Approximate Convex Optimization by Online Game Playing

Elad Hazan *
IBM Almaden Research Center
hazan@us.ibm.com

Abstract

This paper describes a general framework for converting online game playing algorithms into constrained convex optimization algorithms. This framework allows us to convert the well-established bounds on regret of online algorithms to prove convergence in the offline setting.

The resulting algorithms are very simple to implement and analyze, and for some scenarios attain a better rate of convergence than previously known.

1 Introduction

In various prediction and machine learning scenarios an online learner needs to solve an intermediate convex optimization problem. Although convex optimization is a well-established field, with standard packages readily available, the numerous advanced techniques are sometimes inefficient for small instances and require non-trivial expertise to operate.

On the other hand, simple heuristics such as coordinate descent are known to perform well for small instances and far easier to implement and use in a larger software system. Indeed, some machine learning papers analyzed these well-performing heuristics, mostly obtaining only asymptotic convergence results as opposed to convergence rates.

The purpose of this paper is to give the reader a basic technique to design such simple heuristics which are also provably converging. Further, the methodology we describe makes use of (mostly very simple) learning algorithms, with which the reader is probably already familiar. The advantages and disadvantages of the different basic approaches, such as coordinate descent, gradient descent with different step sizes, and quasi-Newton updates are easily obtained. Thus the practitioner can use the fittest heuristic to her needs and be assured of a guaranteed rate of convergence.

The main technical point of here is a "black box" reduction from regret-minimizing algorithms to convex optimization algorithms. We prove that *any* online algorithm with sublinear regret can be used to derive a convex optimization algorithm. The only property used in the analysis is the low regret, and this flexibility allows us to design some new algorithms with an improved rate of convergence.

We do not claim to attain the best running-times. Indeed, interior point methods and other techniques are unquestionably the methods of choice for large industrial applications. However, for certain instances arising in practice, in particular those for which a rough approximation is enough, our approach may supply a rigorous and simple methodology with a low entrance fee.

*Part of this research was supported by Sanjeev Arora's NSF grants MSPA-MCS 0528414, CCF 0514993, ITR 0205594

Relation to previous work: Most of the algorithms we derive are similar to previous "Lagrangian relaxation algorithms", such as the well-cited Plotkin, Shmoys and Tardos algorithms [PST91], and to [You95, KL96, Jan06, Zha03].

The relation of convex optimization to online learning was also observed previously, notably by Freund and Schapire [FS99]. In a different setting, Hart and Mas-Collel [HMC00] prove that in a multi-player game in which every player plays according to a "good algorithm", the empirical distribution of strategies converges to the set of correlated equilibria. This type of result is similar in spirit to ours, although their focus was not optimization.

2 Definitions and Notation

Our task is to minimize a convex function subject to a set of convex constraints in a convex set. We assume known bounds on the objective value $OPT \in [-M, M]$, and as is standard, use binary search to convert the problem into a feasibility problem of the following form:

$$\begin{aligned} \text{find } x \text{ such that } c_j(x) \leq 0 \quad & \forall j \in [m] \\ x \in \mathcal{P} & \end{aligned} \tag{1}$$

(for an ε-approximate solution we need to solve $O(\log \frac{M}{\varepsilon})$ such formulations) where the c_j are convex functions which describe the constraints and $\mathcal{P}$ is a convex set, which is usually described by simpler constraints.

The online algorithms we consider fall into the *online convex optimization* (OCO) framework [Zin03], in which there is a fixed convex compact feasible set $\mathcal{P} \subset \mathbb{R}^n$ and an *arbitrary, unknown* sequence of convex cost functions $f_1, f_2, \ldots, f_T : \mathcal{P} \to \mathbb{R}$. The decision making algorithm makes a sequence of decisions, where the t^{th} decision is the selection of a point $x_t \in \mathcal{P}$ and there is a cost of $f_t(x_t)$ on period t. However, x_t is chosen with only the knowledge of the set $\mathcal{P}$, previous points $x_1, \ldots, x_{t-1}$, and the previous functions $f_1, \ldots, f_{t-1}$. The standard performance measure for an online convex optimization algorithm is its *regret*, which is defined as:

$$\text{Regret}(\mathcal{A}, T) \triangleq \sup_{f_1, \cdots, f_T} \left\{ \sum_{t=1}^{T} f_t(x_t) - \min_{x^* \in \mathcal{P}} \sum_{t=1}^{T} f_t(x^*) \right\}$$

We say that an algorithm $\mathcal{A}$ has *sublinear regret* if $\text{Regret}(\mathcal{A}, T) = o(T)$. In what follows we use a procedure OnlineAlg as a black box. We require of it only to be a sublinear regret algorithm for this setting.

Most efficient online convex optimization algorithm use the gradient ∇f_t of the cost functions. Denote an upper bound on the ℓ_2 norm of the gradients by $G \geq \|\nabla f_t(x)\|_2$, for all $\forall x \in \mathcal{P}$.

3 The basic method

We're now ready to describe the reduction from a given online algorithm OnlineAlg to a constrained convex optimization algorithm.

The meta-algorithm below finds in each iteration a constraint violated by more than ε. The online algorithm is then simulated with the violated constraint as a cost function. We henceforth prove that this simple scheme returns an ε-approximate solution to the convex feasibility program, and bound the number of iterations for the algorithm to terminate.

Meta-Algorithm PrimalGameOpt
Let $t \leftarrow 1$. While Regret(OnlineAlg,t) $\geq \varepsilon t$ do

1. If $t = 1$ set $x_1 \in \mathcal{P}$ arbitrarily. Else let $x_t \leftarrow$ OnlineAlg $(f_1, ..., f_{t-1})$.
2. Find a constraint $c_j(x_t) > \varepsilon$ for some $j \in [m]$. If not found return x_t. Else, let $f_t \leftarrow c_j$.
3. $t \leftarrow t + 1$

Theorem 1. *Suppose OnlineAlg is an online convex optimization algorithm with low regret. If a solution to mathematical program* (1) *exists, then meta-algorithm* PrimalGameOpt *returns an ε-approximate solution in $O(\frac{R}{\varepsilon})$ iterations, where $R = R(OnlineAlg, \varepsilon)$ is the smallest number T which satisfies the inequality $Regret(OnlineAlg, T) \leq \varepsilon T$.*

Proof. If at iteration t the algorithm cannot find a constraint for which $c_j(x_t) > \varepsilon$, then

$$\forall j \in [m] \ . \ c_j(x_t) \leq \varepsilon$$

implying that x_t is a ε-approximate solution.

Otherwise, for every iteration $c_t(x_t) > \varepsilon$. Since we assumed that the mathematical program is feasible, there exists an x^* for which $\forall j \ . \ c_j(x^*) \leq 0$, and hence forall t, $f_t(x^*) \leq 0$.

Since the online algorithm guarantees sub-linear regret, for some iteration T the regret will be $R \leq \varepsilon T$. By definition of regret we have for the feasible $x^* \in \mathcal{P}$,

$$\varepsilon < \frac{1}{T}\sum_{t=1}^{T} f_t(x_t) \leq \frac{1}{T}\sum_{t=1}^{T} f_t(x^*) + \frac{R}{T} \leq \frac{R}{T} \leq \varepsilon$$

Which is a contradiction. We conclude that the algorithm must find an ε-approximate solution in the claimed number of iterations. □

With this theorem we can now easily derive several known (and a few new) algorithms and prove convergence rates for them. For example, using Zinkevich's Online Gradient Descent, we get an algorithm which iteratively updates the solution by adding to it a $\frac{1}{\sqrt{t}}$ multiple of the gradient of a violated constraint and, if necessary, projecting back onto the convex set [1]. It is easy to see that this algorithm gives an ε-approximate solution in $\frac{nG}{\varepsilon^2}$ iterations.

Other useful choices for the online algorithm are the Multiplicative Weights online algorithm (see survey [AHK05] following [LW94]), online coordinate descent, the Online Newton Method [HKKA06], and other online algorithms [2].

4 Other reductions

In this section we describe two other reductions from online learning to convex optimization. The reductions are equally simple, and can be used to derive other convex optimization algorithms which may have useful properties.

[1] for those readers unfamiliar with Zinkevich's gradient descent algorithm, see [Zin03]

[2] Specific choices and applications are given in the full version of the paper, available from the author's webpage

4.1 The Dual Method

The meta algorithm DualGameOpt solves in each iteration an unconstrained optimization problem over the domain $\mathcal{P}$ (which is usually a much simpler task than the original constrained optimization problem), whose result is denoted $x_t \in \mathcal{P}$. This point x_t is used to define a payoff function g_t over the set of all distributions over the constraints ($\mathbb{S}_m$ - the m-dimensional simplex). The online algorithm is applied to these payoff functions.

We prove below that after a certain number of iterations (which depends on the regret of the online algorithm and $\frac{1}{\varepsilon}$), the average point $\frac{1}{T}\sum_{t=1}^{T} x_t$ is an ε-approximate solution if the original convex feasibility program was feasible.

Meta-Algorithm DualGameOpt
Let $t \leftarrow 1$. While Regret(OnlineAlg,t) $\geq \varepsilon t$ do

- If $t = 1$ set $p_1 = \frac{1}{n}\vec{1}$. Else let $p_t \leftarrow$ OnlineAlg $(g_1, ..., g_{t-1})$ for functions g_t as defined below.
- Let $x_t \leftarrow \arg\min_{x \in \mathcal{P}} \sum_j p_t(j) c_j(x)$. If $\sum_j p_t(j) c_j(x_t) \leq 0$ then return $FAIL$. Define the linear function over $\mathbb{S}_m$:
$$g_t(p) \triangleq \sum_{j=1}^{m} p(j) \cdot f_j(x_t)$$
- $t \leftarrow t + 1$

Return $\bar{x} \triangleq \frac{1}{T}\sum_{t=1}^{T} x_t$

Theorem 2. *Suppose OnlineAlg is an online convex optimization algorithm with low regret. If a solution to mathematical program* (1) *exists, then meta-algorithm* DualGameOpt *returns an ε-approximate solution in $O(\frac{R}{\varepsilon})$ iterations, where $R = R(OnlineAlg, \varepsilon)$ is the smallest number T which satisfies the inequality Regret$(OnlineAlg, T) \leq \varepsilon T$.*

Proof. If for some iteration t the algorithm returns $FAIL$, then

$$\forall x \in \mathcal{P} \ . \ \sum_j p_t(j) c_j(x) > 0$$

implying that the mathematical program is infeasible.

Else, in every iteration $\sum_j p_t(j) c_j(x_t) \leq 0$. As before, for some iteration T the regret of the online algorithm will be $R \leq \varepsilon T$. By definition of regret we have (note that this time the online player wants to maximize his payoff, regret is defined analogously)

$$\begin{aligned}
\forall p^* \in \mathbb{S}_m \ . \ 0 \quad &\geq \quad \frac{1}{T}\sum_{t=1}^{T}\sum_j p_t(j) c_j(x_t) = \frac{1}{T}\sum_{t=1}^{T} g_t(p_t) \geq \frac{1}{T}\sum_{t=1}^{T} g_t(p^*) - \frac{R}{T} \\
&= \quad \frac{1}{T}\sum_{t=1}^{T}\sum_j p^*(j) c_j(x_t) - \varepsilon \geq \sum_j p^*(j) \frac{1}{T}\sum_{t=1}^{T} c_j(x_t) - \varepsilon
\end{aligned}$$

Changing sides and using the convexity of the functions c_j with we obtain (for $\bar{x} = \frac{1}{T}\sum_{t=1}^{T} x_t$)

$$\forall p^* \in \mathbb{S}_m \ . \ \sum_j p^*(j) c_j(\bar{x}) \leq \varepsilon$$

Which in turn implies that

$$\forall j \in [m] \ . \ c_j(\bar{x}) \leq \varepsilon$$

Hence $\bar{x}$ is a ε-approximate solution.

$\square$

Note that this algorithm requires in each iteration to solve an optimization problem. This is in contrast to the previous meta algorithm which only needed to find a violated constraint. The computational advantage of this meta algorithm is that it can potentially use simpler learning algorithms since the domain of the online algorithm is the simplex.

This dual method reduces to the original PST [PST91] algorithm where the choice of OnlineAlg is the Multiplicative Weights online algorithm (see [AHK05] following [LW94]).

4.2 The Distributed Method

The final reduction we describe is perhaps the most surprising. The only operations carried out in each iteration are those defined by the online algorithms. The algorithm can be seen as simulating a game between two players, each implemented by an online algorithm. The intuition is taken from the work of Hart and Mas-Collel [HMC00], who show that low-regret players converge to a correlated equilibrium.

Meta-Algorithm PrimalDualGameOpt
Let $t \leftarrow 1$. While Regret(OnlineAlg1,t) $\geq \frac{\varepsilon}{2}t$ or Regret(OnlineAlg2,t) $\geq \frac{\varepsilon}{2}t$ do

- If $t = 1$ set $x_1 \in \mathcal{P}$ arbitrarily. Else let $x_t \leftarrow$ OnlineAlg1 $(f_1, ..., f_{t-1})$ for f_t as defined below. Define the linear function over $\mathbb{S}_m$:

$$g_t(p) \triangleq \sum_{j=1}^{m} p(j) f_j(x_t)$$

- If $t = 1$ set $p_1 = \frac{1}{n}\vec{1}$. Else let $p_t \leftarrow$ OnlineAlg2 $(g_1, ..., g_{t-1})$ for g_t as defined above. Define the convex function over $\mathcal{P}$:

$$f_t(x) \triangleq \sum_{j=1}^{m} p_t(j) c_j(x)$$

- $t \leftarrow t + 1$

If $\bar{x} \triangleq \frac{1}{T}\sum_{t=1}^{T} x_t$ is ε-approximate return $\bar{x}$. Else, return $\bar{p} = \frac{1}{T}\sum_{t=1}^{T} p_t$.

Theorem 3. *Suppose OnlineAlg1, OnlineAlg2 are online convex optimization algorithms with low regret. If a solution to mathematical program* (1) *exists, then meta-algorithm* PrimalDualGameOpt *returns an ε-approximate solution in $O(\frac{R}{\varepsilon})$ iterations, where R is the smallest number T which satisfies the inequality Regret(OnlineAlg1, T) + Regret(OnlineAlg2, T) $\leq \varepsilon T$.*

Proof. Define,

$$\forall\ x \in \mathcal{P}\ , p \in \mathbb{S}_m\ .\ \ g(x,p) \triangleq \sum_{j=1}^{m} p_j c_j(x)$$

The function g is smooth over the convex sets S_m and $\mathcal{P}$, concave (linear) with respect to p and convex with respect to x. For such functions, generalizations to the von Neumann minimax theorem imply that the value $\lambda^* = \min_{x\in\mathcal{P}} \max_{p\in\mathbb{S}_m} g(x,p) = \max_{p\in\mathbb{S}_m} \min_{x\in\mathcal{P}} g(x,p)$ is well defined.

Denote R_1, R_2 the regrets attained by both online algorithms respectively. Using the low regret properties of the online algorithms we obtain for any x^*, p^*

$$\forall x^*, p^* \ . \ \sum_{t=1}^{T} g(x_t, p^*) - R_1 \leq \sum_{t=1}^{T} g(x_t, p_t) \leq \sum_{t=1}^{T} g(x^*, p_t) + R_2 \tag{2}$$

Let x^* be such that $\forall p \in \mathbb{S}_m \ . \ g(x^*, p) \leq \lambda^*$. By convexity of $g(x, p)$ with respect to x,

$$\forall p^* \ . \ g(\bar{x}, p^*) \leq \frac{1}{T} \sum_{t=1}^{T} g(x_t, p^*) \leq \frac{1}{T} \sum_{t=1}^{T} g(x^*, p_t) + \frac{R_2 + R_1}{T} \leq \lambda^* + \varepsilon$$

Similarly, let p^* be such that $\forall x \in \mathcal{P} \ . \ g(x, p^*) \geq \lambda^*$. Then by concavity of g with respect to p and equation 2 we have

$$\forall x^* \ . \ g(x^*, \bar{p}) \geq \frac{1}{T} \sum_{t=1}^{T} g(x^*, p_t) \geq \frac{1}{T} \sum_{t=1}^{T} g(x_t, p^*) - \frac{R_2 + R_1}{T} \geq \lambda^* - \varepsilon$$

Hence, if $\lambda^* \leq 0$, then $\bar{x}$ satisfies

$$\forall p^* \ . \ g(\bar{x}, p^*) \leq \varepsilon \ \Rightarrow \ \forall j \in [m] \ . \ c_j(\bar{x}) \leq \varepsilon$$

And hence is a ε-approximate solution. Else,

$$\forall x^* \ . \ g(x^*, \bar{p}) > -\varepsilon$$

And $\bar{p}$ is a dual solution proving that the following mathematical program is infeasible.

$$\begin{aligned} c_j(x) &\leq -\varepsilon \quad \forall j \in [m] \\ x &\in \mathcal{P} \end{aligned}$$

$\square$

5 Conclusions

We've described three simple methods to convert low-regret online learning algorithms into convex optimization algorithms. The analysis is very simple and leaves much room for modifications and specialization to special cases (i.e using randomized or approximate online algorithms).

The new applications as well as detailed descriptions of how earlier works can be derived appear in the full version of this paper.

References

[AHK05] S. Arora, E. Hazan, and S. Kale. The multiplicative weights update method: a meta algorithm and applications. *Manuscript*, 2005.

[FS99] Y. Freund and R. E. Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29:79–103, 1999.

[HKKA06] Elad Hazan, Adam Kalai, Satyen Kale, and Amit Agarwal. Logarithmic regret algorithms for online convex optimization. *to appear in 19'th COLT*, 2006.

[HMC00] Sergiu Hart and Andreu Mas-Colell. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68(5):1127–1150, 2000.

[Jan06] Klaus Jansen. Approximation algorithm for the mixed fractional packing and covering problem. *SIAM J. on Optimization*, 17(2):331–352, 2006.

[KL96] Philip Klein and Hsueh-I. Lu. Efficient approximation algorithms for semidefinite programs arising from MAX CUT and COLORING. In *Proceedings of the twenty-eighth annual ACM Symposium on the Theory of Computing*, pages 338–347, 1996.

[LW94] Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1 February 1994.

[PST91] Serge A. Plotkin, David B. Shmoys, and Tardos Tardos. Fast approximation algorithm for fractional packing and covering problems. In *Proceedings of the 32nd Annual IEEE Symposium on Foundations of Computer Science, FOCS'91 (San Juan, Puerto Rico, October 1-4, 1991)*, pages 495–504, Los Alamitos-Washington-Brussels-Tokyo, 1991. IEEE Computer Society Press.

[You95] Neal E. Young. Randomized rounding without solving the linear program. In *Proceedings of the Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 170–178, San Francisco, California, 22–24 January 1995.

[Zha03] Tong Zhang. Sequential greedy approximation for certain convex optimization problems. *IEEE Transaction on Information Theory*, 49:682–691, 2003.

[Zin03] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the Twentieth International Conference (ICML)*, pages 928–936, 2003.