When and How to Have Negative Regrets for Online Learners? Profits for Prediction Market Makers as an Example

Po-An Chen and and Chi-Jen Lu and Chuang-Chieh Lin and Yu-Qin Lin

Abstract

The correspondence between an online linear optimization/learning problem using Follow the Regularized Leader and a market making problem using a duality-based cost function market maker has been extensively explored. There, the goal of regret minimization for learners corresponds to the goal of minimizing the worst-case loss for market makers, which is a common objective for prediction market making design. With the insights from online learning about designing no-regret algorithms under a "predictable" or "more regular" loss environment (e.g., specifically, low variation or low deviation), which corresponds to some "patterns" of trade sequences in market making, we aim to achieve market making that furthermore guarantees profits, i.e., negative regrets, under appropriate patterns of trade sequences, which may require conditions other than those suggested by just low deviation or variation.

We propose the optimistic lazy-update online mirror descent algorithm, which can be seen as Be the Regularized Leader with the supposedly unknown current loss vector being estimated by using the "predictability" of the loss sequence. Following the framework of regret analysis for Be the Regularized Leader, we focus on analyzing the hypothetical Be the Regularized Leader with the "known" current loss vector when in each time step, a leader is "strong" compared with the other non-minimizers in terms of its much less current cumulative loss, the regret will be negative in this case, and the more frequent changes of leaders the more negative of the regret. If the immediately previous loss vector is an estimator of the current loss vector, the regret can stay negative whenever the estimation error is small. In addition, we propose a modified optimistic multiplicative update algorithm catching infrequent changes of "dominant experts" quickly enough to beat a fixed best expert in hindsight in cumulative losses thereby to obtain negative regrets. The negative regret bounds of our algorithms ensure the profit bounds of our proposed prediction market makers.

1 Introduction

We live in a world full of random events, and we actually devise ways to deal with happening of good and bad events. One example is to design *Arrow-Debreu securities* [3] whose payoffs depend on the future state of the world. Insurance contracts, options, futures, and many other financial derivatives are all real-world examples of such contingent securities. A set of securities are traded in a *securities market.* Anyone who worries about such bad events might want to buy this security as a form of insurance to hedge the risk. In addition, the information of the trader can be capitalized. The market price of the security can be viewed as traders' collective estimate of how likely some event will occur. Thus, securities markets work as mechanisms for *risk allocation* and *information aggregation.* A *complete* securities market offers at least a finite number of linearly independent securities over a set of mutually exclusive and exhaustive outcomes [2] so a trader may bet on any combination of the securities in a complete securities market to hedge any possible risk and to express any belief over the outcomes. A securities market focusing on information aggregation is called a *prediction market.*

An *automated market maker* is an institution that adaptively sets prices for each security and is willing to accept trades at these prices all the time. Automated market makers for complete markets have been well studied in economics and finance. A common goal of a prediction market making is to upper bound the worst-case loss since it makes sense for a market maker not to lose an arbitrarily large amount of money. Nevertheless, a general market maker provides liquidity to traders and seeks to "profit" from the the buy and sell prices of an asset. Profit making for a market maker has also been considered in some market making works [15, 14], none of which exploits a connection to *no-regret online learning* like those that bound the worst-case loss [7, 2] by bounding the regret that is our main technical approach (formally defined and modeled in Section 2) to achieve not only an upper bounded worst-case loss but also a lower bounded profit for *some classes* of instances.

In [7, 2], the correspondence between an online linear optimization/learning problem using the follow the regularized leader algorithm (see Appendix A.1 for details) and a market making problem using a duality-based cost function market maker has been extensively explored. There, the goal of regret minimization for learners corresponds to the goal of *minimizing the worst-case loss for market makers*, which is a common objective for prediction market making design. With the insights from online learning about designing no-regret algorithms under a "predictable" or "more regular" loss environment, e.g., specifically, low variation or low deviation [12, 9, 17], which corresponds to some "patterns" of trade sequences in market making, *we aim to achieve market making that furthermore guarantees profits, i.e., negative regrets, under appropriate patterns of trade sequences, which may require conditions other than those suggested by just low deviation or variation.*

1.1 Our Results

As the worst case loss of our prediction market making corresponds to the regret of the online learning, we first review no-regret online learning algorithms (i.e., the optimistic agile-update online mirror descent algorithm [17] and in particular, the double agile-update online mirror descent algorithm [9] as a special case). Then, we present our results on profitable market making by designing optimistic lazy-update online mirror descents and analyzing it in the "be-the-regularized-leader" framework (see, e.g., [16], restated in Appendix A.2) considering estimation errors. With linear losses, optimistic lazy-update online mirror descents can be seen as Be the Regularized Leader with the supposedly unknown current loss vector being estimated due to the "predictability" (introduced in Section 2.2).

Following the framework of regret analysis with the help of Be the Regularized Leader, we focus on analyzing Be the Regularized Leader with the known current loss vector: (i) when in each time step, there is a leader that is the "strongest" compared with the other axis non-minimizers in terms of its much less current cumulative loss. The regret will be negative in this case, and the more frequent changes of leaders the more negative of the regret, which turns out to be more profitable of the market making. Finally, if the immediately previous loss vector is an estimator of the current loss vector, the regret can stay negative whenever the estimation error is small. In addition, in the expert setting (ii) we use a modified optimistic-update multiplicative update algorithm of [8] for our purpose of catching the infrequently alternating "dominant experts" quickly enough to beat a fixed best expert in hindsight in cumulative losses (or rewards) thereby to obtain negative regrets as well as profits in market making.

In summary, we provide two types of *non-stochastic* conditions on trade sequences for cost function-based prediction market makers to make profits other than the profitable market making results and approaches of [15, 14].

1.2 Related Work

There have been results considering profitability of automated market making. Othman and Sandholm [15] designed market makers which satisfy the four desiderata for automated market makers, such as bounded loss, profitable, vanishing bid-ask spread, and unlimited market depth, which have been considered oppositional. They considered a constant-utility cost function while the quote prices to the traders involve a liquidity function and a profit function, where the former is used to increase the amount of liquidity and the latter is used to represent part of the savings. Li and Vaughan [14] extended the general framework of [2] by considering the property that price movement slows as the volume of trades gets larger. They provided an axiomatic characterization of a parameterized class of automated market makers with adaptive liquidity. Expressiveness, which measures the degree that traders can push market prices to match their beliefs, is deteriorated. Furthermore, they described and quantified factors for the market maker to make profit, such as sufficient trade volume, existence of disagreed traders and curvature of the price space.

In comparison to prediction market making, a market maker in another specific sense places *limit orders* in terms of *bids* (i.e., passive buy orders) and *asks* (i.e., passive sell orders) on an orderbook of an asset and hopes to profit by exploiting the "bid-ask spreads" of an orderbook, i.e., the difference between the bids and asks. This work is challenging due to inventory risk. To control the inventory risk, a market maker may use *market orders* to aggressively match the asks or the bids to balance the inventory. The profitability of such market making has been confirmed under some *stochastic* assumptions on price time series (e.g., mean reverting price processes [5]).

Different connections to online learning models with low regrets is closely related to this work. Abernethy and Kale [1] considered an online learning setting to extend the prior work in [5]. They proposed a class of market making strategies parameterized by the choice of bid-ask spread and liquidity and view such "spread-based" market making strategies as experts. The performance is measured in terms of *regret*, which is the difference between the learner's value in the end and that of the best strategy in the hindsight. Their no-regret algorithm resembles standard algorithms learning from expert advice, however, it keeps the *state*, represented by the asset inventory of the strategy in each round. For N spread-based strategies in T steps, they showed that the regret is bounded by $O(\sqrt{T})$, when price volatility is bounded.

2 Preliminaries

2.1 Correspondence between Market Making and Online Learning

There have been works focusing on *cost function based markets* [10, 11, 6]. Arrow-Debreu securities, each representing a potential outcome, can be offered by a simple cost function based market maker. The market maker prices each security using a differentiable cost function,which acts as a potential function. It returns the amount of money currently put in the market as a function of the number of each security's shares having been purchased. If a newly coming trader wants to make a purchase (or a sale) of a set of shares for each security, the trader has to pay (or receive, respectively) the cost function value difference (before and after the purchase) to the market maker. The price per share of an infinitesimal amount of a security is so-called the instantaneous price (defined later). In the end, payoffs have to be paid to the traders that hold the shares of the security corresponding to the realized outcome.

In a market using a *duality-based cost function market maker*, the market maker has an outcome space $\mathcal{O}$ and payoff function $\rho : \mathcal{O} \mapsto \mathbb{R}^d$, which define a feasible price space

$\Pi = \mathcal{H}(\rho(\mathcal{O}))$.[1] The market maker must select instantaneous security prices $\mathbf{x} \in \Pi$. The market maker uses a convex function $R(\cdot)$, which leads to the cost function $C(\cdot)$ via conjugate duality [2, Definition 4.1]. The market maker receives security bundle purchases $\mathbf{p}_t$ and maintains a quantity vector $\mathbf{q}_t$, updating according to $\mathbf{q}_t = \mathbf{q}_{t-1} + \mathbf{p}_t$. The market maker sets prices via $\mathbf{x}_t = \arg\max_{\mathbf{x}\in\Pi} \mathbf{x} \cdot \mathbf{q}_{t-1} - R(\mathbf{x})$ where $C(\mathbf{q}) = \max_{\mathbf{x}\in\Pi} \mathbf{x} \cdot \mathbf{q} - R(\mathbf{x})$ and the instantaneous price vector $\mathbf{x} = \nabla C(\mathbf{q})$. The market maker suffers the worst-case (in terms of $\mathbf{q}$) loss $C(\mathbf{q}_0) - C(\mathbf{q}_T) + \max_{\mathbf{x}\in\Pi} \mathbf{x} \cdot \mathbf{q}_T$.[2] It is the worst-case difference between the maximum amount that the market maker might have to pay the traders ($\max_{\mathbf{x}\in\Pi} \mathbf{x} \cdot \mathbf{q}_T$) and the amount collected from the traders by the market maker ($C(\mathbf{q}_T) - C(\mathbf{q}_0)$).

In the setting of *online convex optimization*, we describe an online game between a player and the environment. The player is given a convex set $\mathcal{K} \subseteq \mathbb{R}^d$ and has to make a sequence of decisions $\mathbf{x}_1, \mathbf{x}_2, ... \in \mathcal{K}$. After deciding $\mathbf{x}_t$, the environment reveals a convex loss function $f_t \in \mathcal{F}$, where $\mathcal{F}$ is the set of the adversary's moves, and the player obtains $f_t(\mathbf{x}_t)$. The performance of the player is measured by *regret* defined in the following. In this paper, what is closely related to our problem is a more specific problem of online linear optimization where the loss functions are linear, i.e., $f_t(\mathbf{x}) = \langle \ell_t, \mathbf{x} \rangle$ for some $\ell_t \in R^d$.

We define the player's adaptive strategy $\mathcal{L}$ as a function taking as input a subsequence of loss vectors $\ell_1, ..., \ell_{t-1}$ and returns a point $\mathbf{x}_t \leftarrow \mathcal{L}(\ell_1, ..., \ell_{t-1})$ for $\mathbf{x}_t \in \mathcal{K}$.

Definition 1 *Given an online linear optimization algorithm $\mathcal{L}$ and a sequence of loss vectors $\ell_1, \ell_2, ... \in R^d$, let the regret $Regret(\mathcal{L}; \ell_{1:T})$ be defined as* $\sum_{t=1}^{T} \langle \ell_t, \mathbf{x}_t \rangle - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} \langle \ell_t, \mathbf{x} \rangle$.[3]

A desirable property that one would want an online linear optimization algorithm to have is a regret that scales sublinearly in T. This property can be formally captured as the following.

Property 1 *For any bounded decision set $\mathcal{K} \subseteq R^d$, an algorithm $\mathcal{L}_\mathcal{K}$ that achieves $Regret(\mathcal{L}_\mathcal{K}) = o(T)$ for any sequence of loss vectors $\{\ell_t\}$ with bounded norm has the no-regret property.*

The Follow The Regularized Leader algorithm (FTRL), (e.g., see [16], restated in Appendix A.1 for reference) has been proposed as a correspondence between market making and online learning [7, 2]. We list such a correspondence in the following. Thus, we can focus on designing market makers by designing no-regret online learning algorithms. In an online linear optimization problem using FTRL, the learner is given access to a fixed space of weights $\mathcal{K}$. The learning algorithm must select a weight vector $\mathbf{x} \in \mathcal{K}$. The learner uses a convex regularizer $\mathcal{R}(\cdot)$, which is a parameter of FTRL. The learner receives a loss vector ℓ_t. The learning algorithm maintains a cumulative loss vector L_t and updates according to $L_t = L_{t-1} + \ell_t$. FTRL selects the weights by solving $\mathbf{x}_t = \arg\min_{\mathbf{x}\in\mathcal{K}} \mathbf{x} \cdot L_{t-1} + \frac{1}{\eta}\mathcal{R}(\mathbf{x})$. The learner suffers regret $\sum_{t=1}^{T} \mathbf{x}_t \cdot \ell_t - \min_{\mathbf{x}\in\mathcal{K}} \mathbf{x} \cdot L_T$.

2.2 Predictable Sequences

We use the following definition to model "regular" or "predictable" sequences [17, 9, 18]. Fix a sequence of functions $M_t : \mathcal{F}^{t-1} \to \mathcal{F}$ for each $t \in \{1, ..., T\}$. These functions give a predictable process of the environment $M_1, M_2(\ell_1), ..., M_T(\ell_1, ..., \ell_{T-1})$. Recall that the sequence $\{\ell_t\}$ corresponds to the purchase vectors $\{\mathbf{p}_t\}$ in market making. If

[1] Here $\rho_i(\mathbf{o}) \in \{0, 1\}$ for $\mathbf{o} \in \mathcal{O}$ and security i, and $\mathcal{H}(S)$ is the convex hull of a convex set S.

[2] Note that there is a difference term between $C(\mathbf{q}_T) - C(\mathbf{q}_0)$ and $\sum_{t=1}^{T} \mathbf{x}_t \cdot \mathbf{p}_t$ ignored here since $C(\mathbf{q}_T) - C(\mathbf{q}_0) = \sum_{t=1}^{T} C(\mathbf{q}_t) - C(\mathbf{q}_{t-1}) \approx \sum_{t=1}^{T} \nabla C(\mathbf{q}_{t-1}) \cdot (\mathbf{q}_t - \mathbf{q}_{t-1}) = \sum_{t=1}^{T} \mathbf{x}_t \cdot \mathbf{p}_t$, where the approximation approaches equality when t is continuous.

[3] For a player maximizing her total reward given a sequence of reward vectors, the regret can also be defined accordingly.

$M_t(\ell_1, ..., \ell_{t-1}) = \ell_t$ for all t, then $\{\ell_t\}$ forms a noiseless time series, which should suffer no regret. From the perspective of marking making, this is a predictable sequence of purchases from traders. If the actual sequence is roughly given by this predictable process, i.e., $M_t(\ell_1, ..., \ell_{t-1}) \approx \ell_t$, the sequence can be described as predictable process plus adversarial noise. Like [17, 9] (see Appendix A.3 for the optimistic agile-update mirror descent algorithm), we are also motivated by this idea of decomposition to design online learning algorithms with even negative regrets for some classes of sequences.

In [9], a measure called L_p-deviation for the loss functions $\{f_t\}$ is defined as

$$D_p = \sum_{t=1}^{T} \max_{\mathbf{x} \in \mathcal{K}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|_p^2, \tag{1}$$

which is defined in terms of sequential difference between individual loss function to its previous one. This double (agile-update) mirror descent algorithm (Algorithm 1) and the corresponding regret bounds[4] are given. In particular, M_t predicts the loss gradient at t by the the loss gradient at $t-1$, i.e., $M_t(\ell_1, ..., \ell_{t-1}) = \ell_{t-1}$. Throughout this paper, $\mathcal{B}^{\mathcal{R}}$ denotes the Bregman divergence with respect to a strictly convex function $\mathcal{R}$, in our case, the regularizer.

We focus on the case that $M_t(\ell_1, ..., \ell_{t-1}) = \ell_{t-1}$. Double agile-update online mirror descents, stated in Appendix A.4, achieve so-called *path length bounds* (Theorem 8 and 10 of [9]), which are smaller than $O(\sqrt{T})$ bounds when ℓ_{t-1} is a good proxy for ℓ_t (low deviation).

3 Optimistic Lazy-Update Online Mirror Descents with Predictable Sequences

Since it is known that, with linear losses, the standard lazy-update online mirror descent algorithm is the same as follow the regularized leader [16], which is *not* the case for the standard agile-update online mirror descent algorithm.[5] We then propose the *optimistic lazy-update online mirror descent* algorithm, which is stated in Appendix B.1.

Algorithm 1 Double Lazy-Update Online Mirror Descents

Let $\mathbf{x}_1 = \hat{\mathbf{x}}_1 = \mathbf{y}_1 = (1/d, ..., 1/d)^\top$
for $t \in [T]$ **do**
  Receive f_t and compute $\ell_t = \nabla f_t(\hat{\mathbf{x}}_t)$.
  update
  $\mathbf{x}_{t+1} = \arg\min_{\mathbf{x} \in \mathcal{K}} \mathcal{B}^{\mathcal{R}}(\mathbf{x}, \mathbf{y}_{t+1})$ for $\nabla \mathcal{R}(\mathbf{y}_{t+1}) = \nabla \mathcal{R}(\mathbf{y}_t) - \eta_t \ell_t$,
  $\hat{\mathbf{x}}_{t+1} = \arg\min_{\hat{\mathbf{x}} \in \mathcal{K}} \mathcal{B}^{\mathcal{R}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}_{t+1})$ for $\nabla \mathcal{R}(\hat{\mathbf{y}}_{t+1}) = \nabla \mathcal{R}(\mathbf{y}_{t+1}) - \eta_t \ell_t$.
end for

In particular, it can be instantiated as the *double lazy-update online mirror descent* algorithm with $M_t = \ell_{t-1}$, which is as stated in Algorithm 1. Lazy updates take gradient updates from possibly infeasible points $\mathbf{y}_t$ and $\mathbf{y}_{t+1}$ instead of feasible points $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$, respectively, in agile updates. With linear losses, the equivalence of lazy-update online mirror descents and Follow The Regularized Leader (e.g., see [16]) serves as a good reason

[4]**Theorem 8 of [9]:** When the L_2-deviation of the loss functions is D_2, the regret of our algorithm is at most $O(\sqrt{D_2})$ with a matching lower bound. In the setting of prediction with N expert advice, **Theorem 10 of [9]:** When the L_∞-deviation of the loss functions is D_∞, the regret of our algorithm is at most $O(\sqrt{D_\infty \ln N})$ with a matching lower bound.

[5]Note that agile updates and lazy updates give the same multiplicative updates; however, agile updates and lazy updates give different online mirror descents in general.

for adopting lazy updates, which furthermore motivates the work and leads to Lemma 1, an advanced version of the equivalence.

3.1 Optimistic Lazy-Update Online Mirror Descents As Be The Regularized Leader Estimated with Predictors M_{t+1}

We first show that with linear losses, Double Lazy-Update Online Mirror Descents, an instantiation of Optimistic Lazy-Update Online Mirror Descents, can be seen as Be The Regularized Leader with the unknown current loss vector estimated by the immediately previous loss vector (Lemma 1), which is proved in Appendix B.2. Following the framework of regret analysis with the help of Be the Leader (see, e.g., [16]), we then focus on analyzing Be The Leader with the known current loss vector when *in each time step, a leader is dominating the other non-minimizers in terms of cumulative loss by much.* The regret will be negative in this case, and the more frequent changes of leaders the more negative of the regret. Finally, if the immediately previous loss vector is a good estimator of the current loss vector, the regret can stay negative if the estimation error is small.

Lemma 1 *According to the first update of optimistic lazy-update online mirror descents,* $\nabla\mathcal{R}(\mathbf{y}_{t+1}) = \nabla\mathcal{R}(\mathbf{y}_t) - \eta_t \ell_t$ *and* $\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{K}} \mathcal{B}^{\mathcal{R}}(\mathbf{x}, \mathbf{y}_{t+1})$*, we have that*

$$\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{K}} \left(\eta_t \langle \sum_{s=1}^{t} \ell_s, \mathbf{x}\rangle + \mathcal{R}(\mathbf{x}) \right).$$

For the second update, $\nabla\mathcal{R}(\hat{\mathbf{y}}_{t+1}) = \nabla_t\mathcal{R}(\mathbf{y}_{t+1}) - \eta_t M_{t+1}$ *and* $\hat{\mathbf{x}}_{t+1} = \arg\min_{\hat{\mathbf{x}}\in\mathcal{K}} \mathcal{B}^{\mathcal{R}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}_{t+1})$*, we have that*

$$\hat{\mathbf{x}}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{K}} \left(\eta_t \langle \sum_{s=1}^{t} \ell_s + M_{t+1}, \mathbf{x}\rangle + \mathcal{R}(\mathbf{x}) \right).$$

In particular with $M_{t+1} = \ell_t$*, for the second update of double lazy-update online mirror descents,* $\nabla\mathcal{R}(\hat{\mathbf{y}}_{t+1}) = \nabla\mathcal{R}(\mathbf{y}_{t+1}) - \eta_t \ell_t$ *and* $\hat{\mathbf{x}}_{t+1} = \arg\min_{\hat{\mathbf{x}}\in\mathcal{K}} \mathcal{B}^{\mathcal{R}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}_{t+1})$*, we have that*

$$\hat{\mathbf{x}}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{K}} \left(\eta_t \langle \sum_{s=1}^{t} \ell_s + \ell_t, \mathbf{x}\rangle + \mathcal{R}(\mathbf{x}) \right).$$

Under the following condition in the next subsection, we will analyze its regret by decomposing it into two parts: (1) *the regret of the hypothetical Be The Regularized Leader algorithm* plus (2) *the gap in the cumulative loss between the estimated Be The Regularized Leader, i.e., Optimistic Lazy-Update Online Mirror Descents by Lemma 1, and the hypothetical Be The Regularized Leader.*

3.2 Condition: Frequent Changes of Strong Leaders

The hypothetical Be The Regularized Leader is merely a concept for the convenience of analysis. How can it be implemented? Actually, the cumulative loss of the hypothetical Be the Regularized Leader can be obtained by *evaluating the play* $\mathbf{x}_{t+1}$ *of the follow the regularized leader algorithm using the corresponding immediately previous loss* ℓ_t, i.e., $\sum_{t=1}^{T}\langle \ell_t, \mathbf{x}_{t+1}\rangle$ where $\mathbf{x}_{t+1}$ is the play of Follow The Regularized Leader. Thus, $\sum_{t=1}^{T}\langle \ell_t, \mathbf{x}_{t+1}\rangle$ is the cumulative loss of the hypothetical Be The Regularized Leader.

Let there be K changes of leaders in total in running the Follow The Regularized Leader algorithm over T time steps so there are K periods, each of which has the same minimizer, i.e., the same algorithm-selected leader. Specifically, let $L_t(\mathbf{x}) = \langle \sum_{s=1}^{t} \ell_s, \mathbf{x} \rangle$ for all t, and $\mathbf{x}_1 = \arg\min_{\mathbf{x}\in\mathcal{K}} \mathcal{R}(\mathbf{x})$ and

$$\mathbf{x}_{t_j+1} \in \arg\min_{\mathbf{x}\in\mathcal{K}} \left(L_{t_j}(\mathbf{x}) + \frac{\mathcal{R}(\mathbf{x})}{\eta} \right)$$

for $j = 1, ..., K$ such that the time steps from $t_{j-1}+1$ to t_j are dominated by *the same* algorithm-selected leader $\mathbf{x}_{(t_{j-1}+1)+1} = \ldots = \mathbf{x}_{t_j+1}$ that minimizes the current cumulative loss L_t over period j (where t_j is the last time step during this period for $\mathbf{x}_{t_j+1}$'s domination). That is, by definition, $\mathbf{x}_{t_{j-1}+1} \neq \mathbf{x}_{t_j+1}$ for $j = 1, ..., K$ and let $t_0 = 0$.

We have that $t_K = T$ so $L_t(\mathbf{x}_{t+1}) = L_T(\mathbf{x}^*)$ for $t = t_{K-1}+1, \ldots, t_K$. Note that the best fixed choice in hindsight $\mathbf{x}^*$ might be different from the algorithm selected $\mathbf{x}_{t_K+1}$ since there may be multiple minimizers for each L_{t_K}.

Lemma 2 *For all $t \in \{1, ..., T\}$, if $L_t(\mathbf{x}) - L_t(\mathbf{x}_{t+1}) \geq \delta$ for any dimension i-axis non-minimizer $\mathbf{x} \notin \arg\min_{\mathbf{x}\in\mathcal{K}}\{L_t(\mathbf{x}) + \mathcal{R}(\mathbf{x})/\eta\}$ for $i \in \{1, ..., d\}$ where $\delta > 0$ is the leader's advantage in cumulative loss, then*

$$\sum_{t=1}^{T} \langle \ell_t, \mathbf{x}_{t+1} \rangle - L_T(\mathbf{x}^*) \leq -\delta K + \frac{1}{\eta}(\mathcal{R}(\mathbf{x}^*) - \mathcal{R}(\mathbf{x}_1)).$$

Proof. We will prove this bound by induction on the number of changes K in the dominating leaders. Since $L_{t_K}(\mathbf{x}_{t_K+1}) = L_T(\mathbf{x}^*)$, it is equivalent to show that

$$\sum_{t=1}^{T} \langle \ell_t, \mathbf{x}_{t+1} \rangle - L_{t_K}(\mathbf{x}_{t_K+1}) \leq -\delta K + \frac{1}{\eta}(\mathcal{R}(\mathbf{x}^*) - \mathcal{R}(\mathbf{x}_1)).$$

The base case is when $j = 1$: by $\mathbf{x}_1 = \arg\min_{\mathbf{x}\in\mathcal{K}} R(\mathbf{x})$, $0 \leq \frac{1}{\eta}(\mathcal{R}(\mathbf{x}^*) - \mathcal{R}(\mathbf{x}_1))$. The induction hypothesis is

$$\sum_{t=1}^{t_{K-1}} \langle \ell_t, \mathbf{x}_{t+1} \rangle - L_{t_{K-1}}(\mathbf{x}_{t_{K-1}+1}) \leq -\delta(K-1) + \frac{1}{\eta}(\mathcal{R}(\mathbf{x}^*) - \mathcal{R}(\mathbf{x}_1)).$$

We derive that

$$\begin{aligned}
&\sum_{t=1}^{T} \langle \ell_t, \mathbf{x}_{t+1} \rangle - L_T(\mathbf{x}^*) = \sum_{t=1}^{t_K} \langle \ell_t, \mathbf{x}_{t+1} \rangle - L_{t_K}(\mathbf{x}_{t_K+1}) \\
= \quad & \sum_{t=1}^{t_{K-1}} \langle \ell_t, \mathbf{x}_{t+1} \rangle - L_{t_{K-1}}(\mathbf{x}_{t_K+1}) \\
= \quad & \sum_{t=1}^{t_{K-1}} \langle \ell_t, \mathbf{x}_{t+1} \rangle - L_{t_{K-1}}(\mathbf{x}_{t_{K-1}+1}) - (L_{t_{K-1}}(\mathbf{x}_{t_K+1}) - L_{t_{K-1}}(\mathbf{x}_{t_{K-1}+1})) \\
\leq \quad & -\delta(K-1) + \frac{1}{\eta}(R(\mathbf{x}^*) - R(\mathbf{x}_1)) - \delta \\
= \quad & -\delta K + \frac{1}{\eta}(\mathcal{R}(\mathbf{x}^*) - \mathcal{R}(\mathbf{x}_1)),
\end{aligned}$$

where the second equality comes from the fact that the two equivalent terms that get canceled out are $\sum_{t=t_{K-1}+1}^{t_K} \langle \ell_t, \mathbf{x}_{t+1} \rangle$ and $\sum_{t=t_{K-1}+1}^{t_K} \langle \ell_t, \mathbf{x}_{t_K+1} \rangle$, and the inequality holds by the induction hypothesis and the condition $L_t(\mathbf{x}) - L_t(\mathbf{x}_{t+1}) \geq \delta$ for all every other $\mathbf{x} \notin \arg\min_{\mathbf{x}}(L_t(\mathbf{x}) + \mathcal{R}(\mathbf{x})/\eta)$.

3.3 Profit (Negative Regret) Bounds

Actually, we are implementing the Be The Regularized Leader algorithm by estimating $M_t(\ell_1, \ldots, \ell_{t-1}) = \ell_{t-1}$ under low deviation (so ℓ_{t-1} is a good estimator) and Lemma 2. We have the following profit (regret) bound.

Theorem 3 *If $M_t = M_t(\ell_1, \ldots, \ell_{t-1}) = \ell_{t-1}$ and the condition in Lemma 2 is satisfied, then the regret of the optimistic/double lazy-update online mirror descent algorithm is*

$$\sum_{t=1}^{T} \langle \ell_t, \hat{\mathbf{x}}_t \rangle - L_T(\mathbf{x}^*) \leq O\left(\sqrt{\sum_{t=1}^{T} \|\ell_t\|_2 \|\ell_t - \ell_{t-1}\|_2}\right) - \delta K,$$

by setting $\eta_t = 1/\sqrt{\sum_{s=1}^{t-1} \|\ell_s\|_2 \|\ell_s - \ell_{s-1}\|_2}$. In other words, the profit of our proposed market maker is at least

$$\delta K - O\left(\sqrt{\sum_{t=1}^{T} \|\mathbf{p}_t\|_2 \|\mathbf{p}_t - \mathbf{p}_{t-1}\|_2}\right).$$

Proof. Observe that the regret can be decomposed into

$$\sum_{t=1}^{T} (\langle \ell_t, \hat{x}_t \rangle - \langle \ell_t, \mathbf{x}_{t+1} \rangle) + \sum_{t=1}^{T} \langle \ell_t, \mathbf{x}_{t+1} \rangle - L_T(\mathbf{x}^*),$$

where $\mathbf{x}_{t+1} = \arg\min_{\mathbf{x} \in \mathcal{K}} (\sum_{s=1}^{t} \langle \ell_s, \mathbf{x} \rangle + \frac{\mathcal{R}(\mathbf{x})}{\eta})$. First notice that in [9, Lemma 6], when $\|\cdot\|$ is a norm with dual norm $\|\cdot\|_*$ such that $\frac{1}{2}\|\mathbf{x} - \mathbf{x}'\| \leq \mathcal{B}^{\mathcal{R}}(\mathbf{x}, \mathbf{x}')$ for any $\mathbf{x}, \mathbf{x}' \in \mathcal{K}$, it is shown that

$$\langle \ell_t - \ell_{t-1}, \hat{\mathbf{x}}_t - \mathbf{x}_{t+1} \rangle \leq \|\ell_t - \ell_{t-1}\|_*^2.$$

We cannot expect a result as good as the above for $\langle \ell_t, \hat{\mathbf{x}}_t \rangle - \langle \ell_t, \mathbf{x}_{t+1} \rangle$ since there is an extra term of $\langle \ell_{t-1}, \hat{\mathbf{x}}_t - \mathbf{x}_{t+1} \rangle$ here.

We can simply have that by a generalized Cauchy-Schwartz inequality,

$$\sum_{t=1}^{T} (\langle \ell_t, \hat{\mathbf{x}}_t \rangle - \langle \ell_t, \mathbf{x}_{t+1} \rangle) \leq \sum_{t=1}^{T} \|\ell_t\|_* \|\hat{\mathbf{x}}_t - \mathbf{x}_{t+1}\|.$$

By [9, Proposition 7] whose proof is restated in Appendix B.3 for reference, which ensures that

$$\|\hat{\mathbf{x}} - \mathbf{x}_{t+1}\| \leq \|\nabla R(\hat{\mathbf{y}}_t) - \nabla \mathcal{R}(\mathbf{y}_{t+1})\|_*,$$

we have that

$$\begin{aligned} \|\hat{\mathbf{x}}_t - \mathbf{x}_{t+1}\| &\leq \|(\nabla R(\mathbf{y}_t) - \ell_{t-1}) - (\nabla R(\mathbf{y}_t) - \ell_t)\|_* \\ &\leq \|\ell_t - \ell_{t-1}\|_*, \end{aligned}$$

where $\|\cdot\| = \|\cdot\|_2/\sqrt{\eta_t}$ and $\|\cdot\|_* = \sqrt{\eta_t}\|\cdot\|_2$. Thus,

$$\begin{aligned} \sum_{t=1}^{T} (\langle \ell_t, \hat{\mathbf{x}}_t \rangle - \langle \ell_t, \mathbf{x}_{t+1} \rangle) &\leq \sum_{t=1}^{T} \|\ell_t\|_* \|\ell_t - \ell_{t-1}\|_* \\ &\leq \sum_{t=1}^{T} \eta_t \|\ell_t\|_2 \|\ell_t - \ell_{t-1}\|_2. \end{aligned}$$

Define function $F(t) = \int_0^t f(s)ds$, where continuous function $f(s)$ is defined by connecting $f(0) = 0$ and all $f(s) = \|\ell_s\|_2\|\ell_s - \ell_{s-1}\|_2$ for $s = 1, ..., t$. Along with Lemma 2, we know that the regret is at most

$$
\begin{aligned}
& \sum_{t=1}^{T} \eta_t \|\ell_t\|_2 \|\ell_t - \ell_{t-1}\|_2 + \frac{1}{\eta}(\mathcal{R}(\mathbf{x}^*) - \mathcal{R}(\mathbf{x}_1)) - \delta K \\
\leq \; & \sum_{t=1}^{T} \frac{\|\ell_t\|_2 \|\ell_t - \ell_{t-1}\|_2}{\sqrt{\sum_{s=1}^{t-1} \|\ell_s\|_2 \|\ell_s - \ell_{s-1}\|_2}} + \frac{\mathcal{R}(\mathbf{x}^*) - \mathcal{R}(\mathbf{x}_1))}{\eta} - \delta K \\
\leq \; & c \int_1^T \frac{f(t)}{\sqrt{F(t)}} dt + \frac{\mathcal{R}(\mathbf{x}^*) - \mathcal{R}(\mathbf{x}_1))}{\eta} - \delta K \\
= \; & c\sqrt{F(T)} - c' + \frac{\mathcal{R}(\mathbf{x}^*) - \mathcal{R}(\mathbf{x}_1))}{\eta} - \delta K \\
\leq \; & O\left(\sqrt{\sum_{t=1}^{T} \|\ell_t\|_2 \|\ell_t - \ell_{t-1}\|_2}\right) - \delta K,
\end{aligned}
$$

where the last inequality follows from setting $\eta = 1/\sqrt{\sum_{t=1}^T \|\ell_t\|_2\|\ell_t - \ell_{t-1}\|_2}$.

Remark 1 *Since agile updates and lazy updates give the same multiplicative updates, this result of Theorem 3 would directly hold for agile updates if the regularization function is the relative entropy.*

4 Expert Setting: Modified Optimistic Multiplicative Updates with Predictable Sequences

We use a model with experts inducing $\{0, -1\}$-losses as a special case, where each $\ell_t^{(i)} \in \{0, -1\}$ for $i \in \{1, ..., d\}$[6] in this section for the convenience of analysis. We call the *only* expert that induces a loss of -1 at a time step the *dominant expert* at a that time step.

It is not hard to show that even when there are only few changes of dominant experts in the two-expert case, the Be The Leader algorithm cannot change leaders promptly enough even if the dominant expert changes periodically and not too fast. Be The (Regularized) Leader is simply not enough. Intuitively, for our algorithm to perform well with alternating dominant experts over time, we would want to put enough probability on the current dominant expert but not too much to prevent that probability from being reduced too slow in case of the changes of dominant experts. Therefore, we will resort to a *modified double multiplicative update* algorithm that has the desirable balance properties mentioned above, for negative regrets under some class of loss sequences, which will be explained formally in the following.

4.1 Condition: Alternating Dominant Experts

In [8], they did not only consider loss functions with a small deviation, but also a natural generalization in which the regret is measured against a "dynamic" offline algorithm that can play different strategies in different rounds (under the constraint that the losses' deviation is small). Interestingly, in this paper, we are using this algorithm for a different purpose: we

[6] Here experts corresponds to the securities in market making, i.e., the number of securities.

are applying it for our purpose of catching up with the changes of dominant experts quickly enough to beat still a fixed best expert in hindsight in terms of the cumulative reward. This algorithm can be generalized into the modified optimistic multiplicative update algorithm, stated in Appendix C.1.

Note that when $M_{t+1} = \ell_t$, the modified optimistic multiplicative update algorithm mentioned above can be instantiated into the modified double multiplicative update algorithm [8, Algorithm 2] as follows (Algorithm 2).

Algorithm 2 Modified Double Multiplicative Updates ([Algorithm 2 of [8])

1: Let $\bar{\mathbf{x}}_1 = \mathbf{x}_1 = \hat{\mathbf{x}}_1 = (1/d, ..., 1/d)^\top$
2: **for** $t \in [T]$ **do**
3: Receive f_t and compute $\ell_t = \nabla f_t(\hat{\mathbf{x}}_t)$.
4: **for** $i \in [N]$ **do**
5: update
$\bar{\mathbf{x}}_{t+1}^{(i)} = \mathbf{x}_t^{(i)} e^{-\eta \ell_t^{(i)}} / \bar{Z}_{t+1}$ with $\bar{Z}_{t+1} = \sum_j \mathbf{x}_t^{(j)} e^{-\eta \ell_t^{(j)}}$, and $\mathbf{x}_{t+1}^{(i)} = (1-\beta)\bar{\mathbf{x}}_{t+1}^{(i)} + \beta/N$,
$\hat{\mathbf{x}}_{t+1}^{(i)} = \mathbf{x}_{t+1}^{(i)} e^{-\eta \ell_t^{(i)}} / \hat{Z}_{t+1}$ with $\hat{Z}_{t+1} = \sum_j \mathbf{x}_{t+1}^{(j)} e^{-\eta \ell_t^{(j)}}$.
6: **end for**
7: **end for**

4.2 Profit (Negative Regret) Bounds

For the simplicity of illustration, we consider the two-expert case, i.e., $d = 2$ for the analysis. Suppose that there are $K + 1$ periods of time where each period is defined by its dominant expert for a constantly large $K = O(1)$ with respect to T. We have that $\sum_{i=1}^{K+1} T_i = T$. Since we have two experts, the dominant expert of a period is alternating between the two experts and $D_\infty = K$. We set $\beta = 1/t$. At the end of the first period whose time step is T_1, the dominant expert can *never* have a probability greater than T_1 times the other expert's probability. And it only takes relatively short time for the other expert to offset in the next period. The proofs are deferred to Appendix C.2 and C.3, respectively.

Lemma 4 *At time step T_1, the end of the first period, the probability of the first expert who is the current dominant expert for the first period and that of the other expert is in a ratio at most $O(T_1)$.*

Lemma 5 *After the first period of time steps T_1, it takes at most $O(\log(T_1)/\eta)$ time steps for the ratio (of the first expert's probability to the probability of the other expert who is the current dominant expert) to achieve at most 1 again.*

This means that there is always enough time before the end of this current period for the current dominant expert to offset the probability of the other expert who was the dominant expert of the immediately previous period and to accumulate enough probability. This is proved in Appendix C.4.

Lemma 6 *After the ith period of time steps, it takes at most $O(\log(T_i)/\eta)$ time steps for the ratio of the previous dominant expert's probability to the probability of the other expert who is the current dominant expert (of the $(i + 1)$th period) to achieve at most 1.*

Then, we have the main result of this section.

Theorem 7 *The regret of the modified double multiplicative updates is at most*

$$-T/2 + \sum_{i=1}^{K} \log(T_i)/\eta.$$

With $\eta = O(1/\sqrt{D_\infty}) = O(1/\sqrt{K})$,[7] *the regret is at most $-T/2 + K^{3/2} \log(T/K)$. In other words, the profit of our proposed market maker is at least*

$$T/2 - K^{3/2} \log(T/K).$$

Proof. The cumulative (expected) reward of the modified double multiplicative updates is by Lemma 4, Lemma 5 (a special case of Lemma 6) and Lemma 6.

$$T_1 + \sum_{i=1}^{K} (T_{i+1} - \log(T_i)/\eta) = T - \sum_{i=1}^{K} \log(T_i)/\eta.$$

The cumulative reward of the best in hindsight is $T/2$. Thus, the regret is

$$-T/2 + \sum_{i=1}^{K} \log(T_i)/\eta.$$

To have a negative regret, η has to be set to be greater than $2\sum_{i=1}^{K} \log(T_i)/T$. With $\eta = 1/\sqrt{D_\infty}$, since $D_\infty = K$, we obtain a regret at most

$$-T/2 + K^{3/2} \sum_{i=1}^{K} \log(T_i)/K \leq -T/2 + K^{3/2} \log(T/K),$$

where the inequality holds by the concavity of logarithm.

We have a result for the special case when $T_i = T/(K+1)$ for $i = 1, ..., K+1$, which is proved in Appendix C.5.

Corollary 8 *When T is equally divided into $K+1$ periods, the regret of the modified double multiplicative updates is at most $-T/2 + K \log(T/(K+1))/\eta$. With $\eta = O(1/\sqrt{D_\infty}) = O(1/\sqrt{K})$, the regret is at most*

$$-T/2 - \sqrt{K} \log(T/(K+1)) + K^{\frac{3}{2}} \log(T/K).$$

In other words, the profit of our proposed market maker is at least

$$T/2 + \sqrt{K} \log(T/(K+1)) - K^{\frac{3}{2}} \log(T/K).$$

5 Simulations and Numerical Results

We conduct two numerical experiments to verify our theoretical results in terms of bounding the profits (negative regrets) for the proposed market makers under the two corresponding conditions in Section 3 and 4 (on sequences with low deviation).

In the 1st experiment, we construct 100 sequences of loss vectors ranging between -1 and 1 each dimension that have low deviation in terms of (1) and at the same time satisfy the condition in Lemma 2 for running the optimistic lazy-update mirror descent algorithm to induce negative regrets, i.e., profits, theoretically guaranteed by Theorem 3 and numerically

[7] The learning rate can also be set adaptively as the one in the previous section.

Table 1: Profits of 10 (out of 100) sequences of loss vectors with $T = 100$

	Profits
seq. 1	61.83141206
seq. 2	51.33172529
seq. 3	59.72396573
seq. 4	32.07606561
seq. 5	24.49051896
seq. 6	20.79662341
seq. 7	33.56186039
seq. 8	60.73038448
seq. 9	32.78915073
seq. 10	37.79053322

Table 2: Profits of 3 sequences of $\{-1, 0\}$-loss vectors with $T = 100$

$K + 1$	Profits
2	43.89414975
4	30.24045795
5	22.79767258

with an average profit of 40.95948428. In the 2nd experiment, we construct sequences with two periodically alternating dominant experts to induce profits guaranteed by Theorem 7 and Corollary 8. In the following, some detailed numerical results are listed in Table 1 and 2, respectively, for optimistic lazy-update online mirror descents and modified multiplicative updates.

6 Conclusions and Future Work

We use the correspondence between designing online no-regret algorithms for negative regrets under some predictable (or more regular) loss environments and prediction market making under some patterns of trade sequences. We provide two types of non-stochastic conditions on trade sequences for cost function-based prediction market makers to make profits other than the profitable market making results and approaches of [15, 14].

In this line work of prediction market making [7, 2] and this paper, the market maker sets the price and the traders can bet on the future trends of underlying securities. In another line of work on market making, as discussed in [1, 5], a "passive" market maker makes profit of a security basically by placing limit orders as bids and asks to reap the bid-ask spreads on the orderbook, yet takes exposure risk of the inventory. A promising direction of market making design to try is to combine the merits of both the designs. As an efficient prediction market maker sets the security's price with no regret, one can decide to either replace the announced market price by the predicted prices and place the bids and asks accordingly, or clear the inventory to avoid the inventory risk especially for a large price movement in a single direction. We conjecture that this prediction market making helps reduce the inventory risk while the overall profit can be still guaranteed in terms of risk-adjusted return [13].

Acknowledgements.

We would like to thank Yiling Chen for useful discussions.

References

[1] J. Abernethy and S. Kale. Adaptive market making via online learning. In *Proc. of Conference on Neural Information Processing Systems*, 2013.

[2] J. Abernethy, Y. Chen, and J. W. Vaughan. Efficient market making via convex optimization, and a connection to online learning. *ACM Transactions on Economics and Computation*, 1(1), 2012.

[3] K. J. Arrow. The role of securities in the optimal allocation of risk-bearing. *Review of Economic Studies*, 31(2), 1964.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

[5] T. Chakraborty and M. Kearns. Market making and mean reversion. In *Proc. 12th ACM Conference on Electronic Commerce*, 2011.

[6] Y. Chen and D. M. Pennock. A utility framework for bounded-loss market makers. In *Proc. 23rd Conference on Uncertainty in Artificial Intelligence*, 2007.

[7] Y. Chen and J. W. Vaughan. A new understanding of prediction markets via no-regret learning. In *Proc. 11th ACM Conference on Electronic Commerce*, 2010.

[8] C.-K. Chiang, C.-J. Lee, and C.-J. Lu. Online learning in a gradually evolving world, 2011. Unpublished manuscript.

[9] C.-K. Chiang, T. Yang, C.-J. Lee, M. Mahdavi, C.-J. Lu, and S. Zhu. Online optimization with gradual variations. In *Proc. Conference on Learning Theory*, 2012.

[10] R. Hanson. Combinatorial information market design. *Information Systems Frontiers*, 5(1), 2003.

[11] R. Hanson. Logarithmic market scoring rules for modular combinatorial information aggregation. *Journal of Prediction Markets*, 1(1), 2007.

[12] E. Hazan and S. Kale. Extracting certainty from uncertainty: regret bounded by variation in costs. In *Proc. Conference on Learning Theory*, 2008.

[13] R. Herring, F. X. Diebold, and N. A. Doherty. *The Known, the Unknown, and the Unknowable in Financial Risk Management: Measurement and Theory Advancing Practice.* Princeton University Press, 2010.

[14] X. Li and J. W. Vaughan. An axiomatic characterization of adaptive-liquidity market makers. In *Proc. ACM Conference on Electronic Commerce*, 2013.

[15] A. Othman and T. Sandholm. Profit-charging market makers with bounded loss, vanishing bid/ask spreads, and unlimited market depth. In *Proc. ACM Conference on Electronic Commerce*, 2012.

[16] A. Rakhklin. Lecture notes on online learning. 2009. Draft.

[17] A. Rakhlin and K. Sridharan. Online learning with predictable sequences. In *Proc. Conference on Learning Theory*, 2013.

[18] A. Rakhlin, K. Sridharan, and A. Tewari. Online learning: stochastic, constrained, and smooth adversaries. In *Proc. Conference on NIPS*, 2011.

Po-An Chen
Institute of Information Management, National Yang Ming Chiao Tung University
Hsinchu, Taiwan
Email: poanchen@nycu.edu.tw

Chi-Jen Le
Institute of Information Science, Academia Sinica
Taipei, Taiwan
Email: cjlu@iis.sinica.edu.tw

Chuang-Chieh Lin
Department of Computer Science, Tamkang University
New Taipei, Taiwan
Email: 158778@mail.tku.edu.tw

Yu-Qin Lin
Institute of Information Management, National Yang Ming Chiao Tung University
Hsinchu, Taiwan
Email: athenalin1016@gmail.com

A.1 Follow The Regularized Leader Algorithm

Algorithm 3 Follow the Regularized Leader (FTRL)

1: **for** $t \in [T]$ **do**

2:
$$\mathbf{x}_t = \arg\min_{\mathbf{x}\in\mathcal{K}}[\eta \sum_{s=1}^{t-1} \ell_s(\mathbf{x}) + \mathcal{R}(\mathbf{x})].$$

3: **end for**

A.2 Be The Regularized Leader Algorithm

Algorithm 4 Be the Regularized Leader (BTRL)

1: **for** $t \in [T]$ **do**

2:
$$\mathbf{x}_t = \arg\min_{\mathbf{x}\in\mathcal{K}}[\eta \sum_{s=1}^{t} \ell_s(\mathbf{x}) + \mathcal{R}(\mathbf{x})].$$

3: **end for**

A.3 Optimistic Agile-Update Mirror Descent Algorithm

A.4 Double Agile-Update Mirror Descent Algorithm

Algorithm 5 Optimistic (Agile-Update) Online Mirror Descent Algorithm [17]

Let $\hat{\mathbf{x}}_1 = \mathbf{x}_1 = \arg\min_{\mathbf{x}} \mathcal{R}(\mathbf{x})$ with $\mathcal{R}$ a 1-strongly convex function w.r.t. a norm $\|\cdot\|$
for $t \in [T]$ **do**
  Receive $\hat{\mathbf{x}}_t$
  update
  $\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{K}} \eta_t\langle \mathbf{x}, \ell_t\rangle + \mathcal{B}^{\mathcal{R}}(\mathbf{x}, \mathbf{x}_t)$,
  $\hat{\mathbf{x}}_{t+1} = \arg\min_{\hat{\mathbf{x}}\in\mathcal{K}} \eta_t\langle \hat{\mathbf{x}}, M_{t+1}\rangle + \mathcal{B}^{\mathcal{R}}(\hat{\mathbf{x}}, \mathbf{x}_{t+1})$.
end for

Algorithm 6 Double Agile-Update Online Mirror Descents (Algorithm 1 of [9])

Let $\mathbf{x}_1 = \hat{\mathbf{x}}_1 = (1/d, ..., 1/d)^\top$
for $t \in [T]$ **do**
  Receive f_t and compute $\ell_t = \nabla f_t(\hat{\mathbf{x}}_t)$.
  update
  $\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{K}} \mathcal{B}^{\mathcal{R}}(\mathbf{x}, \mathbf{y}_{t+1})$ for $\nabla\mathcal{R}(\mathbf{y}_{t+1}) = \nabla\mathcal{R}(\mathbf{x}_t) - \eta_t\ell_t$,
  $\hat{\mathbf{x}}_{t+1} = \arg\min_{\hat{\mathbf{x}}\in\mathcal{K}} \mathcal{B}^{\mathcal{R}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}_{t+1})$ for $\nabla\mathcal{R}(\hat{\mathbf{y}}_{t+1}) = \nabla\mathcal{R}(\mathbf{x}_{t+1}) - \eta_t\ell_t$.
end for

B.1 Optimistic Lazy-Update Mirror Descent Algorithm

Algorithm 7 Optimistic Lazy-Update Online Mirror Descents

Let $\mathbf{x}_1 = \hat{\mathbf{x}}_1 = \mathbf{y}_1 = (1/d, ..., 1/d)^\top$
for $t \in [T]$ **do**
  Receive f_t and compute $\ell_t = \nabla f_t(\hat{\mathbf{x}}_t)$.
  update
  $\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{K}} \mathcal{B}^{\mathcal{R}}(\mathbf{x}, \mathbf{y}_{t+1})$ for $\nabla\mathcal{R}(\mathbf{y}_{t+1}) = \nabla\mathcal{R}(\mathbf{y}_t) - \eta_t\ell_t$,
  $\hat{\mathbf{x}}_{t+1} = \arg\min_{\hat{\mathbf{x}}\in\mathcal{K}} \mathcal{B}^{\mathcal{R}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}_{t+1})$ for $\nabla\mathcal{R}(\hat{\mathbf{y}}_{t+1}) = \nabla\mathcal{R}(\mathbf{y}_{t+1}) - \eta_t M_{t+1}$.
end for

B.2 Proof of Lemma 1

Observe that the unconditioned minimum

$$\hat{\mathbf{x}}^*_{t+1} = \arg\min_{\mathbf{x}\in\mathbb{R}^n} \{\langle \sum_{s=1}^{t} \ell_s + M_{t+1}, \mathbf{x}\rangle + \frac{1}{\eta_t}\mathcal{R}(\mathbf{x})\}$$

satisfies, by the first-order condition,

$$\nabla\mathcal{R}(\hat{\mathbf{x}}^*_{t+1}) = -\eta_t \left(\sum_{s=1}^{t} \ell_s + M_{t+1}\right).$$

By definition, $\nabla\mathcal{R}(\mathbf{y}_{t+1}) = -\eta_t \sum_{s=1}^{t} \ell_s$ from the recurrence relation of $\mathbf{y}_t$ and $\nabla\mathcal{R}(\hat{\mathbf{y}}_{t+1}) = \nabla\mathcal{R}(\mathbf{y}_{t+1}) - \eta_t M_{t+1}$ so

$$\nabla\mathcal{R}(\hat{\mathbf{y}}_{t+1}) = -\eta_t \left(\sum_{s=1}^{t} \ell_s + M_{t+1}\right).$$

Since $\mathcal{R}(\cdot)$ is strictly convex, there is only one solution for the equation, and thus $\hat{\mathbf{y}}_{t+1} = \hat{\mathbf{x}}^*_{t+1}$. Moreover, we have that

$$\begin{aligned}
& \mathcal{B}^{\mathcal{R}}(\mathbf{x}, \hat{\mathbf{y}}_{t+1}) \\
= \quad & \mathcal{R}(\mathbf{x}) - \mathcal{R}(\hat{\mathbf{y}}_{t+1}) - \langle \nabla_t \mathcal{R}(\hat{\mathbf{y}}_{t+1}), \mathbf{x} - \hat{\mathbf{y}}_{t+1} \rangle \\
= \quad & \mathcal{R}(\mathbf{x}) - \mathcal{R}(\hat{\mathbf{y}}_{t+1}) + \eta_t \langle \sum_{s=1}^{t} \ell_s + M_{t+1}, \mathbf{x} - \hat{\mathbf{y}}_{t+1} \rangle.
\end{aligned}$$

Minimizing $\mathcal{B}^{\mathcal{R}}(\mathbf{x}, \hat{\mathbf{y}}_{t+1})$ over $\mathbf{x} \in \mathcal{K}$ is equivalent to minimizing $\eta(\sum_{s=1}^{t} \ell_s + M_{t+1})^T \mathbf{x} + \mathcal{R}(\mathbf{x})$ over $\mathbf{x} \in \mathcal{K}$.

B.3 Proof of [9, Proposition 7]

From the property of a norm, we know that

$$\frac{1}{2}\|\hat{\mathbf{x}}_t - \mathbf{x}_{t+1}\| \leq \mathcal{R}(\hat{\mathbf{x}}_t) - \mathcal{R}(\mathbf{x}_{t+1}) - \langle \nabla \mathcal{R}(\mathbf{x}_{t+1}), \hat{\mathbf{x}}_t - \mathbf{x}_{t+1} \rangle,$$

and

$$\frac{1}{2}\|\mathbf{x}_{t+1} - \hat{\mathbf{x}}_t\| \leq \mathcal{R}(\mathbf{x}_{t+1}) - \mathcal{R}(\hat{\mathbf{x}}_t) - \langle \nabla \mathcal{R}(\hat{\mathbf{x}}_t), \mathbf{x}_{t+1} - \hat{\mathbf{x}}_t \rangle.$$

Combining these two bounds, we derive that

$$\|\hat{\mathbf{x}}_t - \mathbf{x}_{t+1}\|^2 \leq \langle \nabla \mathcal{R}(\hat{\mathbf{x}}_t) - \nabla \mathcal{R}(\mathbf{x}_{t+1}), \hat{\mathbf{x}}_t - \mathbf{x}_{t+1} \rangle \tag{2}$$

We need the following well-known fact. See [4, p.139-140] for the proof. Let $\mathcal{X} \subseteq \mathbb{R}^n$ be a convex set and $\mathbf{x} = \arg\min_{\mathbf{z} \in \mathcal{X}} \phi(\mathbf{z})$ for some continuous and differentiable function $\phi : \mathcal{X} \to \mathbb{R}$. Then for any $\mathbf{w} \in \mathcal{X}$,
$\langle \nabla \phi(\mathbf{x}), \mathbf{w} - \mathbf{x} \rangle \geq 0$.

By $\phi(\mathbf{z}) = \mathcal{B}^{\mathcal{R}}(\mathbf{z}, \hat{\mathbf{y}}_t)$, we have $\hat{\mathbf{x}}_t = \arg\min_{\mathbf{z} \in \mathcal{X}} \phi(\mathbf{z})$, $\nabla \phi(\hat{\mathbf{x}}_t) = \nabla \mathcal{R}_t(\hat{\mathbf{x}}_t) - \nabla \mathcal{R}_t(\hat{\mathbf{y}}_t)$, and

$$\langle \nabla \mathcal{R}_t(\hat{\mathbf{x}}_t) - \nabla \mathcal{R}_t(\hat{\mathbf{y}}_t), \mathbf{x}_{t+1} - \hat{\mathbf{x}}_t \rangle \geq 0.$$

By $\phi(\mathbf{z}) = \mathcal{B}^{\mathcal{R}}(\mathbf{z}, \mathbf{y}_{t+1})$, we have $\mathbf{x}_{t+1} = \arg\min_{\mathbf{z} \in \mathcal{X}} \phi(\mathbf{z})$, $\nabla \phi(\mathbf{x}_{t+1}) = \nabla \mathcal{R}_t(\mathbf{x}_{t+1}) - \nabla \mathcal{R}_t(\mathbf{y}_{t+1})$, and

$$\langle \nabla \mathcal{R}_t(\mathbf{x}_{t+1}) - \nabla \mathcal{R}_t(\mathbf{y}_{t+1}), \hat{\mathbf{x}}_t - \mathbf{x}_{t+1} \rangle \geq 0.$$

Combining these two bounds, we obtain that

$$\begin{aligned}
& \langle \nabla \mathcal{R}_t(\hat{\mathbf{x}}_t) - \nabla \mathcal{R}_t(\mathbf{x}_{t+1}), \hat{\mathbf{x}}_t - \mathbf{x}_{t+1} \rangle \\
\leq \quad & \langle \nabla \mathcal{R}_t(\hat{\mathbf{y}}_t) - \nabla \mathcal{R}_t(\mathbf{y}_{t+1}), \hat{\mathbf{x}}_t - \mathbf{x}_{t+1} \rangle
\end{aligned} \tag{3}$$

By Inequalities(2) and (3), we have that

$$\begin{aligned}
\|\hat{\mathbf{x}}_t - \mathbf{x}_{t+1}\|^2 & \leq \langle \nabla \mathcal{R}_t(\hat{\mathbf{y}}_t) - \nabla \mathcal{R}_t(\mathbf{y}_{t+1}), \hat{\mathbf{x}}_t - \mathbf{x}_{t+1} \rangle \\
& \leq \|\nabla \mathcal{R}_t(\hat{\mathbf{y}}_t) - \nabla \mathcal{R}_t(\mathbf{y}_{t+1})\|_* \|\hat{\mathbf{x}}_t - \mathbf{x}_{t+1}\|,
\end{aligned}$$

where the last inequality holds by a generalized Cauchy-Schwartz inequality. We have the proposition.

C.1 Modified Optimistic Multiplicative Updates

Algorithm 8 Modified Optimistic Multiplicative Updates

1: Let $\bar{\mathbf{x}}_1 = \mathbf{x}_1 = \hat{\mathbf{x}}_1 = (1/d, ..., 1/d)^\top$
2: **for** $t \in [T]$ **do**
3: Receive f_t and compute $\ell_t = \nabla f_t(\hat{\mathbf{x}}_t)$.
4: **for** $i \in [N]$ **do**
5: update
$\bar{\mathbf{x}}_{t+1}^{(i)} = \mathbf{x}_t^{(i)} e^{-\eta \ell_t^{(i)}} / \bar{Z}_{t+1}$ with $\bar{Z}_{t+1} = \sum_j \mathbf{x}_t^{(j)} e^{-\eta \ell_t^{(j)}}$, and $\mathbf{x}_{t+1}^{(i)} = (1-\beta)\bar{\mathbf{x}}_{t+1}^{(i)} + \beta/d$,
$\hat{\mathbf{x}}_{t+1}^{(i)} = \mathbf{x}_{t+1}^{(i)} e^{-\eta M_{t+1}^{(i)}} / \hat{Z}_{t+1}$ with $\hat{Z}_{t+1} = \sum_j \mathbf{x}_{t+1}^{(j)} e^{-\eta M_{t+1}^{(j)}}$.
6: **end for**
7: **end for**

C.2 Proof of Lemma 4

At time step T_1, we use the following proposition to claim that the probability of the first expert is at most c for a constant c while that of the other expert is at least $1/T_1$ according to the first probability update in Algorithm 3, and thus, the ratio of these two probabilities is at most $c \cdot T_1$.

Proposition 9 *At time step t, the ratio of the probability of the first expert to that of the other expert is at most ct for some constant c.*

We show this by induction. The induction hypothesis is that the probability of the first expert is at most c for some constant c while that of the other expert is at least $1/(t-1)$ at times step $t-1 \geq 2$. Thus, the ratio of these two probabilities is at most $c(t-1)$.

The base case is when $t = 2$: the probability of the first expert is $1 \cdot e^\eta(1-1/2) + (1 \cdot e^\eta + 1)/(2 \cdot 2) = e^\eta/2 + (e^\eta+1)/4 = (3e^\eta+1)/4 = c$; the probability of the other expert is $1 \cdot 1 \cdot (1-1/2) + (1 \cdot e^\eta + 1)/(2 \cdot 2) = 1/2 + (e^\eta+1)/4 = (e^\eta+3)/4 \geq 1/2$. The base case holds. The induction step works as follows: the probability of the first expert is at most

$$\begin{aligned} & c \cdot e^\eta(1-1/t) + (c \cdot e^\eta + 1/(t-1) \cdot 1)/(2 \cdot t) \\ = \; & c \cdot e^\eta - c \cdot e^\eta/(2t) + 1/(2t(t-1)) \leq c' \end{aligned}$$

for some constant c'; the probability of the other expert is at least

$$\begin{aligned} & 1/(t-1) \cdot 1 \cdot (1-1/t) + (c \cdot e^\eta + 1/(t-1) \cdot 1)/(2 \cdot t) \\ = \; & 1/(t-1) - 1/(2t(t-1)) + (c \cdot e^\eta)/(2t) \geq 1/t. \end{aligned}$$

Hence, the ratio of these two probabilities is at most ct.

C.3 Proof of Lemma 5

At time step $T_1 + 1$, the ratio of the first expert's probability to the probability of the other expert who is the current dominant expert is

$$\begin{aligned} & \frac{(c \cdot T_1/\bar{Z}_{T_1+1})(1-1/(T_1+1)) + 1/(2(T_1+1))}{(e^\eta/\bar{Z}_{T_1+1})(1-1/(T_1+1)) + 1/(2(T_1+1))} \\ & = \frac{c \cdot T_1(1-1/(T_1+1)) + \bar{Z}_{T_1+1}/(2(T_1+1))}{e^\eta(1-1/(T_1+1)) + \bar{Z}_{T_1+1}/(2(T_1+1))} \\ & \leq c \cdot \frac{T_1}{e^\eta} = c \cdot T_1 \cdot e^{-\eta}, \end{aligned}$$

where $\bar{Z}_{T_1+1} = c \cdot T_1 + e^{\eta}$ and the inequality is from the assumption that $c \cdot T_1 \geq e^{\eta}$. Thus, at time step $T_1 + k$, the ratio is at most

$$c \cdot T_1 \cdot e^{-k\eta} = 1,$$

which implies that $k = \log(T_1)/\eta$.

C.4 Proof of Lemma 6

At time step $\sum_{j=1}^{i} T_j$, the end of the ith period, the probability of the first expert who was the (immediately previous) dominant expert for the $(i-1)$th period and that of the other expert who is the current dominant expert is in a ratio at least c/T_i for a constant c if $i \geq 2$ is even; the probability of the first expert who is the current dominant expert and that of the other expert who is the previous dominant expert for the $(i-1)$th period is in a ratio at most $c'(T_{i-1})$ for a constant c' if $i \geq 3$ is odd. Thus, similarly by Lemma 5, we have the lemma.

C.5 Proof of Corollary 8

The cumulative (expected) reward of the modified double multiplicative updates is by Lemma 4, Lemma 5 (a special case of Lemma 6) and Lemma 6.

$$\begin{aligned} & T/(K+1) + (T/(K+1) - \log(T/(K+1))/\eta)K \\ = \quad & T - K\log(T/(K+1))/\eta. \end{aligned}$$

The cumulative reward of the best in hindsight is $T/2$. Thus, the regret is

$$-T/2 + K\log(T/(K+1))/\eta.$$

To have a negative regret, η has to be set to be greater than $2K\log(T/(K+1))/T$. With $\eta = 1/\sqrt{D_\infty}$, since $D_\infty = K$, we obtain a regret at most $-T/2 - \sqrt{K}\log(T/(K+1)) + K^{\frac{3}{2}}\log(T/(K+1)) \leq -T/2 - \sqrt{K}\log(T/(K+1)) + K^{\frac{3}{2}}\log(T/K)$.