# Designing smoothing functions for improved worst-case competitive ratio in online optimization

**Authored by:**

Maryam Fazel
Reza Eghbali

**Abstract**

Online optimization covers problems such as online resource allocation, online bipartite matching, adwords (a central problem in e-commerce and advertising), and adwords with separable concave returns. We analyze the worst case competitive ratio of two primal-dual algorithms for a class of online convex (conic) optimization problems that contains the previous examples as special cases defined on the positive orthant. We derive a sufficient condition on the objective function that guarantees a constant worst case competitive ratio (greater than or equal to $\frac{1}{2}$) for monotone objective functions. We provide new examples of online problems on the positive orthant % and the positive semidefinite cone that satisfy the sufficient condition. We show how smoothing can improve the competitive ratio of these algorithms, and in particular for separable functions, we show that the optimal smoothing can be derived by solving a convex optimization problem. This result allows us to directly optimize the competitive ratio bound over a class of smoothing functions, and hence design effective smoothing customized for a given cost function.

## 1 Paper Body

Given a proper convex cone K ? Rn , let ? : K 7? R be an upper semi-continuous concave function. Consider the optimization problem Pm maximize ? ( t=1 At xt ) (1) subject to xt ? Ft , ?t ? [m], where for all t ? [m] := {1, 2, . . . , m}, xt ? Rl are the optimization variables and Ft are compact convex constraint sets. We assume At ? Rn?l maps Ft to K; for example, when K = Rn+ and Ft ? Rl+ , this assumption is satisfied if At has nonnegative entries. We consider problem (1) in the online setting, where it can be viewed as a sequential game between a player (online algorithm) and an adversary. At each step t, the adversary reveals At , Ft and the algorithm chooses x ?t ? Ft . The performance of the algorithm is measured by its competitive ratio, i.e., the ratio of objective value at x ?1 , .

. . . , x $?m$ to the offline optimum. Problem (1) covers (convex relaxations of) various online combinatorial problems including online bipartite matching [14], the ?adwords? problem [16], and the secretary problem [15]. More generally, it covers online linear programming (LP) [6], online packing/covering with convex cost [3, 4, 7], and generalization of adwords [8]. In this paper, we study the case where $??(u) ? K ?$ for all $u ? K$, i.e., $?$ is monotone with respect to the cone K. The competitive performance of online algorithms has been studied mainly under the worst-case model (e.g., in [16]) or stochastic models (e.g., in [15]). In the worst-case model one is interested in lower bounds on the competitive ratio that hold for any $(A1 , F1 ), . . . , (Am , Fm )$. In stochastic models, adversary choses a probability distribution from a family of distributions to generate

$(A1 , F1 ), . . . , (Am , Fm )$, and the competitive ratio is calculated using the expected value of the algorithm?s objective value. Online bipartite matching and its generalization, the ?adwords? problem, are the two main problems that have been studied under the worst case model. The greedy algorithm achieves a competitive ratio of 1/2 while the optimal algorithm achieves a competitive ratio of 1?1/e (as bid to budget ratio goes to zero) [16, 5, 14, 13]. A more general version of Adwords in which each agent (advertiser) has a concave cost has been studied in [8]. The majority of algorithms proposed for the problems mentioned above rely on a primal-dual framework [5, 6, 3, 8, 4]. The differentiating point among the algorithms is the method of updating the dual variable at each step, since once the dual variable is updated the primal variable can be assigned using a simple complementarity condition. A simple and efficient method of updating the dual variable is through a first order online learning step. For example, the algorithm stated in [9] for online linear programming uses mirror descent with entropy regularization (multiplicative weight updates algorithm) once written in the primal dual language. Recently, the work in [9] was independently extended to random permutation model in [12, 2, 11]. In [2], the authors provide competitive difference bound for online convex optimization under random permutation model as a function of the regret bound for the online learning algorithm applied to the dual. In this paper, we consider two versions of the greedy algorithm for problem (1), a sequential update and a simultaneous update algorithm. The simultaneous update algorithm, Algorithm 2, provides a direct saddle-point representation of what has been described informally in the literature as ?continuous updates? of primal and dual variables. This saddle point representation allows us to generalize this type of updates to non-smooth function. In section 2, we bound the competitive ratios of the two algorithms. A sufficient condition on the objective function that guarantees a non-trivial worst case competitive ratio is introduced. We show that the competitive ratio is at least 12 for a monotone non-decreasing objective function. Examples that satisfy the sufficient condition (on the positive orthant and the positive semidefinite cone) are given. In section 3, we derive optimal algorithms, as variants of greedy algorithm applied to a smoothed version of ?. For example, Nesterov smoothing provides optimal algorithm for the adwords problem. The main contribution of this paper is to show how one can derive the optimal smoothing function (or from the dual point of view the optimal

2

regularization function) for separable $\Omega$ on positive orthant by solving a convex optimization problem. This gives an implementable algorithm that achieves the optimal competitive ratio derived in [8]. We also show how this convex optimization can be modified for the design of smoothing function specifically for the sequential algorithm. In contrast, [8] only considers continuous updates. The algorithms considered in this paper and their general analysis are the same as those we considered in [10]. In [10], the focus is on non-monotone functions and online problems on the positive semidefinite cone. However, the focus of this paper is on monotone functions on the positive orthant. In [10], we only considered Nesterov smoothing and only derived competitive ratio bounds for the simultaneous algorithm. Notation. Given a function $\phi : \mathbb{R}^n \mapsto \mathbb{R}$, $\phi^*$ denotes the concave conjugate of $\phi$ defined as $\phi^*(y) = \inf_u \langle y, u \rangle - \phi(u)$, for all $y \in \mathbb{R}^n$. For a concave function $\phi$, $\partial\phi(u)$ denotes the set of supergradients of $\phi$ at $u$, i.e., the set of all $y \in \mathbb{R}^n$ such that $\forall u_0 \in \mathbb{R}^n : \phi(u_0) \le \langle y, u_0 \rangle - u \rangle + \phi(u)$. The set $\partial\phi$ is related to the concave conjugate function $\phi^*$ as follows. For an upper semi-continuous concave function $\phi$ we have $\partial\phi(u) = \arg\min_y \langle y, u \rangle - \phi^*(y)$. A differentiable function $\phi$ has a Lipschitz continuous gradient with respect to $\|\cdot\|$ with continuity parameter $1/\sigma > 0$ if for all $\forall u, u_0 \in \mathbb{R}^n$, $\|\partial\phi(u_0) - \partial\phi(u)\|_* \le 1/\sigma \|u - u_0\|$, where $\|\cdot\|_*$ is the dual norm to $\|\cdot\|$. The dual cone $K^*$ of a cone $K \subseteq \mathbb{R}^n$ is defined as $K^* = \{y \mid \langle y, u \rangle \ge 0 \ \forall u \in K\}$. Two examples of self-dual cones are the positive orthant $\mathbb{R}^n_+$ and the cone of $n \times n$ positive semidefinite matrices $S^n_+$. A proper cone (pointed convex cone with nonempty interior) $K$ induces a partial ordering on $\mathbb{R}^n$ which is denoted by $\preceq_K$ and is defined as $x \preceq_K y \Leftrightarrow y - x \in K$. 1.1

Two primal-dual algorithms

The (Fenchel) dual problem for problem (1) is given by

$$\text{minimize} \quad \sum_{t=1}^{m} \sigma_t(A_t y) - \phi^*(y), \quad \text{(2)}$$

where the optimization variable is $y \in \mathbb{R}^n$, and $\sigma_t$ denotes the support function for the set $F_t$ defined as $\sigma_t(z) = \sup_{x \in F_t} \langle x, z \rangle$. A pair $(x^*, y^*) \in (F_1 \times \cdots \times F_m) \times K^*$ is an optimal primal-dual pair if and only if

$$x^*_t \in \arg\max_x \langle x, A_t^T y^* \rangle, \quad x \in F_t$$

$$\sum_{t=1}^{m} X \ y^* \in \partial\phi(A_t x^*_t), \quad \forall t \in [m].$$

Based on these optimality conditions, we consider two algorithms. Algorithm 1 updates the primal and dual variables sequentially, by maintaining a dual variable $y^*_t$ and using it to assign $x^*_t \in \arg\max_{x \in F_t} \langle x, A_t^T y^*_t \rangle$. The then algorithm updates the dual variable based on the second optimality condition. By the assignment rule, we have $A_t x^*_t \in \partial\sigma_t(\Delta y_t)$, and the dual variable update can be viewed as $y^*_{t+1} \in \arg\min_y \langle \sum_{s=1}^{t} A_s x^*_s, y \rangle - \phi^*(y)$. Therefore, the dual update is the same as the update in dual averaging [18] or Follow The Regularized Leader (FTRL) [20, 19, 1] algorithm with regularization $\phi^*(y)$. Algorithm 1 Sequential Update Initialize $y^*_1 \in \partial\phi(0)$ for $t \in 1$ to $m$

do Receive $A_t$ , $F_t$ x ?t ? argmaxx?Ft hx, ATt y?t i Pt y?t+1 ? ??( s=1 As x ?s ) end for Algorithm 2 updates the primal and dual variables simultaneously, ensuring that t X x ?t ? argmaxhx, ATt y?t i, y?t ? ??( As x ?s ). x?Ft

s=1

This algorithm is inherently more complicated than algorithm 1, since finding x ?t involves solving a saddle-point problem. This can be solved by a first order method like mirror descent algorithm for saddle point problems. In contrast, the primal and dual updates in algorithm 1 solve two separate maximization and minimization problems 1 . Algorithm 2 Simultaneous Update for t ? 1 to m do Receive $A_t$ , $F_t$ Pt?1 (? yt , x ?t ) ? arg miny maxx?Ft hy, At x + s=1 As x ?s i ? ? ? (y) end for

2

Competitive ratio bounds and examples for ?

In this section, we derive bounds on the competitive ratios of Algorithms 1 and 2 by bounding their respective duality gaps. We begin by stating a sufficient condition on ? that leads to non-trivial competitive ratios, and we assume this condition holds in the rest of the paper. Roughly, one can interpret this assumption as having ?diminishing returns? with respect to the ordering induced by a cone. Examples of functions that satisfy this assumption will appear later in this section. Assumption 1 Whenever u ?K v, there exists y ? ??(u) that satisfies y ?K ? z for all z ? ??(v). When ? is differentiable, assumption 1 simplifies to u ?K v ? ??(u) ?K ? ??(v). That is, the gradient, as a map from Rn (equipped with ?K ) to Rn (equipped with ?K ? ), is order-reversing.When ? is twice differentiable, assumption 1 is equivalent to hw, ?2 ?(u)vi ? 0, for all u, v, w ? K. For example, this is equivalent to Hessian being element-wise non-positive when K = Rn+ . Pm Let define y?m+1 to be the minimum element in ??( t=1 At x ?t ) with respect ordering ?K ? (such Pto m an element ?t ) and Psim = Pm exists in the superdifferential by Assumption (1)). Let Pseq = ? ( t=1 At x ? ( t=1 At x ?t ) denote the primal objective values for the primal solution produced by the algorithms 1 Also if the original problem is a convex relaxation of an integer program, meaning that each Ft = convFt where Ft ? Zl , then x ?t can always be chosen to be integral while integrality may not hold for the solution of the second algorithm.

3

Pm Pm 1 and 2, and Dseq = t=1 ?t (ATt y?t ) ? ? ? (? ym+1 ) and Dsim = t=1 ?t (ATt y?t ) ? ? ? (? ym+1 ) denote the corresponding dual objective values. The next lemma provides a lower bound on the duality gaps of both algorithms. Lemma 1 The duality gaps for the two algorithms can be lower bounded as Psim ? Dsim ? ? ? (? ym+1 ) + ?(0),

Pseq ? Dseq ? ? ? (? ym+1 ) + ?(0) +

m X hAt x ?t , y?t+1 ? y?t i t=1

Furthermore, if ? has a Lipschitz continuous gradient with parameter 1/? with respect to k?k, Pm 2 1 Pseq ? Dseq ? ? ? (? ym+1 ) + ?(0) ? 2? ?t k . (3) t=1 kAt x Note that right hand side of (3) is exactly the regret bound of the FTRL algorithm (with a negative sign) [19]. The proof is given in the appendix. To simplify the notation in the rest of the paper, we assume ?(0) =

0 by replacing ?(u) with ?(u) ? ?(0). To quantify the competitive ratio of the algorithms, we define ?? as ?? = sup {c — ? ? (y) ? c?(u), y ? ??(u), u ? K},

(4)

Since ? ? (y) + ?(u) = hy, ui for all y ? ??(u), ?? is equivalent to ?? = sup{c — hy, ui ? (c + 1)?(u), y ? ??(u) u ? K}.

(5)

Note that ?1 ? ?? ? 0, since for any u ? K and y ? ??(u), by concavity of ? and the fact that y ? K ? , we have 0 ? hy, ui ? ?(u) ? ?(0). If ? is a linear function then ?? = 0, while if 0 ? ??(u) for some u ? K, then ?? = ?1. The next theorem provides lower bounds on the competitive ratio of the two algorithms. Theorem 1 If Assumption 1 holds, we have Psim ?

1 D? , 1 ? ??

Pseq ?

m X 1 hAt x ?t , y?t+1 ? y?t i) (D? + 1 ? ?? t=1

where D? is the dual optimal objective. If ? has a Lipschitz continuous gradient with parameter 1/? with respect to k?k, Pm 2 1 1 (D? ? 2? ?t k ). Pseq ? 1?? (6) t=1 kAt x ? Pt Pm ?s ?K s=1 As x ?s for all Proof: Consider the simultaneous update algorithm. We have s=1 As x Pt t, since As Fs ?PK for all s. Since y?t ? ??( s=1 As x ?s ) and y?m+1 was picked to be the minimum m element in ??( s=1 As x ?s ) with respect to ?K ? , by Assumption 1, we have y?t ?K ? y?m+1 . Since At x ? K for all x ? Ft , we get hAt x, y?t i ? hAt x, y?m+1 i; therefore, ?t (ATt y?t ) ? ?t (ATt y?m ). Thus Dsim =

m X

?t (ATt y?t ) ? ? ? (? ym ) ?

m X

t=1

?t (ATt y?m+1 ) ? ? ? (? ym ) ? D? .

t=1

Now Lemma 1 and definition of ?? give the desired result. The proof for Algorithm 1 follows similar steps.

We now consider examples of ? that satisfy Assumption 1 and derive lower bound on ?? for those examples. Examples on positive orthant. Let K = Rn+ and note that K ? = K. To simplify the notation we use ? instead of ?Rn+ . Assumption 1 is satisfied for a twice differentiable function if and only if Pn the Hessian is element-wise non-positive over Rn+ . If ? is separable, i.e., ?(u) = i=1 ?i (ui ), Assumption 1 is satisfied since by concavity for each ?i we have ??i (ui ) ? ??i (vi ) when ui ? vi . In the basic adwords problem, for all t, Ft = {x ? Rl+ — 1T x ? 1}, At is a diagonal matrix with non-negative entries, and Pn Pn ?(u) = i=1 ui ? i=1 (ui ? 1)+ , (7) 4

where (?)+ = max{?, 0}. In this problem, ? ? (y) = 1T (y ? 1). Since 0 ? ??(1) we have ?? = ?1 by (5); therefore, the competitive ratio of algorithm 2 is 12 . Let r = maxt,i,j At,i,j , then we have Pm ?t , y?t+1 ? y?t i ? nr. Therefore, the competitive ratio of algorithm 1 goes to 12 as r (bid to t=1 hAt x budget ratio) goes to zero. In adwords with concave returns studied in [8], At is diagonal for all t and ? is separable 2 . Pn For any p ? 1 let Bp be the lp -norm ball. We can rewrite the penalty function ? i=1 (ui ? 1)+ in Pn

the adwords objective using the distance from B? : we have i=1 (ui ? 1)+ = d1 (u, B? ), where d1 (?, C) is the l1 norm distance from set C. For p ? [1, ?) the function ?d1 (u, Bp ) although not separable it satisfies Assumption 1. The proof is given in the supplementary materials. n Examples on the positive semidefinite cone. Let K = S+ and note that K ? = K. Two examples that satisfy Assumption 1 are ?(U ) = log det(U + A0 ), and ?(U ) = trU p with p ? (0, 1). We refer the reader to [10] for examples of online problems that entails log det in the objective function and competitive ratio analysis of the simultanuous algorithm for these problems.

Smoothing of ? for improved competitive ratio

3

The technique of ?smoothing? an (potentially non-smooth) objective function, or equivalently adding a strongly convex regularization term to its conjugate function, has been used in several areas. In convex optimization, a general version of this is due to Nesterov [17], and has led to faster convergence rates of first order methods for non-smooth problems. In this section, we study how replacing ? with a appropriately smoothed function ?S helps improve the performance of the two algorithms discussed in section 1.1, and show that it provides optimal competitive ratio for two of the problems mentioned in section 2, adwords and online LP. We then show how to maximize the competitive ratio of both algorithms for a separable ? and compute the optimal smoothing by solving a convex optimization problem. This allows us to design the most effective smoothing customized for a given ?: we maximize the bound on the competitive ratio over the set of smooth functions.(see subsection 3.2 for details). Let ?S denote an upper semi-continuous concave function (a smoothed version of ?), and suppose ?S satisfies Assumption 1. The algorithms we consider in this section are the same as Algorithms 1 and 2, but with ? replacing ?S . Note that the competitive ratio is computed with respect to the original problem, that is the offline primal and dual optimal values are still the same P ? and D? as before. Pm Pm From LemmaP 1, we have that Dsim ? ?S ( t=1 At x ?t )?? ? (? ym+1 ) and Dseq ? ?S ( t=1 At x ?t )? m ? ? (? ym+1 ) ? t=1 hAt x ?t , y?t+1 ? y?t i. To simplify the notation, assume ?S (0) = 0 as before. Define ??,?S = sup{c —? ? (y) ? ?S (u) + (c ? 1)?(u), y ? ??S (u), u ? K}. Then the conclusion of Theorem 1 for Algorithms 1 and 2 applied to the smoothed function holds with ?? replaced by ??,?S . 3.1

Nesterov Smoothing

We first consider Nesterov smoothing [17], and apply it to examples on non-negative orthant. Given a proper upper semi-continuous concave function ? : Rn 7? R ? {??}, let ?S = (? ? + ?? )? . Note that ?S is the supremal convolution of ? and ?. If ? and ? are separable, then ?S satisfies Assumption 1 for K = Rn+ . Here we provide example of Nesterov smoothing for functions on non-negative orthant. Adwords: The optimal competitive ratio for the Adwords problem is 1 ? e?1 . This is achieved by Pm e smoothing ? with ?? (y) = i=1 (yi ? e?1 ) log(e ? (e ? 1)yi ) ? 2yi , which gives ( eui ?exp (ui )+1 ui ? [0, 1] e?1 ?S,i (ui ) ? ?S,i (0) = 1 ui ¿ 1, e?1 2

Note that in this case one can remove the assumption that ??i ? R+ since

if $y_{t,i} = 0$ for some $t$ and $i$, then $x_{s,i} = 0$ for all $s \geq t$.

5

3.2

## Computing optimal smoothing for separable functions on $\mathbb{R}^n_+$

We now tackle the problem of finding the optimal smoothing for separable functions on the positive orthant, which as we show in an example at the end of this section is not necessarily given by Nesterov smoothing. Given a separable monotone $\Phi(u) = \sum_{i=1}^n \phi_i(u_i)$ and $\Phi_S(u) = \sum_{i=1}^n \phi_{S,i}(u_i)$ on $\mathbb{R}^n_+$ we have that $\Delta_{\Phi,\Phi_S} \geq \min_i \Delta_{\phi_i,\phi_{S,i}}$. To simplify the notation, drop the index $i$ and consider $\phi : \mathbb{R}_+ \to \mathbb{R}$. We formulate the problem of finding $\phi_S$ to maximize $\Delta_{\phi,\phi_S}$ as an optimization problem. In section 4 we discuss the relation between this optimization method and the optimal algorithm presented in [8]. We set $\phi_S(u) = \int_{\mathbb{R}_u} y(s)ds$ with $y$ a continuous function ($y \in C[0, \infty)$), and state the infinite dimensional convex optimization problem with $y$ as a variable, minimize $\theta \int_{\mathbb{R}_u}$ (8) subject to $y(s)ds \geq \psi \; \phi(y(u)) \geq \psi\phi(u), \forall u \in [0, \infty) \; 0 \leq y \leq C[0, \infty)$, where $\psi = 1 - \Delta_{\phi,\phi_S}$ (theorem 1 describes the dependence of the competitive ratios on this parameter). Note that we have not imposed any condition on $y$ to be non-increasing (i.e., the corresponding $\phi_S$ to be concave). The next lemma establishes that every feasible solution to the problem (8) can be turned into a non-increasing solution. **Lemma 2** Let $(y, \psi)$ be a feasible solution for problem (8) and define $y^*(t) = \inf_{s \leq t} y(s)$. Then $(y^*, \psi)$ is also a feasible solution for problem (8). In particular if $(y, \psi)$ is an optimal solution, then so is $(y^*, \psi)$. The proof is given in the supplement.

Re-

visiting the adwords problem, we observe that the optimal solution is given by $y(u) =$

$e^? \exp(u) \; e^? 1$

$+$

,

which is the derivative of the smooth function we derived using Nesterov smoothing in section 3.1. The optimality of this $y$ can be established by providing a dual certificate, a measure $\mu$ corresponding to the inequality constraint, that together with $y$ satisfies the optimality condition. If we set $d\mu = \exp(1 - u)/(e - 1) \, du$, the optimality conditions are satisfied with $\psi = (1 - 1/e)^{-1}$. Also note that if $\phi$ plateaus (e.g., as in the adwords objective), then one can replace problem (8) with a problem over a finite horizon. **Theorem 2** Suppose $\phi(t) = c$ on $[u_0, \infty)$ ($\phi$ plateaus). Then problem (8) is equivalent to minimize subject to

$\theta \int_{\mathbb{R}_u} 0 \; 0$

$y(s)ds \geq \psi \; \phi(y(u)) \geq \psi\phi(u),$

$y(u_0) = 0,$

$\forall u \in [0, u_0]$

(9)

$0$

$y \in C[0, u_0].$

So for a function ? with a plateau, one can discretize problem (9) to get a finite dimensional problem, minimize

?

subject to

h

Pt

s=1

y[s] ? ? ? (y[t]) ? ??(ht)

?t ? [d]

(10)

y[d] = 0, where h = u /d is the discretization step. Figure 1a shows the optimal smoothing for the piecewise linear function ?(u) = min(.75, u, .5u + .25) by solving problem (10). We point out that the optimal smoothing for this function is not given by Nesterov?s smoothing (even though the optimal smoothing can be derived by Nesterov?s smoothing for a piecewise linear function with only two pieces, like the adwords cost function). Figure 1d shows the difference between the conjugate of the optimal smoothing function and ? ? for the piecewise linear function, which we can see is not concave. We simulated the performance of the simultaneous algorithm on a dataset with n = m, Ft simplex, and At diagonal. We varied m in the range from 1 to 30 and for each m calculated the the smallest competitive ratio achieved by the algorithm over (10m)2 random permutation of A1 , . . . , Am . Figure 1i depicts this quantity vs. m for the optimal smoothing and the Nesterov smoothing. For the Nesterov ? ? ? e ? ? smoothing we used the function ? (y) = (y ? e?1 ) log( e ? ( e ? 1)y) ? 23 y. Pm In cases where a bound umax on t=1 At Ft is known, we can restrict t to [0, umax ] and discretize problem (8) over this interval. However, the conclusion of Lemma 2 does not hold for a finite horizon 0

6

and we need to impose additional linear constraints y[t] ? y[t ? 1] to ensure the monotonicity of y. We find the optimal smoothing for two examples of this kind: ?(u) = log(1 + u) over [0, 100] ? (Figure 1b), and ?(u) = u over [0, 100] (Figure 1c). Figure 1e shows the competitive ratio achieved with the optimal smoothing of ?(u) ? = log(1 + u) over [0, umax ] as a function of umax . Figure 1f depicts this quantity for ?(u) = u. 3.3

Competitive ratio bound for the sequential algorithm

In this section we provide a lower bound on the competitive ratio of the sequential algorithm (Algorithm 1). Then we modify Problem (8) to find a smoothing function that optimizes this competitive ratio bound for the sequential algorithm. Theorem 3 Suppose ?S is differentiable on an open set containing K and satisfies Assumption 1. In addition, suppose there exists c ? K such that At Ft ?K c for all t, then 1 D? , Pseq ? 1 ? ??,?S + ?c,?,?S where ? is given by ?c,?,?S = inf{r — hc, ??S (0) ? ??S (u)i ? r?(u), u ? K} Since ?S satisfies Assumption 1, we have y?t+1 ?K ? y?t . Therefore, we can write: Pm Pm ?t , y?t ? y?t+1 i ? t=1 hc, y?t ? y?t+1 i = hc, y?0 ? y?m+1 i (11) t=1 hAt x Pm Now by combining the duality gap by Lemma 1 with 11, we get Dseq ? ?S ( t=1 At x ?t ) ? Pgiven m ? ? (? ym+1 )+hc, ??S (0) ? ??S ( t=1 At x ?t )i.

The conclusion follows from the definition of $?_{?,?S}$ , $?_{c,?,?S}$ and the fact that Dseq ? D? .

Proof:

Based on the result of the previous theorem we can modify the optimization problem set up in Section 3.2 for separable functions on Rn+ to maximize the lower bound on the competitive ratio of the sequential algorithm. Note that when ? and ?S are separable, we have $?_{c,?,?S}$ ? maxi $?_{ci},_{?i},_{?S}$ i . Therefore, similar to the previous section to simplify the notation we drop the index i and assume ? is a function of a scalar variable. The optimization problem for finding ?S that minimizes $?_{c,?,?S}$ ? $??,?S$ is as follows: minimize subject to

? Ru 0

y(s)ds + c(? 0 (0) ? y(u)) ? ? ? (y(u)) ? ??(u),

?u ? [0, ?)

(12)

y ? C[0, ?).

1 u?1 For adwords, the optimal solution is given by ? = 1?exp(? and y(u) = ? 1 ? exp , 1 1+c c+1 ) +

?1 which gives a competitive ratio of 1 ? exp c+1 . In Figure 1h we have plotted the competitive ratio achieved by solving problem 12 for ?(u) = log det(1 + u) with umax = 100 as a function of c. Figure 1g shows the competitive ratio as a function of c for the piecewise linear function ?(u) = min(.75, u, .5u + .25).

4

Discussion and Related Work

We discuss results and papers from two communities, computer science theory and machine learning, related to this work. Online optimization. In [8], the authors proposed an optimal algorithm for adwords with differentiable concave returns (see examples in section 2). Here, ?optimal? means that they construct an instance of the problem for which competitive ratio bound cannot be improved, hence showing the bound is tight. The algorithm is stated and analyzed for a twice differentiable, separable ?(u). The assignment rule for primal variables in their proposed algorithm is explained as a continuous process. A closer look reveals that this algorithm falls in the framework of algorithm 2, with the only difference being that at each step, (? xt , y?t ) are chosen such that x ?t ? argmaxhx, ATt y?t i ?i ? [n] : y?t,i = ??i (vi (ui )), 7

Pt ui = ( t=1 As x ?s )i ,

0.8

0.6

5

25

4

20

3

15

2

10

0.4

0.2

?S ?

1

?S ?

0

0

0 0

0.5

1

1.5

2

0

20

40

60

u

0

100

40

0.9

0.2 0.1

0.4

0.6

0.8

1

0.85 0.8

0.9 0.85 0.8 0.75

0

1

500

0.7

1000

0

200

400

umax

y

0.4

1000

1 Optimal smoothing Nesterov smoothing

0.95

competitive ratio

Competitive ratio

0.5

800

(f)

0.8

0.6

600

umax

(e)

(d) 0.7

100

0.95

0.7

-0.1

80

(c)

0.75

0

60

u

comp. ratio

0.3

0.2

20

(b) 0.95

comp. ratio

?S? (y) ? ?? (y)

(a)

Competitive ratio

80

u

0.4

0

?S ?

5

0.6

0.4

0.2

0.9 0.85 0.8 0.75

0.3

0 0

0.2

0.4

0.6

0.8

1

0

20

40

60

c
c
(g)
(h)
80
100
0.7 0
10
20
30
m
(i)

Figure 1: Optimal?smoothing for ?(u) = min(.75, u, .5u+.25) (a), ?(u) = log(1+u) over [0, 100] (b), and ?(u) = u over [0, 100] (c). The competitive ratio ? achieved by the optimal smoothing as a function of umax for ?(u) = log(1 + u) (e) and ?(u) = u (f). ?S? ? ? ? for the piecewise linear function (d). The competitive ratio achieved by the optimal smoothing for the sequential algorithm as a function of c for ?(u) = min(.75, u, .5u + .25) (g) and ?(u) = log(1 + u) with umax = 100 (h). i, Competitive ratio of the simultaneous algorithm for ?(u) = min(.75, u, .5u + .25) as a function of m with optimal smoothing and Nesterov smoothing (see text).

where $v_i : R+ 7? R+$ is an increasing differentiable function given as a solution of a nonlinear differential equation that involves $?_i$ and may not necessarily have a closed form. The competitive ratio is also given based on the differential equation. They prove that this gives the optimal competitive ratio for the instances where $?_1 = ?_2 = . . . = ?_m$ . Note that this is equivalent of setting $?_{S,i} (u_i ) = ?(v_i (u_i )))$. Since $v_i$ is nondecreasing $?_{S,i}$ is a concave function. On the other hand, given a concave function $?_{S,i} (R+ ) ? ?_i (R+ )$, we can set $v_i : R+ 7? R+$ as $v_i (u) = \inf\{z — ?_i (z) ? ?_{S,i} (u)\}$. Our formulation in section 3.2 provides a constructive way of finding the optimal smoothing. It also applies to non-smooth ?. Online learning. As mentioned before, the dual update in Algorithm 1 is the same as in Follow-theRegularized-Leader (FTRL) algorithm with ?? ? as the regularization. This primal dual perspective has been used in [20] for design and analysis of online learning algorithms. In the online learning literature, the goal is to derive a bound on regret that optimally depends on the horizon, m. The goal in the current paper is to provide competitive ratio for the algorithm that depends on the function ?. Regret provides a bound on the duality gap, and in order to get a competitive ratio the regularization function should be crafted based on ?. A general choice of regularization which yields an optimal regret bound in terms of m is not enough for a competitive ratio argument, therefore existing results in online learning do not address our aim.
8

# 2 References

[1] Jacob Abernethy, Elad Hazan, and Alexander Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In COLT, pages 263?274, 2008. [2] Shipra Agrawal and Nikhil R Devanur. Fast algorithms for online stochastic convex programming. arXiv preprint arXiv:1410.7596, 2014. [3] Yossi Azar, Ilan Reuven Cohen, and Debmalya Panigrahi. Online covering with convex objectives and applications. arXiv preprint arXiv:1412.3507, 2014. [4] Niv Buchbinder, Shahar Chen, Anupam Gupta, Viswanath Nagarajan, et al. Online packing and covering framework with convex objectives. arXiv preprint arXiv:1412.8347, 2014. [5] Niv Buchbinder, Kamal Jain, and Joseph Seffi Naor. Online primal-dual algorithms for maximizing ad-auctions revenue. In Algorithms?ESA 2007, pages 253?264. Springer, 2007. [6] Niv Buchbinder and Joseph Naor. Online primal-dual algorithms for covering and packing. Mathematics of Operations Research, 34(2):270?286, 2009. [7] TH Chan, Zhiyi Huang, and Ning Kang. Online convex covering and packing problems. arXiv preprint arXiv:1502.01802, 2015. [8] Nikhil R Devanur and Kamal Jain. Online matching with concave returns. In Proceedings of the forty-fourth annual ACM symposium on Theory of computing, pages 137?144. ACM, 2012. [9] Nikhil R Devanur, Kamal Jain, Balasubramanian Sivan, and Christopher A Wilkens. Near optimal online algorithms and fast approximation algorithms for resource allocation problems. In Proceedings of the 12th ACM conference on Electronic commerce, pages 29?38. ACM, 2011. [10] R. Eghbali, M. Fazel, and M. Mesbahi. Worst Case Competitive Analysis for Online Conic Optimization. In 55th IEEE conference on decision and control (CDC). IEEE, 2016. [11] Reza Eghbali, Jon Swenson, and Maryam Fazel. Exponentiated subgradient algorithm for online optimization under the random permutation model. arXiv preprint arXiv:1410.7171, 2014. [12] Anupam Gupta and Marco Molinaro. How the experts algorithm can help solve lps online. arXiv preprint arXiv:1407.5298, 2014. [13] Bala Kalyanasundaram and Kirk R Pruhs. An optimal deterministic algorithm for online b-matching. Theoretical Computer Science, 233(1):319?325, 2000. [14] Richard M Karp, Umesh V Vazirani, and Vijay V Vazirani. An optimal algorithm for on-line bipartite matching. In Proceedings of the twenty-second annual ACM symposium on Theory of computing, pages 352?358. ACM, 1990. [15] Robert Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms, pages 630?631. Society for Industrial and Applied Mathematics, 2005. [16] Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. Adwords and generalized online matching. Journal of the ACM (JACM), 54(5):22, 2007. [17] Yu Nesterov. Smooth minimization of non-smooth functions. Mathematical programming, 103(1):127?152, 2005. [18] Yurii Nesterov. Primal-dual subgradient methods for convex problems. Mathematical programming, 120(1):221?259, 2009. [19] Shai Shalev-Shwartz and Yoram Singer. Online learning: Theory, algorithms, and applications. 2007. [20] Shai Shalev-Shwartz and Yoram Singer. A primal-dual perspective of online learning algorithms. Machine Learning, 69(2-3):115?142, 2007. 9