A Tale of Two Metrics: Simultaneous Bounds on Competitiveness and Regret

Lachlan Andrew
Swinburne University of Technology

Siddharth Barman
Caltech

Katrina Liggett
Caltech

Minghong Lin
Caltech

Adam Meyerson
Google Inc

Alan Roytman
UCLA

Adam Wierman
Caltech

1. INTRODUCTION

We consider algorithms for "smoothed online convex optimization" (SOCO) problems, which are a hybrid between online convex optimization (OCO) and metrical task system (MTS) problems. Historically, the performance metric for OCO was regret and that for MTS was competitive ratio (CR). There are algorithms with either sublinear regret or constant CR, but no known algorithm achieves both simultaneously. We show that this is a fundamental limitation – no algorithm (deterministic or randomized) can achieve sublinear regret and a constant CR, even when the objective functions are linear and the decision space is one dimensional. However, we present an algorithm that, for the important one dimensional case, provides sublinear regret and a CR that grows arbitrarily slowly.

A SOCO problem is defined as follows. There is a convex decision/action space $F \subseteq (\mathbb{R}^+)^n$ and a sequence of cost functions $\{c^1, c^2, \dots\}$, where each $c^t : F \to \mathbb{R}^+$. At each time t, a learner/algorithm chooses an action vector $x^t \in F$ and the environment chooses a cost function c^t. The algorithm is then evaluated on the cost function and pays a *switching cost* corresponding to the difference between the actions.

The main difference between regret and CR is that the former compares the performance of an algorithm to that of the *static* optimal solution, while the latter compares with the *dynamic* optimal solution. It is desirable for an algorithm to perform well relative to both benchmarks. For example, in machine learning, the former is appropriate if the concept being learned is static, while the latter is appropriate if the concept is dynamic; if it is not known a priori whether the concept is static or dynamic, then it is important to have an algorithm that performs well in both cases.

OCO and MTS were connected in [1], which studies the special case of fixed and constant switching costs. It shows how to translate bounds on regret into bounds on the CR, and vice versa. Later, [2] used a primal-dual approach to develop an algorithm for the "α-unfair competitive ratio," a hybrid of the CR and regret defined below. The algorithm allows a tradeoff, but does not *simultaneously* perform well for regret and CR. There is also work achieving simultaneous guarantees with respect to the static and dynamic optimal solutions in other settings, e.g., [3], and there have been some attempts to use algorithmic approaches from machine learning in the context of MTSs [4, 5].

SIGMETRICS'13, June 17–21, 2013, Pittsburgh, PA, USA.
ACM 978-1-4503-1900-3/13/06.

We consider several measures of cost. All are special cases of the *α-penalized cost with lookahead i*,

$$C_i^\alpha(A) = \mathbb{E}\left[\sum_{t=1}^{T} c^t(x^{t+i}) + \alpha\|x^{t+i} - x^{t+i-1}\|\right].$$

When α is omitted, it is assumed to be 1.

Within the OCO literature, the typical benchmark that is compared against is the optimal offline static action, i.e.,

$$OPT_s := \min_{x \in F} \sum_{t=1}^{T} c^t(x).$$

The ***regret*** of an online learning algorithm is defined as the (additive) difference between its cost and the cost of the optimal static action vector. Specifically, the regret of Algorithm A on instances $\mathfrak{C}$, $R^0(A)$, is less than $\rho(T)$ if for any sequence of cost functions $(c^1, \dots, c^T) \in \mathfrak{C}$,

$$C_0^0(A) - OPT_s \leq \rho(T) \tag{1}$$

In this traditional definition of regret, there is no switching costs or lookahead. A natural generalization is $R_i(A)$ for which $C_0^0(A)$ is replaced by $C_i^1(A)$ in (1).

Within the MTS literature, the typical benchmark that is compared against is the optimal offline (dynamic) solution:

$$OPT_d := \min_{x \in F^T} \sum_{t=1}^{T} c^t(x^t) + \|x^t - x^{t-1}\|.$$

Note that the minimal cost is the same regardless of lookahead since the cost functions are fixed.

The ***competitive ratio*** compares the cost of an algorithm to that of the offline optimal. The cost typically considered is $C_1(A)$, but more generally the competitive ratio with lookahead i, denoted by $CR_i(A)$, is $\rho(T)$ if for any sequence of cost functions $(c^1, \dots, c^T) \in \mathfrak{C}$

$$C_i(A) \leq \rho(T) OPT_d + O(1). \tag{2}$$

The case of $i = 1$ corresponds to the typical CR studied in the MTS literature.

There are a variety of options for bridging the use of OPT_s by regret and of OPT_d by CR. For example, Adaptive-Regret [6] is defined as the maximum regret over any interval, where the "static" optimum can differ for different intervals, and Internal regret [7] compares the online policy against a simple perturbation of that policy. The metric we use is the ***α-unfair competitive ratio*** [1, 2, 8], and we denote it with $CR_i^\alpha(A)$ in the case of i-lookahead. Formally, $CR_i^\alpha(A)$ is defined exactly the same as the competitive ratio except that the benchmark for comparison is

$$OPT_d^\alpha = \min_{x\in F^T} \sum_{t=1}^{T} c^t(x^t) + \alpha\|x^t - x^{t-1}\|,$$

where $\alpha \geq 1$. Specifically, $CR_i^\alpha(A)$ is $\rho(T)$ if (2) holds with OPT_d replaced by OPT_d^α. Note that OPT_d^α transitions between the dynamic optimal (when $\alpha = 1$) and the static optimal (for large enough α).

2. INCOMPATIBILITY

It is natural to seek algorithms that perform well with respect to both regret (i.e., a static benchmark) and CR (i.e., a dynamic benchmark). However, to date no algorithm has achieved this. For example, online gradient descent [9] has a regret of $O(\sqrt{T})$, or even $O(\log T)$ when the cost function has minimal curvature [10], but has infinite CR. Conversely, algorithms for general MTS problems typically have CR $O(n)$ for a decision space of size n (i.e., $O(1)$ with respect to the number of tasks T) but have linear regret, even for the special case of a one dimensional decision space with convex costs [11].

This is due to a fundamental incompatibility:

Theorem 1. *Consider an arbitrary seminorm $\|\cdot\|$ on $\mathbb{R}^n$, constants $\gamma > 0$, $\alpha > 0$ and $i \in \mathbb{N}$.*

There is a $\mathfrak{C}$ containing a single sequence of cost functions such that, for large enough T, for all deterministic and randomized algorithms A,

$$CR_{i+1}^\alpha(A) + \frac{R_i(A)}{T} \geq \gamma, \tag{3}$$

Moreover, for any deterministic or randomized online algorithm A, there is a $\mathfrak{C}$ consisting of two cost functions such that for large enough T,

$$CR_0^\alpha(A) + \frac{R_0(A)}{T} \geq \gamma \tag{4}$$

The incompatibility (3), which applies to the traditional notions of regret (R_0^0) and competitive ratio (CR_1^1) in the OCO and MTS communities, arises since CR_{i+1}^α and R_i require x_t to minimize c_{t-i} and c_{t-i-1} simultaneously.

The proof of (4) uses linear costs $c^t(x) = a(1-x) + b$ for $\mathbb{E}\left[x^t\right] \leq 1/2$ and $c^t(x) = ax + b$ otherwise, on decision space $[0, 1]$, where x^t is the (random) choice of the algorithm at round t. Here, a and b are fixed constants and the expectation is taken over the marginal distribution of x^t conditioned on $c_1, \ldots, c_{t-1}$, averaging out the dependence on the realizations of $x_1, \ldots, x_{t-1}$.

3. TRADEOFF

To circumvent the incompatibility in Theorem 1, we present an algorithm "Random Bias Greedy" (RBG) for one-dimensional decision spaces that is $O(1)$ (α-unfair) competitive and has ϵT regret for arbitrarily small ϵ.

The algorithm takes a norm N as its input:

Algorithm 1 (RANDOM BIAS GREEDY, RBG(N)). *Given a norm N, define $w^0(x) = N(x)$ for all x and $w^t(x) = \min_y\{w^{t-1}(y) + c^t(y) + N(x-y)\}$. Generate a random number $r \in (-1, 1)$. For each time step t, go to the state x^t which minimizes $Y^t(x^t) = w^{t-1}(x^t) + rN(x^t)$.*

Theorem 2. *For a SOCO problem in a one-dimensional normed space $\|\cdot\|$, running RBG(N) with a one-dimensional norm having $N(1) = \gamma\|1\|$ as input (where $\gamma \geq 1$) attains an α-unfair competitive ratio CR_1^α of $(1+\gamma)/\min\{\gamma, \alpha\}$ and a regret R_0 of $O(\max\{T/\gamma, \gamma\})$ with probability 1.*

To prove Theorem 2, let $c(A) := \sum_{t=1}^T c^t(x^{t+1})$, let $s(A) := \sum_{t=1}^T \|x^{t+1} - x^t\|$ and let OPT_N be the dynamic optimum under norm N with $N(1) = \gamma\|1\|$ ($\gamma \geq 1$). Theorem 2 follows from the following lemmas.

Lemma 3. *Consider a one-dimensional SOCO problem with norm $\|\cdot\|$ and an online algorithm A which, when run with norm N, satisfies $c(A(N)) \leq OPT_N + O(1)$ along with $s(A(N)) \leq \beta OPT_N + O(1)$ with $\beta = O(1)$. Fix a norm N such that $N(1) = \gamma\|1\|$ with $\gamma \geq 1$. Then $A(N)$ has α-unfair competitive ratio $CR_1^\alpha(A(N)) = (1+\beta)\max\{\frac{\gamma}{\alpha}, 1\}$ and regret $R_0(A(N)) = O(\max\{\beta T, (1+\beta)\gamma\})$ for the original SOCO problem with norm $\|\cdot\|$.*

Lemma 4. *Given a one-dimensional SOCO problem with norm $\|\cdot\|$,*

$$\mathbb{E}\left[c(RBG(N))\right] \leq OPT_N$$
$$\mathbb{E}\left[s(RBG(N))\right] \leq OPT_N/\gamma.$$

Acknowledgements

This work was supported by NSF grants CCF 0830511, and CNS 0846025, Microsoft Research, the Lee Center for Advanced Networking, and ARC grants FT0991594 and DP130101378.

4. REFERENCES

[1] A. Blum and C. Burch, "On-line learning and the metrical task system problem," *Machine Learning*, vol. 39, pp. 35–58, 2000.

[2] N. Buchbinder, S. Chen, J. Naor, and O. Shamir, "Unified algorithms for online learning and competitive analysis," in *Proc. Conf. on Learning Techniques (COLT)*, 2012.

[3] A. Blum, S. Chawla, and A. Kalai, "Static optimality and dynamic search-optimality in lists and trees," in *Proceedings of SODA*, 2002, pp. 1–8.

[4] A. Blum, C. Burch, and A. Kalai, "Finely-competitive paging," in *Proceedings of FOCS*, 1999, pp. 450–457.

[5] J. Abernethy, P. Bartlett, N. Buchbinder, and I. Stanton, "A regularization approach to metrical task systems," in *Proceedings of ALT*, 2010, pp. 270–284.

[6] E. Hazan and C. Seshadhri, "Efficient learning algorithms for changing environments," in *Proc. International Conference on Machine Learning*. ACM, 2009, pp. 393–400.

[7] A. Blum and Y. Mansour, "From external to internal regret," *Learning Theory*, pp. 1–10, 2005.

[8] A. Blum, H. Karloff, Y. Rabani, and M. Saks, "A decomposition theorem and bounds for randomized server problems," in *Proc. IEEE Symp. Foundations of Computer Science (FOCS)*, 1992.

[9] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. Int. Conf. Machine Learning (ICML)*, T. Fawcett and N. Mishra, Eds. AAAI Press, 2003, pp. 928–936.

[10] E. Hazan, A. Agarwal, and S. Kale, "Logarithmic regret algorithms for online convex optimization," *Mach. Learn.*, vol. 69, pp. 169–192, Dec. 2007.

[11] M. Lin, A. Wierman, L. L. H. Andrew, and E. Thereska, "Dynamic right-sizing for power-proportional data centers," in *Proc. IEEE INFOCOM*, 2011.