t\in[T]}\left|g_i(t)\right|\right] + \frac{1}{\eta}c_i\cdot\log(\frac{1}{\pi_i}\cdot\frac{c_i}{c_{\min}}) + \frac{1}{\eta}\cdot 2c_{\min}\\ &= \eta\cdot\mathbb{E}\left[\sum_{t\in[T]}\left|g_i(t)\right|\right] + O\left(\frac{1}{\eta}\log\left(\frac{1}{\pi_i}\cdot\frac{c_i}{c_{\min}}\right)\cdot c_i\right) \ .\end{aligned}$$

So the theorem follows by choosing $\eta=\epsilon$. For $i=i_{\min}$, note that $\mu_j\le\pi_j$ for all $j\neq i_{\min}$ and, thus, $\mu_{i_{\min}} = 1-\sum_{j\neq i_{\min}}\mu_j \ge 1-\sum_{j\neq i_{\min}}\pi_j = \pi_{i_{\min}} = \pi_{i_{\min}}\frac{c_{\min}}{c_{i_{\min}}}$. The theorem then holds following the same calculation as in the $j\neq i_{\min}$ case.

6.2 Multi-scale bandit problem with symmetric range

We start by showing the following regret bound, whose proof is an alteration of that for Lemma 4.1 under symmetric range (and is deferred to the appendix, Section A.6). Next, we prove Theorem 2.10.

Lemma 6.2. *For any exploration rate $0<\gamma\le\min\{\frac{1}{2},\frac{c_{\min}}{c_{\max}}\}$ and any learning rate $0<\eta\le\frac{\gamma}{k}$, the Bandit-MSMW algorithm (Algorithm 2) achieves the following regret bound:*

$$\forall i\in A: \mathbb{E}\left[\text{REGRET}_i\right] \le O\left(\frac{1}{\eta}\log\left(\frac{k}{\gamma}\right)\cdot c_i + \gamma T\cdot c_{\max}\right)$$

Proof of Theorem 2.10 Let $\gamma = \epsilon\frac{c_{\min}}{c_{\max}}$ and $\eta=\frac{\gamma}{k}$ in Lemma 6.2. Theorem follows noting that $\gamma c_{\max} = \epsilon c_{\min} \le \epsilon c_i$.

References

Shipra Agrawal and Nikhil R Devanur. 2014. Bandits with concave rewards and convex knapsacks. In *Proceedings of the fifteenth ACM conference on Economics and computation*. ACM, 989–1006.

Kareem Amin, Afshin Rostamizadeh, and Umar Syed. 2013. Learning prices for repeated auctions with strategic buyers. In *Advances in Neural Information Processing Systems*. 1169–1177.

Sanjeev Arora, Elad Hazan, and Satyen Kale. 2012. The Multiplicative Weights Update Method: a Meta-Algorithm and Applications. *Theory of Computing* 8, 1 (2012), 121–164.

Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. 1995. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on*. IEEE, 322–331.

Moshe Babaioff, Shaddin Dughmi, Robert Kleinberg, and Aleksandrs Slivkins. 2015. Dynamic pricing with limited supply. *ACM Transactions on Economics and Computation* 3, 1 (2015), 4.

Ashwinkumar Badanidiyuru, Robert Kleinberg, and Aleksandrs Slivkins. 2013. Bandits with knapsacks. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*. IEEE, 207–216.

Ziv Bar-Yossef, Kirsten Hildrum, and Felix Wu. 2002. Incentive-compatible online auctions for digital goods. In *Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 964–970.

Omar Besbes and Assaf Zeevi. 2009. Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms. *Operations Research* 57, 6 (2009), 1407–1420.

Avrim Blum and Jason D Hartline. 2005. Near-optimal online auctions. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 1156–1163.

Avrim Blum, Vijay Kumar, Atri Rudra, and Felix Wu. 2004. Online learning in online auctions. *Theoretical Computer Science* 324, 2-3 (2004), 137–146.

Sébastien Bubeck. 2011. Introduction to online optimization. *Lecture Notes* (2011), 1–86.

Richard Cole and Tim Roughgarden. 2014. The sample complexity of revenue maximization. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*. 243–252.

Arnoud V den Boer. 2015. Dynamic pricing and learning: historical origins, current research, and new directions. *Surveys in operations research and management science* 20, 1 (2015), 1–18.

Nikhil R Devanur, Zhiyi Huang, and Christos-Alexandros Psomas. 2016. The sample complexity of auctions with side information. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing*. ACM, 426–439.

Nikhil R Devanur, Yuval Peres, and Balasubramanian Sivan. 2014. Perfect bayesian equilibria in repeated sales. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 983–1002.

Peerapong Dhangwatnotai, Tim Roughgarden, and Qiqi Yan. 2014. Revenue maximization with a single sample. *Games and Economic Behavior* (2014).

Dylan Foster, Satyen Kale, Mehryar Mohri, and Karthik Sridharan. 2017. Personal communication. (2017).

Yoav Freund and Robert E Schapire. 1995. A desicion-theoretic generalization of on-line learning and an application to boosting. In *European conference on computational learning theory*. Springer, 23–37.

Yannai A Gonczarowski and Noam Nisan. 2017. Efficient Empirical Revenue Maximization in Single-Parameter Auction Environments. In *Proceedings of the ACM STOC*.

Zhiyi Huang, Yishay Mansour, and Tim Roughgarden. 2015. Making the most of your samples. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*. ACM, 45–60.

Robert Kleinberg and Tom Leighton. 2003. The value of knowing a demand curve: Bounds on regret for online posted-price auctions. In *Foundations of Computer Science, 2003. Proceedings. 44th Annual IEEE Symposium on*. IEEE, 594–605.

Jamie H Morgenstern and Tim Roughgarden. 2015. On the pseudo-dimension of nearly optimal auctions. In *Advances in Neural Information Processing Systems*. 136–144.

Roger B. Myerson. 1981. Optimal Auction Design. *Mathematics of Operations Research* 6, 1 (1981), 58–73.

Tim Roughgarden and Okke Schrijvers. 2016. Ironing in the dark. In *Proceedings of the 2016 ACM Conference on Economics and Computation*. ACM, 1–18.

Ilya Segal. 2003. Optimal pricing mechanisms with unknown demand. *The American economic review* 93, 3 (2003), 509–529.

Vasilis Syrgkanis. 2017. A Sample Complexity Measure with Applications to Learning Optimal Auctions. *arXiv preprint arXiv:1704.02598* (2017).

Kalyan T Talluri and Garrett J Van Ryzin. 2006. *The theory and practice of revenue management*. Vol. 68. Springer Science & Business Media.

A Other Deferred Proofs and Discussions

A.1 Discussion on choice of $\boldsymbol{\pi}$ for bandit symmetric range

We now describe how the choice of initial distribution $\boldsymbol{\pi}$ affects the bound given in Theorem 2.7.

- When the action set is finite, we can choose $\boldsymbol{\pi}$ to be the uniform distribution to get the term

  $$O\big(\frac{1}{\epsilon}\log(kc_i/c_{\min})\cdot c_i\big)$$

  This recovers the standard bound by setting $c_i = c_{\max}$ for all $i \in A$.

- We can choose $\pi_i = \frac{c_i}{\sum_{j\in A} c_j}$ to get $O\big(\frac{1}{\epsilon}\log(\sum_{j\in A} c_j/c_{\min})\cdot c_i\big)$. In particular, if the c_i's form an arithmetic progression with a constant difference then this is just $O\big(\frac{\log k}{\epsilon}\cdot c_i\big)$.

- If there are infinitely many experts but $\sum_{i\in A} c_i^{-1}$ is convergent, e.g., $c_i = c_{\min}\cdot(1+\epsilon)^{i-1}$, then we can choose $\pi_i = \frac{c_i^{-1}}{\sum_{j\in A} c_j^{-1}}$ for all $i \in A$. This gives $O\left(\frac{1}{\eta}\log\left(\sum_j \frac{c_{\min}}{c_j}\cdot\frac{c_i^2}{c_{\min}^2}\right)\cdot c_i\right)$.

A.2 Log factor dependence for symmetric range - proof of Theorem 2.9

Proof of Theorem 2.9 We first show that for any online learning algorithm, and any sufficiently large $h > 1$, there is an instance that has two experts with $c_1 = 1$ and $c_2 = h$ with $T = \Theta(\log h)$ rounds, such that either

$$\mathbb{E}\left[\text{REGRET}_1\right] > \tfrac{1}{2}T + \sqrt{h}\ , \qquad \text{or} \qquad \mathbb{E}\left[\text{REGRET}_2\right] > \tfrac{1}{2}Th + \tfrac{1}{5}h\log_2 h\ .$$

We will construct this instance with $T = \frac{1}{2}\log_2 h - 1$ rounds adaptively that always has gain 0 for action 1 and gain either h or $-h$ for action 2. The proof of the theorem then follows as $c_{\min} = 1$, $c_{\max} = h$, $T = \frac{1}{2}\log_2 h - 1$, and $k = 2$ in this instance. Let q_t denote the probability that the algorithm picks action 2 in round t after having the same rewards 1 and h for the two actions respectively in the first $t-1$ rounds. We will first show that (1) if the algorithm has small regret with respect to action 1, then q_t must be upper bounded since the adversary may let action 2 have cost $-h$ in any round t in which q_t is too large. Then, we will show that (2) since q_t is upper bounded for any $1 \le t \le T$, the algorithm must have large regret with respect to action 2.

We proceed with the upper bounding q_t's. Concretely, we will show the following lemma.

Lemma A.1. *Suppose $\mathbb{E}\left[\text{REGRET}_1\right] \le \frac{1}{2}T + \sqrt{h}$. Then, for any $1 \le t \le T$, we have $q_t \le \frac{2^t}{\sqrt{h}}$.*

Proof of Lemma A.1 We will prove by induction on t. Consider the base case $t = 1$. Suppose for contradiction that $q_1 > \frac{2}{\sqrt{h}}$. Then, consider an instance in which action 2 always has gain. In this case, the expected gain of the algorithm (even if it always correctly picks action 1 in the remaining instance) is at most $q_1\cdot(-h) < -2\sqrt{h}$. This is a contradiction to the assumption that $\mathbb{E}\left[\text{REGRET}_1\right] \le \frac{1}{2}T + \sqrt{h} < 2\sqrt{h}$.

Next, suppose the lemma holds for all rounds prior to round t. Then, the expected gain of algorithm in the first $t-1$ rounds if arm 2 has gain H is

$$\sum_{\ell=1}^{t-1} q_\ell \cdot h \le \sum_{\ell=1}^{t-1} 2^\ell\sqrt{h} = \big(2^t - 2\big)\sqrt{h}\ .$$

Suppose for contradiction that $q_t > \frac{2^t}{\sqrt{h}}$. Then, consider an instance in which action 2 has gain H in the first $t-1$ rounds and $-H$ afterwards. In this case, the expected gain of the algorithm (even if it always correctly picks action 1 after round t) is at most

$$\big(2^t - 2\big)\sqrt{h} + q_t(-h) < \big(2^t - 2\big)\sqrt{h} + 2^t\sqrt{h} < -2\sqrt{h} \ .$$

This is a contradiction to the assumption that $\mathbb{E}\left[\text{REGRET}_1\right] \le \frac{1}{2}T + \sqrt{h} < 2\sqrt{h}$.

Consider an instance in which action 2 always has gain H. Suppose that $\mathbb{E}\left[\text{REGRET}_1\right] \le \frac{1}{2}T + \sqrt{h}$. As an immediate implication of the above lemma, the algorithm is that the expected gain of the algorithm is upper bounded by:

$$\sum_{t=1}^{T} q_t h \le \sum_{t=1}^{T} 2^t\sqrt{h} < 2^{T+1}\sqrt{h} = h \ .$$

Note that in this instance $\mathbb{E}\left[G_2\right] = T \cdot h$. Thus, the regret w.r.t. action 2 is at least $(T-1)h$, which is greater than $\frac{1}{2} \cdot \mathbb{E}\left[G_2\right] + \frac{1}{5}h\log_2 h$ for sufficiently large h.

A.3 Regret lower-bound for symmetric range -proof of Theorem 2.12

Proof of Theorem 2.12 We first show that for any online multi-scale bandits algorithm problem, and there is an instance that has two arms with $c_1 = 1$ and $c_2 = h$ for some sufficiently large h, a sufficiently large T, and $\epsilon = \sqrt{\frac{h}{256T}}$, such that either

$$\mathbb{E}\left[\text{REGRET}_1\right] > \epsilon T + \tfrac{1}{256\epsilon}h \ , \qquad \text{or} \qquad \mathbb{E}\left[\text{REGRET}_2\right] > \epsilon T h + \tfrac{1}{256\epsilon}h^2$$

We will prove the existence of this instance by looking at the stochastic setting, i.e., the gain vectors $\mathbf{g}(t)$'s are i.i.d. for $1 \le t \le T$. We consider two instances, both of which admit a fixed gain of 0 for action 1. In the first instance, the gain of action 2 is h with probability $\frac{1}{2} - 2\epsilon$, and $-h$ otherwise. Hence, the expected gain of playing action 2 is $-4\epsilon h$ per round in instance 1. In the second instance, the gain of action 2 is h with probability $\frac{1}{2} + 2\epsilon$, and $-h$ otherwise. Hence, the expected gain of playing action two is $4\epsilon h$ per round in instance 2. Note this proves the theorem, as $c_{\min} = 1$, $c_{\max} = h$, $k = 2$ and and $T = \frac{h}{256\epsilon^2}$.

Suppose for contradiction that the algorithm satisfies:

$$\mathbb{E}\left[\text{REGRET}_1\right] \le \epsilon T + \tfrac{1}{256\epsilon}h = \tfrac{1}{128\epsilon}h \quad , \quad \mathbb{E}\left[\text{REGRET}_2\right] \le \epsilon h T + \tfrac{1}{256\epsilon}h^2 = \tfrac{1}{128\epsilon}h^2 \ .$$

Let N_1 denote the expected number of times that the algorithm plays action 2 in instance 1. Then, the expected regret with respect to action 1 in instance 1 is $N_1 \cdot 4\epsilon h$. By the assumption that $\mathbb{E}\left[\text{REGRET}_1\right] \le \frac{1}{128\epsilon}h$, we have $N_1 \le \frac{1}{512\epsilon^2}$.

Next, by standard calculation, we get that the Kullback-Leibler (KL) divergence of the observed rewards in a single round in the two instances is 0 if action 1 is played and is at most $64\epsilon^2$ (for $0 < \epsilon < 0.1$) if action 2 is played. So the KL divergence of the observed reward sequences in the two instances is at most $64\epsilon^2 \cdot N_1 \le \frac{1}{8}$.

Then, we use a standard inequality about KL divergences. For any measurable function $\psi : X \mapsto \{1, 2\}$, we have $\Pr_{X\sim\rho_1}\big(\psi(X) = 2\big) + \Pr_{X\sim\rho_2}\big(\psi(X) = 1\big) \ge \frac{1}{2}\exp\big(-KL(\rho_1, \rho_2)\big)$. For any $1 \le t \le T$, let ρ_1 and ρ_2 be the distribution of observed rewards up to a round t in the two instances, and let $\psi(X)$ be the action played by the algorithm. By this inequality and the above bound on the KL divergence between the observed rewards in the two instances, we get that in each round, the probability that the algorithm plays action 2 in instance 1, plus the probability that the algorithm plays action 1 in instance 2, is at least $\frac{1}{2}\exp\left(-\frac{1}{8}\right) > \frac{2}{5}$ in any round t. Thus, the expected number of times that the algorithm plays action 1 in instance 2 from round 1 to T, denoted as N_2, is at least $N_2 \geq \frac{2}{5}\cdot T - N_1 \geq \frac{1}{3}\cdot T$, where the second inequality holds for sufficiently large h. Therefore, the expected regret w.r.t. action 2 in instance 2 is at least: $4\epsilon h \cdot \frac{1}{3}\cdot T = \frac{4}{3}\epsilon h T > \frac{1}{128\epsilon}h^2$. This is a contradiction to our assumption that $\mathbb{E}\left[\text{REGRET}_2\right] \leq \frac{1}{128\epsilon}h^2$.

A.4 Proof of Lemma 3.4

Proof of Lemma 3.4 We have:

$$\sum_{i\in A} q_i \cdot G_i - \mathbb{E}\left[G_{\text{ALG}}\right] = \sum_{t\in[T]} \mathbf{q}\cdot\mathbf{g}(t) - \sum_{t\in[T]} \mathbf{p}(t)\cdot\mathbf{g}(t) = \sum_{t\in[T]} \mathbf{g}(t)\cdot\big(\mathbf{q}-\mathbf{p}(t)\big) \tag{25}$$

By applying the regret bound of OMD (Lemma 3.1) to upper-bound the RHS, we have

$$\sum_{i\in A} q_i \cdot G_i - \mathbb{E}\left[G_{\text{ALG}}\right] \leq \frac{1}{\eta}\sum_{t\in[T]} D_F(\mathbf{p}(t),\mathbf{w}(t+1)) + \frac{1}{\eta}D_F(\mathbf{q},\mathbf{p}(1)) \tag{26}$$

To bound the first term in regret, a.k.a *local norm*, we have:

$$\begin{aligned} D_F(\mathbf{p}(t),\mathbf{w}(t+1)) &= \sum_{i\in A} c_i \cdot (p_i(t)\ln(\frac{p_i(t)}{w_i(t+1)}) - p_i(t) + w_i(t+1)) \\ &= \sum_{i\in A} c_i \cdot p_i(t)(-\eta\cdot\frac{g_i(t)}{c_i} - 1 + exp(\eta\cdot\frac{g_i(t)}{c_i})) \end{aligned} \tag{27}$$

Note that $\eta\cdot\frac{g_i(t)}{c_i} \in [-1,1]$ because $g_i(t)\in[-c_i,c_i]$ and $0<\eta\leq 1$. By $\exp(x)-x-1\leq x^2$ for $-1\leq x\leq 1$ and that $\eta g_i(t)\in[-c_i,c_i]$, the above is upper bounded by $\eta^2\sum_{i\in A} p_i(t)\frac{(g_i(t))^2}{c_i}$. We can also rewrite the second term in regret. In fact, if we set $\mathbf{p}(1)=\boldsymbol{\mu}$, then

$$\frac{1}{\eta}\cdot D_F(\mathbf{q},\mathbf{p}(1)) = \frac{1}{\eta}\cdot\sum_{i\in A} c_i\Big(q_i\ln\big(\frac{q_i}{\mu_i}\big) - q_i + \mu_i\Big)$$

By summing the upper-bounds $\eta^2\sum_{i\in A} p_i(t)\frac{(g_i(t))^2}{c_i}$ on each term of local norm in (27) for $t\in[T]$ and putting all the pieces together, we get the desired bound.

We also provide an elementary proof of this lemma using first principles.

Proof of Lemma 3.4 from first principles Based on the update rule of Algorithm 1, we have

$g_i(t) = \frac{c_i}{\eta}\log(\frac{w_i(t+1)}{p_i(t)})$ for any $i \in A$. Therefore:

$$\begin{aligned}
\mathbf{g}(t)\cdot\big(\mathbf{q}-\mathbf{p}(t)\big) &= \sum_{i\in A} g_i(t)\big(q_i - p_i(t)\big) \\
&= \sum_{i\in A}\frac{c_i}{\eta}\cdot\log\big(\frac{w_i(t+1)}{p_i(t)}\big)\cdot\big(q_i-p_i(t)\big) \\
&= \frac{1}{\eta}\Bigg(\sum_{i\in S} c_i\cdot q_i\cdot\log\big(\frac{w_k(t+1)}{p_k(t)}\big) + \sum_{i\in A} c_i\cdot p_i(t)\cdot\log\big(\frac{p_i(t)}{w_i(t+1)}\big)\Bigg) \\
&= \frac{1}{\eta}\Bigg(\sum_{i\in S} c_i\cdot q_i\cdot\log\big(\frac{w_k(t+1)}{p_k(t+1)}\big) + \sum_{i\in S} c_i\cdot q_i\cdot\log\big(\frac{p_k(t+1)}{p_k(t)}\big) \\
&\qquad + \sum_{i\in A} c_i\cdot p_i(t)\cdot\log\big(\frac{p_i(t)}{w_i(t+1)}\big)\Bigg) \qquad (28)
\end{aligned}$$

Now, note that due to the normalization step of Algorithm 1, for any $i \in S$ we have:

$$c_i\cdot\log(\frac{w_i(t+1)}{p_i(t+1)}) = \lambda = \sum_{j\in A} c_j\cdot p_j(t+1)\cdot\frac{\lambda}{c_j} = \sum_{j\in A} c_j\cdot p_j(t+1)\cdot\log(\frac{w_j(t+1)}{p_j(t+1)})$$

So the first summation in (28) is equal to:

$$\begin{aligned}
\sum_{i\in S} c_i\cdot q_i\cdot\log\big(\frac{w_k(t+1)}{p_k(t+1)}\big) &= \sum_{i\in S} q_i\cdot\sum_{j\in A} c_j\cdot p_j(t+1)\cdot\log(\frac{w_j(t+1)}{p_j(t+1)}) \\
&= \sum_{j\in A} c_j\cdot p_j(t+1)\cdot\log(\frac{w_j(t+1)}{p_j(t+1)}) \\
&= \sum_{i\in A} c_i\cdot p_i(t+1)\cdot\log(\frac{w_i(t+1)}{p_i(t+1)}) \qquad (29)
\end{aligned}$$

Combining Eqn. (28) and (29), we have:

$$\begin{aligned}
\mathbf{g}(t)\cdot\big(\mathbf{q}-\mathbf{p}(t)\big) = \frac{1}{\eta}\sum_{i\in A} c_i\cdot\Bigg(p_i(t)\cdot\log(\frac{p_i(t)}{w_i(t+1)}) + p_i(t+1)\cdot\log(\frac{w_i(t+1)}{p_i(t+1)})\Bigg) \\
+\frac{1}{\eta}\sum_{i\in S} c_i\cdot q_i\cdot\log\big(\frac{p_i(t+1)}{p_i(t)}\big)
\end{aligned}$$

The 2nd part is a telescopic sum when we sum over t. We will upper bound the 1st part as follows. By $\log(x) \le (x-1)$, we get that:

$$\begin{aligned}
&\sum_{i\in A} c_i\cdot\Bigg(p_i(t)\cdot\log(\frac{p_i(t)}{w_i(t+1)}) + p_i(t+1)\cdot\log(\frac{w_i(t+1)}{p_i(t+1)})\Bigg) \\
&\quad\le \sum_{i\in A} c_i\cdot\Bigg(p_i(t)\cdot\log(\frac{p_i(t)}{w_i(t+1)}) - p_i(t+1) + w_i(t+1)\Bigg) \\
&\quad= \sum_{i\in A} c_i\cdot\big(p_i(t)-p_i(t+1)\big) + \sum_{i\in A} c_i\cdot\Bigg(p_i(t)\cdot\log(\frac{p_i(t)}{w_i(t+1)}) - p_i(t) + w_i(t+1)\Bigg)
\end{aligned}$$

Again, the 1st part is a telescopic sum when we sum over t. We will further work on the 2nd part. By the relation between $w_i(t+1)$ and $p_i(t)$, we get that:

$$\sum_{i\in A} c_i \cdot \left(p_i(t)\cdot \log(\frac{p_i(t)}{w_i(t+1)}) - p_i(t) + w_i(t+1)\right) = \sum_{i\in A} c_i \cdot p_i(t) \left(-\eta \cdot \frac{g_i(t)}{c_i} - 1 + \exp(\eta \cdot \frac{g_i(t)}{c_i})\right)$$

Note that $\eta \cdot \frac{g_i(t)}{c_i} \in [-1,1]$ because $g_i(t) \in [-c_i, c_i]$ and $0 < \eta \leq 1$. By $\exp(x) - x - 1 \leq x^2$ for $-1 \leq x \leq 1$ and that $\eta g_i(t) \in [-c_i, c_i]$, the above is upper bounded by $\eta^2 \sum_{i\in A} p_i(t)\frac{(g_i(t))^2}{c_i}$. Putting together, we get that:

$$\mathbf{g}(t)\cdot\big(\mathbf{q} - \mathbf{p}(t)\big) \leq \frac{1}{\eta}\sum_{i\in S} c_i \cdot \left(q_i \cdot \log\big(\frac{p_i(t+1)}{p_i(t)}\big) + p_i(t) - p_i(t+1)\right) + \eta \sum_{i\in A} p_i(t)\frac{(g_i(t))^2}{c_i}$$

Summing over t, we have:

$$\mathbf{g}(t)\cdot\big(\mathbf{q} - \mathbf{p}(t)\big) \leq \frac{1}{\eta}\sum_{i\in S} c_i \cdot \left(q_i \cdot \log\big(\frac{p_i(T+1)}{p_i(1)}\big) + p_i(1) - p_i(T+1)\right) + \eta \sum_{t\in[T]}\sum_{i\in A} p_i(t)\frac{(g_i(t))^2}{c_i}$$

Finally, by $\log(x) \leq (x-1)$, we get that $q_i \log\big(\frac{p_i(T+1)}{q_i}\big) \leq p_i(T+1) - q_i$. Hence, we have:

$$\mathbf{g}(t)\cdot\big(\mathbf{q} - \mathbf{p}(t)\big) \leq \frac{1}{\eta}\sum_{i\in S} c_i \cdot \left(q_i \cdot \log\big(\frac{q_i}{p_i(1)}\big) + p_i(1) - q_i\right) + \eta \sum_{t\in[T]}\sum_{i\in A} p_i(t)\frac{(g_i(t))^2}{c_i}$$

The lemma then follows by our choice of the initial distribution.

A.5 Proof of OMD regret bound

In order to prove the OMD regret bound, we need some properties of Bregman divergence.

Lemma A.2 (Properties of Bregman divergence (Bubeck, 2011))**.** *Suppose $F(\cdot)$ is a Legendre function and $D_F(\cdot,\cdot)$ is its associated Bregman divergence as defined in Definition 3.1. Then:*

- *$D_F(x,y) > 0$ if $x \neq y$ as F is strictly convex, and $D_F(x,x) = 0$.*
- *$D_F(.,y)$ is a convex function for any choice of y.*
- *(*Pythagorean theorem*) If $\mathcal{A}$ is a convex set, $a \in \mathcal{A}$, $b \notin \mathcal{A}$ and $c = \underset{x\in\mathcal{A}}{\mathit{argmin}}\,(D_F(x,b))$, then*

$$D_F(a,c) + D_F(c,b) \leq D_F(a,b)$$

Given Lemma A.2, we are now ready to prove Lemma 3.1.

Proof of Lemma 3.1 To obtain the OMD regret bound, we have:

$$\begin{aligned}\mathbf{q}\cdot\mathbf{g}(t) - \mathbf{p}(t)\cdot\mathbf{g}(t) &= \frac{1}{\eta}(\mathbf{q} - \mathbf{p}(t))\cdot(\nabla F(\mathbf{w}(t+1)) - \nabla F(\mathbf{p}(t)))\\ &= \frac{1}{\eta}(D_F(qb,\mathbf{p}(t)) + D_F(\mathbf{p}(t),\mathbf{w}(t+1)) - D_F(qb,\mathbf{w}(t+1)))\\ &\overset{(1)}{\leq} \frac{1}{\eta}D_F(\mathbf{p}(t),\mathbf{w}(t+1)) + \frac{1}{\eta}\left(D_F(\mathbf{q},\mathbf{p}(t)) - D_F(\mathbf{q},\mathbf{p}(t+1))\right) \qquad (30)\end{aligned}$$

where in (1) we use $D_F(\mathbf{p}(t+1), \mathbf{w}(t+1)) \geq 0$ and $D_F(\mathbf{q}, \mathbf{p}(t+1)) + D_F(\mathbf{p}(t+1), \mathbf{w}(t+1)) \leq D_F(\mathbf{q}, \mathbf{w}(t+1))$ due to Pythagorean theorem (Lemma A.2). By summing up both hand sides of (30) for $t = 1, \cdots, T$ we have:

$$\sum_{t\in[T]} \mathbf{g}(t) \cdot \big(\mathbf{q} - \mathbf{p}(t)\big) \leq \frac{1}{\eta} \sum_{t\in[T]} D_F(\mathbf{p}(t), \mathbf{w}(t+1)) + \frac{1}{\eta} D_F(\mathbf{q}, \mathbf{p}(1)) \tag{31}$$

A.6 Symmetric range bandit regret bound - proof of Lemma 6.2

Proof of Lemma 6.2 We further define:

$$\begin{aligned} \widetilde{G}_{\text{ALG}} &\triangleq \textstyle\sum_{t\in[T]} g_{i_t}(t) = \sum_{t\in[T]} \tilde{\mathbf{p}}(t) \cdot \tilde{\mathbf{g}}(t) \ , \\ \widetilde{G}_j &\triangleq \textstyle\sum_{t\in[T]} \tilde{g}_j(t) \ . \end{aligned}$$

In expectation over the randomness of the algorithm, we have:

1. $\mathbb{E}\left[G_{\text{ALG}}\right] = \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right]$; and
2. $G_j = \mathbb{E}\left[\widetilde{G}_j\right]$ for any $j \in A$.

Hence, to upper bound $\mathbb{E}\left[\text{REGRET}_i\right] = G_i - \mathbb{E}\left[G_{\text{ALG}}\right]$, it suffices to upper bound $\mathbb{E}\left[\widetilde{G}_i - \widetilde{G}_{\text{ALG}}\right]$.

By the definition of the probability that the algorithm picks each arm, i.e., $\tilde{\mathbf{p}}(t)$, and that reward of each round is at least $-c_{\max}$, we have that:

$$\mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \geq (1-\gamma) \sum_{t\in[T]} \mathbf{p}(t) \cdot \tilde{\mathbf{g}}(t) - \gamma T c_{\max} \ .$$

Hence, for any benchmark distribution $\mathbf{q}$ over A, we have that:

$$\begin{aligned} \textstyle\sum_{j\in A} q_j \cdot \mathbb{E}\left[\widetilde{G}_j\right] - \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] &\leq \mathbb{E}\left[\textstyle\sum_{j\in A} q_j \cdot \widetilde{G}_j - \sum_{t\in[T]} \mathbf{p}(t) \cdot \tilde{\mathbf{g}}(t)\right] + \tfrac{\gamma}{1-\gamma} \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] + \tfrac{\gamma}{1-\gamma} T c_{\max} \\ &\leq \mathbb{E}\left[\textstyle\sum_{j\in A} q_j \cdot \widetilde{G}_j - \sum_{t\in[T]} \mathbf{p}(t) \cdot \tilde{\mathbf{g}}(t)\right] + 2\gamma \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] + 2\gamma T c_{\max} \\ &\leq \mathbb{E}\left[\textstyle\sum_{j\in A} q_j \cdot \widetilde{G}_j - \sum_{t\in[T]} \mathbf{p}(t) \cdot \tilde{\mathbf{g}}(t)\right] + 4\gamma T c_{\max} \ . \end{aligned} \tag{32}$$

where the 2nd inequality is due to $\gamma \leq \frac{1}{2}$, and the 3rd inequality follows by that $c_{\max}$ is the largest possible reward per round.

Next, we upper bound the 1st term on the RHS of (32). Note that $\mathbf{p}(t)$'s are the probability of choosing experts by MSMW when the experts have rewards $\tilde{\mathbf{g}}(t)$'s. By Lemma 3.4, we have that for any benchmark distribution $\mathbf{q}$ over S, the Bandit-MSMW algorithm satisfies that:

$$\sum_{j\in A} q_j \cdot \widetilde{G}_j - \sum_{t\in[T]} \mathbf{p}(t) \cdot \tilde{\mathbf{g}}(t) \leq \eta \sum_{t\in[T]} \sum_{j\in A} \frac{p_j(t)}{c_j} \cdot \left(\tilde{g}_j(t)\right)^2 + \frac{1}{\eta} \sum_{j\in A} c_j \left(q_j \ln\left(\frac{q_j}{p_j(1)}\right) - q_j + p_j(1) \right) \ . \tag{33}$$

For any $t \in [T]$ and any $j \in A$, by the definition of $\tilde{g}_j(t)$, it equals $\frac{g_j(t)}{\tilde{p}_j(t)}$ with probability $\tilde{p}_j(t)$, and equals 0 otherwise. Thus, if we fix the random coin flips in the first $t-1$ rounds and, thus, fix $\tilde{\mathbf{p}}(t)$, and take expectation over the randomness in round t, we have that:

$$\mathbb{E}\left[\frac{p_j(t)}{c_j} \cdot \left(\tilde{g}_j(t)\right)^2\right] = \frac{p_j(t)}{c_j} \cdot \tilde{p}_j(t) \cdot \left(\frac{g_j(t)}{\tilde{p}_j(t)}\right)^2 = \frac{p_j(t)}{\tilde{p}_j(t)} \frac{\left(g_j(t)\right)^2}{c_j} \ .$$

Further note that $\tilde{p}_j(t) \geq (1-\gamma)p_j(t)$, and $|g_j(t)| \leq c_j$, the above is upper bounded by $\frac{1}{1-\gamma}|g_j(t)| \leq 2|g_j(t)| \leq 2c_{\max}$. Putting together with (33), we have that for any $0 < \eta \leq \frac{\gamma}{n}$:

$$\begin{aligned}\mathbb{E}\left[\sum_{j\in A} q_j \cdot \widetilde{G}_j - \sum_{t\in[T]} \mathbf{p}(t)\cdot\tilde{\mathbf{g}}(t)\right] &\leq \eta \sum_{t\in[T]}\sum_{j\in A} 2c_{\max} + \frac{1}{\eta}\sum_{j\in A} c_j \left(q_j \ln\left(\frac{q_j}{p_j(1)}\right) - q_j + p_j(1)\right)\\ &= 2\eta T k c_{\max} + \frac{1}{\eta}\sum_{j\in A} c_j \left(q_j \ln\left(\frac{q_j}{p_j(1)}\right) - q_j + p_j(1)\right)\end{aligned}$$

Combining with (32), we have (recall that $\eta \leq \frac{\gamma}{k}$):

$$\begin{aligned}\sum_{j\in A} q_j \cdot \mathbb{E}\left[\widetilde{G}_j\right] - \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] &\leq 2\eta T k c_{\max} + \frac{1}{\eta}\sum_{j\in A} c_j \left(q_j \ln\left(\frac{q_j}{p_j(1)}\right) - q_j + p_j(1)\right) + 4\gamma T c_{\max}\\ &\leq \frac{1}{\eta}\sum_{j\in A} c_j \left(q_j \ln\left(\frac{q_j}{p_j(1)}\right) - q_j + p_j(1)\right) + 6\gamma T c_{\max}\end{aligned}$$

Let $\mathbf{q} = (1-\gamma)\mathbf{1}_i + \frac{\gamma}{k}\mathbf{1}$. Recall that $\mathbf{p}(1) = (1-\gamma)\mathbf{1}_{i_{\min}} + \frac{\gamma}{k}\mathbf{1}$ (recall $i_{\min}$ is the arm with minimum range $c_{i_{\min}}$). Similar to the discussion for the expert problem in Section 3.3, the 1st term on the RHS is upper bounded by $O\left(\frac{1}{\eta}\log\left(\frac{k}{\gamma}\right)\cdot c_i\right)$. Hence, we have:

$$\sum_{j\in A} q_j \cdot \mathbb{E}\left[\widetilde{G}_j\right] - \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \leq O\left(\frac{1}{\eta}\log\left(\frac{k}{\gamma}\right)\cdot c_i\right) + 6\gamma T c_{\max} . \tag{34}$$

Further, the LHS is lower bounded as:

$$(1-\gamma)\mathbb{E}\left[\widetilde{G}_i\right] + \frac{\gamma}{k}\sum_{j\in A}\mathbb{E}\left[\widetilde{G}_j\right] - \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] \geq (1-\gamma)\mathbb{E}\left[\widetilde{G}_i\right] - \gamma T c_{\max} - \mathbb{E}\left[\widetilde{G}_{\text{ALG}}\right] .$$

The lemma then follows by putting it back to (34) and rearranging terms.