Theoretical Computer Science ••• (••••) •••–•••

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Scale-free online learning ☆

Francesco Orabona [a,1], Dávid Pál [b,*]

[a] Stony Brook University, Stony Brook, NY 11794, USA
[b] Yahoo Research, 11th Floor, 229 West 43rd Street, New York, NY 10036, USA

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Online algorithms
Optimization
Regret bounds
Online learning

ABSTRACT

We design and analyze algorithms for online linear optimization that have optimal regret and at the same time do not need to know any upper or lower bounds on the norm of the loss vectors. Our algorithms are instances of the Follow the Regularized Leader (FTRL) and Mirror Descent (MD) meta-algorithms. We achieve adaptiveness to the norms of the loss vectors by scale invariance, i.e., our algorithms make exactly the same decisions if the sequence of loss vectors is multiplied by any positive constant. The algorithm based on FTRL works for any decision set, bounded or unbounded. For unbounded decisions sets, this is the first adaptive algorithm for online linear optimization with a non-vacuous regret bound. In contrast, we show lower bounds on scale-free algorithms based on MD on unbounded domains.

1. Introduction

Online Linear Optimization (OLO) is a problem where an algorithm repeatedly chooses a point w_t from a convex decision set K, observes an arbitrary, or even adversarially chosen, loss vector ℓ_t and suffers the loss $\langle \ell_t, w_t \rangle$. The goal of the algorithm is to have a small cumulative loss. The performance of an algorithm is evaluated by the so-called regret, which is the difference of the cumulative losses of the algorithm and of the (hypothetical) strategy that would choose in every round the same best point in hindsight.

OLO is a fundamental problem in machine learning [2–4]. Many learning problems can be directly phrased as OLO, e.g., learning with expert advice [5–8] and online combinatorial optimization [9–11]. Other problems can be reduced to OLO, e.g., online convex optimization [12], [4, Chapter 2], online classification [13,14] and regression [15], [2, Chapters 11 and 12], multi-armed bandits problems [2, Chapter 6], [16,17], and batch and stochastic optimization of convex functions [18,19]. Hence, a result in OLO immediately implies other results in all these domains.

The adversarial choice of the loss vectors received by the algorithm is what makes the OLO problem challenging. In particular, if an OLO algorithm commits to an upper bound on the norm of future loss vectors, its regret can be made arbitrarily large through an adversarial strategy that produces loss vectors with norms that exceed the upper bound.

For this reason, most of the existing OLO algorithms receive as an input—or implicitly assume—an upper bound B on the norm of the loss vectors. The input B is often disguised as the learning rate, the regularization parameter, or the parameter of strong convexity of the regularizer. However, these algorithms have two obvious drawbacks.

☆ A preliminary version of this paper was presented at ALT 2015.

* Corresponding author.
E-mail addresses: francesco@orabona.com (F. Orabona), dpal@yahoo-inc.com (D. Pál).

[1] Work done while at Yahoo Research.

https://doi.org/10.1016/j.tcs.2017.11.021
0304-3975/

Table 1
Selected results for OLO. Best results in each column are in bold.

Algorithm	Decisions set(s)	Regularizer(s)	Scale-free
HEDGE [7]	Probability Simplex	Negative Entropy	No
GIGA [20]	Any Bounded	$\frac{1}{2}\|w\|_2^2$	No
RDA [21]	**Any**	**Any Strongly Convex**	No
FTRL-PROXIMAL [22,23]	Any Bounded	$\frac{1}{2}\|w\|_2^2$+ any convex func. [a]	**Yes**
ADAGRAD MD [24]	Any Bounded	$\frac{1}{2}\|w\|_2^2$+ any convex func.	**Yes**
ADAGRAD FTRL [24]	**Any**	$\frac{1}{2}\|w\|_2^2$+ any convex func.	No
ADAHEDGE [25]	Probability Simplex	Negative Entropy	**Yes**
NAG [26]	$\{u : \max_t \langle \ell_t, u\rangle \le C\}$	$\frac{1}{2}\|w\|_2^2$	N/A[b]
SCALE INVARIANT ALGORITHMS [27]	**Any**	$\frac{1}{2}\|w\|_p^2$+ any convex func. $1 < p \le 2$	N/A[b]
SCALE-FREE MD **[this paper]**	$\sup_{u,v\in K} \mathcal{B}_f(u,v) < \infty$	**Any Strongly Convex**	**Yes**
SOLO FTRL **[this paper]**	**Any**	**Any Strongly Convex**	**Yes**

[a] Even if, in principle the FTRL-Proximal algorithm can be used with any proximal regularizer, to the best of our knowledge a general way to construct proximal regularizers is not known. The only proximal regularizer we are aware is based on the 2-norm.

[b] These algorithms attempt to produce an invariant sequence of predictions $\langle w_t, \ell_t\rangle$, rather than a sequence of invariant w_t.

First, they do not come with any regret guarantee for sequences of loss vectors with norms exceeding B. Second, on sequences of loss vectors with norms bounded by $b \ll B$, these algorithms fail to have an optimal regret guarantee that depends on b rather than on B.

There is a clear practical need to design algorithms that adapt automatically to the norms of the loss vectors. A natural, yet overlooked, design method to achieve this type of adaptivity is by insisting to have a **scale-free** algorithm. That is, with the same parameters, the sequence of decisions of the algorithm does not change if the sequence of loss vectors is multiplied by a positive constant. The most important property of scale-free algorithms is that both their loss and their regret scale linearly with the maximum norm of the loss vector appearing in the sequence.

1.1. Previous results

The majority of the existing algorithms for OLO are based on two generic algorithms: FOLLOW THE REGULARIZER LEADER (FTRL) and MIRROR DESCENT (MD). FTRL dates back to the potential-based forecaster in [2, Chapter 11] and its theory was developed in [28]. The name FOLLOW THE REGULARIZED LEADER comes from [16]. Independently, the same algorithm was proposed in [29] for convex optimization under the name DUAL AVERAGING and rediscovered in [21] for online convex optimization. Time-varying regularizers were analyzed in [24] and the analysis tightened in [27]. MD was originally proposed in [18] and later analyzed in [30] for convex optimization. In the online learning literature it makes its first appearance, with a different name, in [15].

Both FTRL and MD are parametrized by a function called a *regularizer*. Based on different regularizers different algorithms with different properties can be instantiated. A summary of algorithms for OLO is presented in Table 1. All of them are instances of FTRL or MD.

Scale-free versions of MD include ADAGRAD MD [24]. However, the ADAGRAD MD algorithm has a non-trivial regret bounds only when the Bregman divergence associated with the regularizer is bounded. In particular, since a bound on the Bregman divergence implies that the decision set is bounded, the regret bound for ADAGRAD MD is vacuous for unbounded sets. In fact, as we show in Section 4.1, ADAGRAD MD and similar algorithms based on MD incur $\Omega(T)$ regret, in the worst case, if the Bregman divergence is not bounded.

Only one scale-free algorithm based on FTRL was known. It is the ADAHEDGE [25] algorithm for learning with expert advice, where the decision set is bounded. An algorithm based on FTRL that is "almost" scale-free is ADAGRAD FTRL [24]. This algorithm fails to be scale-free due to "off-by-one" issue; see [23] and the discussion in Section 3. Instead, FTRL-PROXIMAL [22,23] solves the off-by-one issue, but it requires proximal regularizers. In general, proximal regularizers do not have a simple form and even the simple 2-norm case requires bounded domains to achieve non-vacuous regret.

For unbounded decision sets no scale-free algorithm with a non-trivial regret bound was known. Unbounded decision sets are practically important (see, e.g., [31]), since learning of large-scale linear models (e.g., logistic regression) is done by gradient methods that can be reduced to OLO with decision set $\mathbb{R}^d$.

1.2. Overview of the results

We design and analyze two scale-free algorithms: SOLO FTRL and SCALE-FREE MD. A third one, ADAFTRL, is presented in the Appendix. SOLO FTRL and ADAFTRL are based on FTRL. ADAFTRL is a generalization of ADAHEDGE [25] to arbitrary strongly convex regularizers. SOLO FTRL can be viewed as the "correct" scale-free version of the diagonal version of ADAGRAD