Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 Keller Hall
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 12-008

Online Quadratically Constrained Convex Optimization with Applications to Risk Adjusted Portfolio Selection

Puja Das and Arindam Banerjee

February 28, 2012

Online Quadratically Constrained Convex Optimization with Applications to Risk Adjusted Portfolio Selection

Puja Das
Dept of Computer Science & Engg
University of Minnesota, Twin Cities
Minneapolis, MN 55455
pdas@cs.umn.edu

Arindam Banerjee
Dept of Computer Science & Engg
University of Minnesota, Twin Cities
Minneapolis, MN 55455
banerjee@cs.umn.edu

Abstract

While online convex optimization has emerged as a powerful large scale optimization approach, much of existing literature assumes a simple way to project onto a given feasible set. The assumption is often not true, and the projection step usually becomes the key computational bottleneck. Motivated by applications in risk-adjusted portfolio selection, in this paper we consider online quadratically constrained convex optimization problems, where the feasible set involves intersections of ellipsoids. We show that regret guarantees for the online problem can be achieved by solving a suitable quadratically constrained quadratic program (QCQP) at each step, and present an efficient algorithm for solving QCQPs based on the alternating directions method. We then specialize the general framework to risk adjusted portfolio selection. Through extensive experiments on two real world stock datasets, our proposed algorithm RAMP is shown to significantly outperform existing approaches at any given risk level and match the performance of the best heuristics which do not accommodate risk constraints.

1 Introduction

Online convex optimization (OCO) and stochastic gradient descent (SGD) have emerged as powerful large scale optimization approaches for learning models from huge amounts of data [32, 21, 6, 7, 2, 3]. Instead of handling the full dataset at once, such approaches update the model incrementally based on one data point at a time. SGD is applicable for batch problems where the method makes multiple passes or epochs over the data. OCO is applicable to online settings, where the convex function to be optimized can itself change over time [32]. Such approaches have strong theoretical performance guarantees and have demonstrated remarkable empirical performance [26, 2].

For constrained optimization problems, which forms the basis for learning most widely used models in data mining, the literature on OCO (and SGD) is not as mature. The literature suggests using projected gradient descent (or Newton) updates, where one takes a gradient step based on the current data point and projects the solution to the feasible set. In recent literature, theoretical guarantees have been established for such updates [32, 21, 23]. In practice, however, performing a projection onto a given feasible set is an excrutiatingly time consuming process. Standard approaches, such as Bregman's algorithm which perform cyclic projections onto constituent convex sets followed by nonlinear corrections [11], are inherently sequential and hence can be rather slow in practice especially if the number of constraints is large.

In this paper, we consider online quadratically constrained convex optimization (QCCO) problems. The feasible set F for such problems is determined by a collection of quadratic and linear constraints. In other words, the feasible set is the intersection of ellipsoids, hyperplanes, and half-spaces. As in online convex optimization, learning proceeds in rounds. In round t, the algorithm picks $\mathbf{x}_t \in F$, then nature reveals a convex function f_t, and the algorithm incurs a cost of $f_t(\mathbf{x}_t)$. We illustrate that for any sequence of convex functions, regret guarantees can be obtained by solving a suitable quadratically constrained quadratic program (QCQP) at each step. We propose an efficient primal-dual algorithm for solving QCQPs based on the alternating directions method [8]. The advantage of the proposed approach that it can handle the constraints in parallel using auxiliary variables, so the number of constraints is not an issue in itself.

A key practical motivation behind considering online QCCO problems is portfolio selection under risk constraints. While considerable work has been done in online portfolio selection over the past two decades [15, 22, 1, 14], almost all of them choose not to model the risk associated with the portfolios. Since risk models play a central role in finance [27, 28], online algorithms which disregard risk have had only limited impact, if any, in practice.

Using the general online QCCO framework, we propose Risk Adjusted Meta Portfolio (RAMP), an online portfolio selection algorithm which attempts to maximize wealth under risk constraints. There is extensive literature in finance dwelling on the balance between risk and return and methods for doing the same [4] chapter 4. Our work is motivated by Markowitz's mean-variance portfolio theory [27] which has had profound influence in economic modeling in finance domain [29]. Despite the model's theoretical success, the instable nature of the posed optimization problem has hindered its practical application. [17] found that none of the existing work based on Markowitz's framework is able to outperform the naive equally weighted portfolio.

We model risk in terms of the variance of the portfolio, which is incorporated as a quadratic constraint in the optimization framework. By posing risk as a constraint as opposed to minimizing it as suggested by Markowitz's framework, we also avoid the empirical instability that is inherent of that framework. Further, instead of constructing the portfolio from scratch, we focus on building a meta-portfolio [16] by suitably combining portfolios recommended by a set of existing algorithms [22, 1, 14]. While such algorithms may not have any control on the risk of the portfolios suggested, the meta-portfolio we infer is gauarnteed to satisfy the pre-specified risk constraints, possibly at the cost of accumulating less wealth.Our experiments on the NYSE and S&P500 dataset show that for a fixed risk, RAMP always outperforms the existing online portfolio selection algorithms [22, 1, 14].

The rest of the paper is organized as follows. Section 2 introduces online QCCO, along with regret analysis and the primal-dual algorithm. Section 3 introduces RAMP as a special case of the online QCCO framework. We present details of the extensive experiments we have performed in Section 4, and conclude in Section 5.

2 Online QCQO

Consider the following general form of a *Quadratically Constrained Convex Optimization* (QCCO) problem:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \frac{1}{2}\mathbf{x}^T P_i x + q_i^T x + r_i \leq 0 \ , \quad i = 1, \dots, m \\ & \mathbf{a}_j^T \mathbf{x} = b_j \ , \quad j = 1, \dots, k \ , \end{aligned} \tag{1}$$

where $P_i \in \mathbb{R}^{n \times n}, i = 1, \dots, m$ are positive definite matrices. In QCCO problems, one minimizes a (strictly) convex function over a feasible set F determined by the intersection of ellipsoids and hyperplanes. Quadratically Constrained Quadratic Programs (QCQPs) form an important special case of QCCOs [19, 9].

We focus on an online setting for QCCOs where the convex objective function keeps changing over time while the feasible set stays fixed. The optimization proceeds in round where in round t, the algorithm has to first pick $\mathbf{x}_t \in F$; then, nature reveals f_t and the value of the objective function $f_t(x_t)$ is determined. The strictly convex function f_t chosen by nature can be arbitrary, even adversarial, as long as it satisfies some minimal regularity conditions, which we discuss shortly. The goal is to design an algorithm which picks the sequence of $\mathbf{x}_t$ such that the cumulative objective function value of the adaptive algorithm is competitive with that of the single best $x \in F$ chosen in hindsight. More precisely, we want a sequence $\mathbf{x}_t$ such that

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) \leq \min_{x \in F} \ \sum_{t=1}^{T} f_t(\mathbf{x}) + o(T) \ . \tag{2}$$

With the regret defined as

$$R(T) = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{x \in F} \sum_{t=1}^{T} f_t(\mathbf{x}) \ . \tag{3}$$

which we simply want to be sublinear [21, 23].

The above formulation can be viewed as a special case of the online convex optimization (OCO) framework [32]. While OCO has been widely studied over the past few years from a theoretical perspective, the main algorithm analyzed in this context is projected gradient descent and its variants [32, 3], which needs to perform a projection to the given convex feasible set at every step. While such a projection may be efficiently doable for simple feasible sets, such as the unit L_1 or L_2 ball [18], the projection will undoubtedly be the most time consuming step for more complex feasible sets, including the one considered in QCCO in (1). Assuming the feasible set F to be an intersection of simpler convex sets, Bregman's algorithm can provide a fairly general purpose approach for solving such projection problems by cyclic projections into individual convex sets. However, two aspects of Bregman's algorithm can become rather restrictive especially when the feasible set is defined as the intersection of a number of nonlinear constraints: (i) For anything other than linear equality constraints, each projection has to be followed by a correction by solving a nonlinear equation which can be time consuming [11] and (ii) The algorithm is inherently sequential so that only one constraint can be worked on at any given time. In particular, while online QCQO problems can in principle be solved by projected gradient methods, one has to perform sequential cyclic projections on ellipsoids in each step [11], with suitable corrections by solving nonlinear equations, leading to a rather inefficient algorithm.

In this paper, we propose an efficirnt algorithm for QCCO based on the alternating directions method of multipliers (ADMMs) [8]. The proposed approach is efficient as it directly addresses the two issues discussed above. First, the ADMM algorithm can work with the nonlinear ellipsoidal constraints in parallel. Further, the only nonlinear aspect of the proposed solution is to essentially find the zero-crossing of a monotonic function in a bounded region, which can be done efficiently using standard methods, such as bisection search or Newton-Raphson [10].

We divide our analysis into two parts. We first show that the update for projected gradient descent method, for which regret guarantees already exist [3, 32], can be viewed as the solution of a QCQP, where the quadratic objective gets suitably determined at every step and the quadratic constraints are the same as the original problem. Then, we present an efficient ADMM algorithm to solve general QCQPs, which can be readily applied to obtain $\mathbf{x}_{t+1}$ at each stage of the online algorithm.

2.1 Regret Analysis with Modified QCQP

From recent work in OCO, the projected gradient descent update for a sequence of strictly convex functions f_t is given by:

$$\mathbf{y}_{t+1} = \mathbf{x}_t - \eta_t \nabla f_t(\mathbf{x}_t) \tag{4}$$

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x} \in F} \|\mathbf{x} - \mathbf{y}_{t+1}\|_2 \ . \tag{5}$$

Assuming that the functions are strictly convex so that $\nabla^2 f_t \succ \delta \mathbb{I}, \forall t$ and the gradients have bounded norms, i.e., $\|\nabla f_t\| \leq G$, then existing OCO analysis [20] yields the following regret bound:

Theorem 1 *The projected gradient descent algorithm with step sizes $\eta_t = \frac{1}{\alpha t}$ achieves the following regret:*

$$R(T) = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in F} \sum_{t=1}^{T} f_t(\mathbf{x}) \leq \frac{G^2}{\alpha}(1 + \log T) \ . \tag{6}$$

The update in (5) can be seen as the solution to a proximal optimization problem based on the first order approximation of the function f_t, i.e.,

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x} \in F} \left\{ \mathbf{x}^T \nabla f_t(\mathbf{x}_t) + \frac{1}{2\eta_t} \|\mathbf{x} - \mathbf{x}_t\|^2 \right\} \ . \tag{7}$$

When the convex set F is determined by quadratic constraints as in (1), the optimization problem in (7) is a QCQP. Next we outline an efficient algorithm for solving general QCQPs which can be used for computing $\mathbf{x}_{t+1}$ in the context of online QCCO.

2.2 ADMM for QCQPs

Following (1) and (7), the optimization problem to be solved at each stage of online QCCO can be posed as a QCQP given by:

$$
\begin{aligned}
\min_{\mathbf{x}} \quad & \frac{1}{2}\mathbf{x}^T P\mathbf{x} + \mathbf{q}^T\mathbf{x} + r \\
\text{s.t.} \quad & \frac{1}{2}\mathbf{x}^T P_i\mathbf{x} + q_i^T\mathbf{x} + r_i \leq 0 \ , \quad i = 1, \ldots, m \\
& a_j^T\mathbf{x} = b_j \ , \quad j = 1, \ldots, k \ .
\end{aligned} \tag{8}
$$

For online QCCO, the parameters in the objective function are given by $P = \frac{1}{\eta_t}\mathbb{I}$, $q = \nabla f_t(x_t) + \mathbf{x}_t$, and $r = \frac{1}{2\eta_t}\|\mathbf{x}_t\|^2$. We present our analysis in this section for general $\{P, \mathbf{q}, r\}$ to illustrate the fact that the proposed ADMM algorithm can be used to solve any QCQPs efficiently.

We start by introducing a set of auxiliary variables—$\mathbf{y}_i$ for each ellipsoidal constraint and $\mathbf{z}_i$ for each linear constraint with the additional constraint set $\mathbf{y}_i = \mathbf{x}$ and $\mathbf{z}_j = \mathbf{x}$. Further, we rewrite the quadratic objective function as a sum of quadratic objectives, one corresponding to each $\mathbf{y}_i$. The modified problem stated below is exactly equivalent to the original problem in (8):

$$
\begin{aligned}
\min_{\mathbf{x},\mathbf{y}_i,\mathbf{z}_j} \quad & \frac{1}{2m}\sum_{i=1}^{m}\left\{\mathbf{y}_i^T P\mathbf{y}_i + \mathbf{q}^T\mathbf{y}_i + r\right\} \\
\text{s.t.} \quad & \frac{1}{2}\mathbf{y}_i^T P_i\mathbf{y}_i + q_i^T\mathbf{y}_i + r_i \leq 0 \ , \quad i = 1, \ldots, m \\
& a_j^T\mathbf{z}_j = b_j \ , \quad j = 1, \ldots, k \\
& \mathbf{x} = \mathbf{y}_i \ , \quad i = 1, \ldots, m \\
& \mathbf{x} = \mathbf{z}_j \ , \quad j = 1, \ldots, k \ .
\end{aligned} \tag{9}
$$

Disregarding both sets of original constraints, the augmented Lagrangian for the problem, based on the equality constraints $\mathbf{y}_i = \mathbf{x}$ and $\mathbf{z}_j = \mathbf{x}$, is given by

$$
\begin{aligned}
L_\beta(\mathbf{x}, \mathbf{y}_i, \mathbf{z}_j, \sigma_i, \gamma_j) = & \frac{1}{2m}\sum_{i=1}^{m}\mathbf{y}_i^T P\mathbf{y}_i + \mathbf{q}^T\mathbf{y}_i + r + \sum_{i=1}^{m}\sigma_i(\mathbf{y}_i - \mathbf{x}) + \sum_{j=1}^{k}\gamma_j(\mathbf{z}_j - \mathbf{x}) \\
& + \frac{\beta}{2}\left\{\sum_{i=1}^{m}\|\mathbf{y}_i - \mathbf{x}\|^2 + \sum_{j=1}^{k}\|\mathbf{z}_j - \mathbf{x}\|^2\right\} \ .
\end{aligned} \tag{10}
$$

The quadratic constraints cannot be directly handled by the ADMM. Hence, we ensure the constraint on each $\mathbf{y}_i$ while performing the corresponding iterative update in the ADMM. Similarly, the individual linear constraints ${a^{(j)}}^T\mathbf{z}_j = b_j$ are satisfied while performing the ADMM update on $\mathbf{z}_j$. In particular, the updates for the primal variables $\{\mathbf{y}_i, \mathbf{z}_j, \mathbf{x}\}$ and the dual variables $\{\sigma_i, \gamma_j\}$ are given by:

$$
\begin{aligned}
\mathbf{y}_i^{(t+1)} &= \underset{\mathbf{y}_i: \frac{1}{2}\mathbf{y}_i^T P_i \mathbf{y}_i + q_i^T \mathbf{y}_i + r_i \le 0}{\operatorname{argmin}} \left\{ \frac{1}{2m} \mathbf{y}_i^T P \mathbf{y}_i + \mathbf{q}^T \mathbf{y}_i + \sigma_i^{(t)} (\mathbf{y}_i - \mathbf{x}^{(t)}) + \frac{\beta}{2} \|\mathbf{y}_i - \mathbf{x}^{(t)}\|^2 \right\}, \quad i = 1, \dots, m && (11) \\
\mathbf{z}_j^{(t+1)} &= \underset{\mathbf{z}_j: a_j^T \mathbf{z}_j = b_j}{\operatorname{argmin}} \left\{ \gamma_j^{(t)} (\mathbf{z}_j - \mathbf{x}^{(t)}) + \frac{\beta}{2} \|\mathbf{z}_j - \mathbf{x}^{(t)}\|^2 \right\}, \quad j = 1, \dots, k && (12) \\
\mathbf{x}^{(t+1)} &= \underset{\mathbf{x}}{\operatorname{argmin}} \left\{ \sum_{i=1}^m \sigma_i^{(t)} (\mathbf{y}_i^{(t+1)} - \mathbf{x}) + \sum_{j=1}^k \gamma_j^{(t)} (\mathbf{z}_j^{(t+1)} - \mathbf{x}) + \frac{\beta}{2} \left\{ \sum_{i=1}^m \|\mathbf{y}_i^{(t+1)} - \mathbf{x}\|^2 + \sum_{j=1}^k \|\mathbf{z}_j^{(t+1)} - \mathbf{x}\|^2 \right\} \right\} && (13) \\
\sigma_i^{(t+1)} &= \sigma_i^{(t)} + \beta (\mathbf{y}_i^{(t+1)} - \mathbf{x}^{(t+1)}) , \quad i = 1, \dots, m && (14) \\
\gamma_j^{(t+1)} &= \gamma_j^{(t)} + \beta (\mathbf{z}_j^{(t+1)} - \mathbf{x}^{(t+1)}) , \quad j = 1, \dots, k . && (15)
\end{aligned}
$$

The update for the dual variables in (14) and (15) are already in closed form. Note that the update for $\mathbf{x}^{(t+1)}$ in (13) can be obtained in closed form:

$$
\mathbf{x}^{(t+1)} = \frac{1}{m+k} \left\{ \sum_{i=1}^m \left(\mathbf{y}_i^{(t+1)} + \frac{\sigma_i^{(t)}}{\beta} \right) + \sum_{j=1}^k \left(\mathbf{z}_j^{(t+1)} + \frac{\gamma_j^{(t)}}{\beta} \right) \right\} . \tag{16}
$$

Further, the update for $\mathbf{z}_j^{(t+1)}$ in (12) can also be obtained in closed form by projection of the unconstrained solution onto the hyperplane constraint: for $j = 1, \dots, k$

$$
\mathbf{z}_j^{(t+1)} = \left(\mathbf{x}^{(t)} - \frac{\gamma_j^{(t)}}{\beta} \right) + \frac{1}{\|\mathbf{a}_j\|^2} \left(b_j - \mathbf{a}_j^T \left(\mathbf{x}^{(t)} - \frac{\gamma_j^{(t)}}{\beta} \right) \right) . \tag{17}
$$

Finally, we focus on the update for $\mathbf{y}_i^{(t+1)}$ in (11) which is a QCQP with only one quadratic constraint. Denoting $\mathbf{y} = \mathbf{y}_i$, the problem is given by

$$
\min_{\mathbf{y}} \ \frac{1}{2} \mathbf{y}^T \Gamma \mathbf{y} + \mathbf{g}^T y + d_0 \quad \text{s.t.} \quad \frac{1}{2} \mathbf{y}^T P_i \mathbf{y} + \mathbf{q}_i^T \mathbf{y} + r_i \le 0 , \tag{18}
$$

where $\Gamma = \frac{1}{m} P + \beta \mathbb{I}$, $\mathbf{g} = \mathbf{q} + \beta \mathbf{x}^{(t)} + \sigma_i^{(t)} \mathbf{e}$ where $\mathbf{e}$ is the all ones vector, and $d = \frac{\beta}{2} \|\mathbf{x}^{(t)}\|^2 - \sigma_i^{(t)} \mathbf{x}^{(t)}$. Let

$$
\mathbf{v} = \frac{1}{\rho_i} (P_i^{1/2} \mathbf{y} + P_i^{-1/2} \mathbf{q}_i) , \tag{19}
$$

where $\rho_i = (2 \mathbf{q}_i^T P_i^{-1} \mathbf{q}_i)^{1/2}$. With this change of variables, the original problem can be written as:

$$
\min_{\mathbf{v}} \ \frac{1}{2} \mathbf{v}^T A \mathbf{v} + \mathbf{h}^T \mathbf{v} + d \quad \text{s.t.} \mathbf{v}^T \mathbf{v} \le 1 , \tag{20}
$$

where $A = \rho_i^2 P_i^{-1/2} P P_i^{-1/2}$, $\mathbf{h} = b_i P_i^{-1/2} \mathbf{g} - P_i^{-1/2} P P_i^{-1} \mathbf{q}_i$, and $d = d_0 - \frac{1}{2} \mathbf{q}_i^T P_i^{-1} P P_{-i}^{-1} \mathbf{q}_i - \mathbf{q}^T P_i^{-1} \mathbf{q}_i$. The Lagrangian for (20) is given by

$$
L(\mathbf{v}, \lambda) = \frac{1}{2} \mathbf{v}^T A \mathbf{v} + \mathbf{h}^T \mathbf{v} + d + \lambda (\mathbf{v}^T \mathbf{v} - 1) , \tag{21}
$$

where $\lambda \ge 0$. Setting gradient w.r.t. $\mathbf{v}$ to zero, we obtain

$$
\mathbf{v} = -(A + \lambda \mathbb{I})^{-1} \mathbf{h} . \tag{22}
$$

From the complimentary slackness condition, we have $\lambda (\mathbf{v}^T \mathbf{v} - 1) = 0 \Rightarrow \lambda = 0$ or $\mathbf{v}^T \mathbf{v} - 1 = 0$. For the latter condition to be true, from (22) we must have

$$
f(\lambda) = \mathbf{h}^T (A + \lambda \mathbb{I})^{-2} \mathbf{h} - 1 = 0 . \tag{23}
$$

Let $\alpha_h, h = 1, \ldots, p$ denote the (positive) eigenvalues of A. Then, the eigenvalues of $(A + \lambda\mathbb{I})$ are $(\alpha_h + \lambda), h = 1, \ldots, p$, which are also positive for $\lambda \geq 0$. If $\phi_h, h = 1, \ldots, p$ denote the eigenvectors of $(A + \lambda\mathbb{I})$, then

$$f(\lambda) = \mathbf{h}^T(A + \lambda\mathbb{I})^{-2}\mathbf{h} - 1 = \sum_{h=1}^{p} \frac{\|\mathbf{h}^T\phi_h\|^2}{(\alpha_h + \lambda)^2} - 1 \ .$$

It is easy to verify that $f(\lambda)$ is a decreasing function of λ for $\lambda \geq 0$, e.g., the gradient is negative. Thus, if $f(0) \geq 0$, then $f(\lambda)$ has a zero-crossing in $\lambda \geq 0$ which can be found efficiently using a root finding method, such as bisection search or Newton-Raphson [10]. If $f(0) < 0$, then $\mathbf{v}^T\mathbf{v} - 1 = 0$ cannot be satisfied implying $\lambda = 0$ from the complimentary slackness condition. Then, λ yields the optimal $\mathbf{v}$ from (22), which from (19) gives the solution to the original QCQP in (18). Thus, (11) can also be solved efficiently since the only computation involved is root finding of a monotonic decreasing function.

We highlight two additional advantages of the ADMM approach considered. The updates for the $\mathbf{y}_i$ corresponding to different quadratic constraints for $i = 1, \ldots, m$ can be done in parallel, since there are no interaction terms. This is in sharp contrast with a projected gradient approach based on Bregman's algorithm which uses cyclic projections where at any point only one constraint can be considered. Further, if the original problem has linear inequality constraints of the form $C_j^T x_j \leq d_j$, they can be handled similar to the equality constraints since the projection to a single half-space constraint is also closed form.

2.3 Online QCCO with Varying Constraints

We consider a generalization of the QCCO problem where the constraints themselves can change over time. In particular, we assume that the quadratic constraints have parameters $\{P_i^t, \mathbf{q}_i^t, r_i^t\}$ which are also chosen by nature and revealed before the algorithm picks $\mathbf{x}_t$. As before, the convex function f_t is revealed by nature after $\mathbf{x}_t$ is chosen. If F_t denotes the constraint set in step t, let $F_{(T)} = \cap_{t=1}^T F_t$. We assume that $F_{(T)}$, the intersection of the quadratic constraints across all time steps, is non-empty. The QCQP to be solved at each step of the online QCCO is similar to (7) with the constraint set being F_{t+1}, i.e.,

$$\mathbf{x}_{t+1} = \underset{x \in F_{t+1}}{\operatorname{argmin}} \left\{ \mathbf{x}^T \nabla f_t(\mathbf{x}_t) + \frac{1}{2\eta_t}\|\mathbf{x} - \mathbf{x}_t\|^2 \right\} \ . \tag{24}$$

The above updates can be shown to have logarithmic regret w.r.t. the best solution $\mathbf{x} \in F_{(T)}$.

Theorem 2 *The projected gradient descent algorithm with step sizes $\eta_t = \frac{1}{\alpha t}$ achieves the following regret:*

$$R(T) = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in F_{(T)}} \sum_{t=1}^{T} f_t(\mathbf{x}) \leq \frac{G^2}{\alpha}(1 + \log T) \ . \tag{25}$$

Proof: The proof follows from a modification of the standard argument for projected gradient descent [32]. Let $\mathbf{x}^* = \operatorname{argmin}_{x \in F_{(T)}} \sum_{t=1}^{T} f_t(\mathbf{x}_t)$. Since f_t is α-strong convex, we have

$$f_t(x^*) \geq f_t(x_t) + \nabla f_t(\mathbf{x}_t)^T(x^* - \mathbf{x}_t) + \frac{\alpha}{2}\|\mathbf{x}^* - \mathbf{x}_t\|^2$$
$$2(f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*)) \leq 2\nabla f_t(\mathbf{x}_t)^T(\mathbf{x}_t - \mathbf{x}^*) - \alpha\|\mathbf{x}^* - \mathbf{x}_t\|^2 \ . \tag{26}$$

Since $\mathbf{x}_{t+1}$ is a projection of $(\mathbf{x}_t - \eta_t\nabla f_t(\mathbf{x}_t))$ onto F_{t+1}, and $\mathbf{x}^* \in F_{(T)} \subseteq F_t$, from the generalized Pythagoras theorem,

$$\begin{aligned} \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 &\leq \|\mathbf{x}_t - \eta_t\nabla f_t(\mathbf{x}_t) - \mathbf{x}^*\|^2 \\ &= \|\mathbf{x}_t - \mathbf{x}^*\|^2 + \eta_t^2\|\nabla f_t(\mathbf{x}_t)\|^2 - 2\eta_t\nabla f_t(\mathbf{x}_t)(\mathbf{x}_t - \mathbf{x}^*) \end{aligned}$$

which implies

$$2\nabla f_t(\mathbf{x}_t)^T(\mathbf{x}_t - \mathbf{x}^*) \leq \frac{1}{\eta_t}\left\{\|\mathbf{x}_t - \mathbf{x}^*\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2\right\} + \eta_t G^2 \ . \tag{27}$$

Plugging (27) in (26), summing over all t, and using the fact that $\eta_t = \frac{1}{\alpha t}$ yields the regret bound. ■

3 Risk Adjusted Meta Portfolio

In 1951, Harry Markowitz ushered in the modern era of portfolio theory suggested that single minded pursuit of high return might not constitute a good strategy [27]. Instead the investors should balance a desire for high returns with a desire for low risk. One way to measure risk is the variability of the portfolio [27]. In the traditional Markowitz portfolio optimization framework, the objective is to find a portfolio which has minimal variance for a given expected return c, i.e., $\min_p \; p^T \Sigma \mathbf{p}$, s.t. $p^T \mu = c, p^T \mathbb{1} = 1$, where Σ is correlation between the assets under consideration, μ is the vector of expected return of individual assets, c is the expected return of the portfolio, and p is the portfolio vector, i.e., a probability distribution over the assets.

We make three important modifications to the Markowitz framework. First, the classical framework and almost all of its variants for risk adjusted portfolio selection assume static data and static portfolios, expected to be applied as a 'buy-and-hold' strategy. Over the past two decades, several ideas on online portfolio selection have been proposed [16, 22, 1, 14], although almost all of them ignore the risk modeling aspect. We consider online risk adjusted portfolio selection, which brings in ideas from both lines of development.

Second, we pose the problem as one of maximizing returns using the risk as constraint. The constraint considers relative risk w.r.t. a given strategy such as the uniform portfolio u. If Σ_t is the (estimated) covariance among returns in time t, for an online portfolio p_t, the risk constraint is given by:

$$p_t^T \Sigma_t p_t \leq \alpha_{\text{risk}} \; u^T \Sigma_t u \; , \tag{28}$$

where $\alpha_{\text{risk}} \in \mathbb{R}_{++}$ is the relative risk factor which determines the allowable risk w.r.t. the uniform portfolio in time t. The advantage of the relative formulation, as opposed to an absolute formulation, is that the natural variability of the market is already taken into account. The above constraint in addition to the fact that p_t has to be a probability distribution sets up the (varying) quadratic constraints for the problem.

Third, instead of developing our own algorithm for portfolio selection based on risk constraints, we combine portfolios recommended by existing algorithms [16, 22, 1, 14, 12], referred to as base portfolios in the sequel. The base algorithms need not have an explicit way to satisfy the risk constraints at each step. In prior work, we have illustrated strong performance of such meta portfolios (MPs) in terms of the overall returns without any risk constraints [16].

On day t let r_t be the vector of price relatives, $\Sigma_t \in \mathbb{R}^{n \times n}$ be the (estimated) covariance matrix of price relatives, $X_{t+1} = [x_{t+1,1} \cdots x_{t+1,k}] \in \mathbb{R}^{n \times k}$ be the matrix of base portfolios from k base algorithms, and w_t be the probability distribution on the algorithm iterates on day t, yielding a meta portfolio $p_t = X_t w_t$. Further, the covariance Σ_t is estimated using a smoothed kernel estimator [31]. The constraint set for time $(t+1)$ is given by $F_{t+1} = \{w \in \mathbb{R}^k | w^T X_{t+1}^T \Sigma_t X_{t+1} w \leq \alpha_{\text{risk}} u^T \Sigma_t u, w^T e = 1, w_i \geq 0\}$. Further, the intersection of all such constraints is denoted as $F_{(T)} = \cap_{t=1}^T F_t$. Note that $F_{(T)}$ is non-empty by design since the uniform portfolio $u \in F_t, \forall t$.

The goal is to select a sequence of weights $w_{t+1} \in F_{t+1}$ over portfolios X_{t+1} from base algorithms such that in terms of the logarithmic wealth accumulated over time, the cumulative regret w.r.t. any $w \in F_{(T)}$ is sublinear, i.e.,

$$\max_{w \in F_{(T)}} \sum_{t=1}^T \log(r_t^T X_t w) - \sum_{t=1}^T \log(r_t^T X_t w_t) \leq o(T) \; . \tag{29}$$

The problem of picking the sequence of w_t can be viewed as a special case of the online QCCO framework in Section 2 with convex function $f_t(w) = -\log(r_t^T X_t w)$. Following (24), the update is given by

$$w_{t+1} = {}_{\mathbf{x} \in F_t} \left\{ -\frac{r_t^T X_t w}{r_t^T X_t w_t} + \frac{1}{2\eta_t} \|w - w_t\|^2 \right\} \; . \tag{30}$$

The risk adjusted w_{t+1} obtained by this method, is the meta portfolio for our Risk Adjusted Meta Portfolio (RAMP).By introducing auxiliary variables for the linear constraints in F_{t+1}, and denoting $\alpha_{\text{risk}} u^T \Sigma_t u = b_{t+1}$, the minimization problem can be equivalently written as:

$$\min_{w,y} \quad -\frac{r_t^T X_t w}{r_t^T X_t w_t} + \frac{1}{2\eta_t}\|w - w_t\|^2 \tag{31}$$
$$\text{s.t. } w^T X_{t+1}^T \Sigma_t X_{t+1} w \leq b_{t+1}, \quad y^T e = 1, y_i \geq 0, \quad w = y \ .$$

The risk adjusted w_{t+1} obtained by this method, is the meta portfolio for our Risk Adjusted Meta Portfolio (RAMP). Ignoring all constraints from F_{t+1} for now, with $\nu = \frac{1}{\beta}\lambda$ where λ is the Lagrangian multiplier for the constraint $w = y$, the augmented Lagrangian is given by

$$L(w, y, u) = -\frac{r_t^T X_t w}{r_t^T X_t w_t} + \frac{1}{2\eta_t}\|w - w_t\|^2 + \frac{\beta}{2}\|w - y + \nu\|^2 \ . \tag{32}$$

Then, the ADMM updates are given by

$$w^{k+1} = \min_{w^T X_{t+1}^T \Sigma_t X_{t+1} w \leq b_t} - r_t^T X_t w / r_t^T X_t w_t + \frac{1}{2\eta_t}\|w - w_t\|_2^2 + \frac{\beta}{2}\|w - y^k + \nu^k\|_2^2 \tag{33}$$

$$y^{k+1} = \min_{y \in \Delta_n} \ \|y - (w^{k+1} + \nu^k)\|_2^2 \tag{34}$$

$$\nu^{k+1} = \nu^k + (w^{k+1} - y^{k+1}) \ . \tag{35}$$

The update for w^{k+1} solves a QCQP with a single quadratic constraint, which was discussed in Section 2. The update for y^{k+1} does an Euclidean projection into the probability simplex, for which there are efficient algorithms [18]. Thus, the updates are efficient, and are run iteratively till convergence. To keep track of progress, we compute the primal and dual residuals respectively given by

$$r^{k+1} = w^{k+1} - y^{k+1} \ , \quad s^{k+1} = \beta(y^{k+1} - y^k). \tag{36}$$

The stopping criteria based on the primal and dual residuals from [8] is adopted for the updates. The stopiing criteria is as follows: $\|r^{k+1}\| \leq \epsilon^{pri}$ and $\|s^{k+1}\| \leq \epsilon^{dual}$, where $\epsilon^{pri} = \sqrt{k}\epsilon^{abs} + \epsilon^{rel}\max\{\|w^{k+1}\|, \|y^{k+1}\|\}$. $\epsilon^{dual} = \sqrt{k}\epsilon^{abs} + \epsilon^{rel}\|\lambda^{k+1}\|$. For our experiments, we used $\epsilon^{pri} = \epsilon^{dual} \simeq 10^{-4}$.

4 Experimental Results

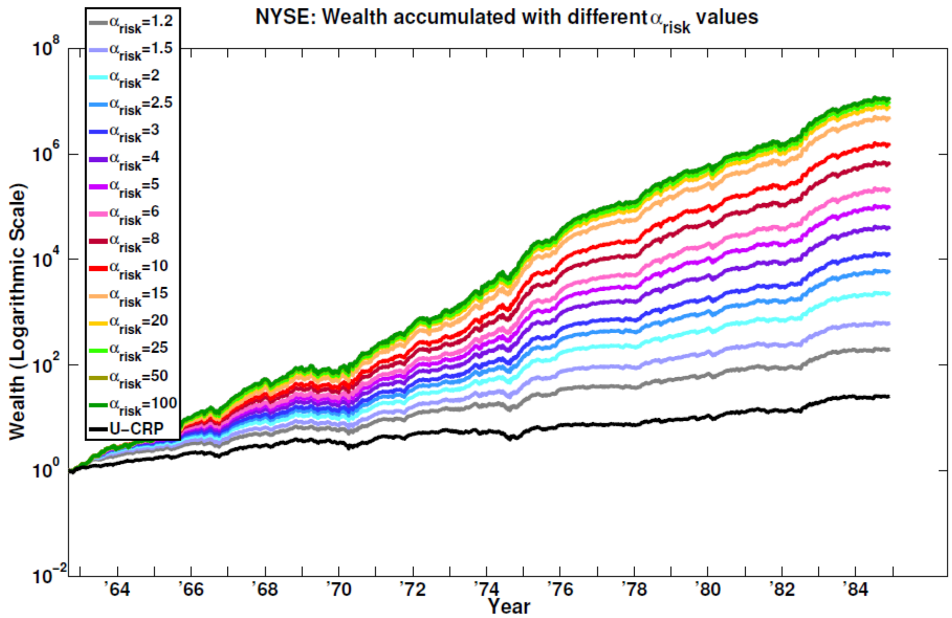
Datasets: The experiments were conducted on two real-world datasets: the New York Stock Exchange dataset (NY-SE) [14] and a Standard & Poor's 500 (S&P 500) dataset. The NYSE dataset consists of 36 stocks with data accumulated over a period of 22 years from July 3, 1962 to Dec 31 1984. The dataset captures the bear market that lasted between January 1973 and December 1974. However, all of the 36 stocks increase in value in the 22-year run.

The S&P500 dataset that we used for our experiments consists of 258 stocks which were present in the S&P500 index in 2011 and were alive since January 1990. This period of 22 years from 1990 to 2011 covers bear and bull markets of recent times.

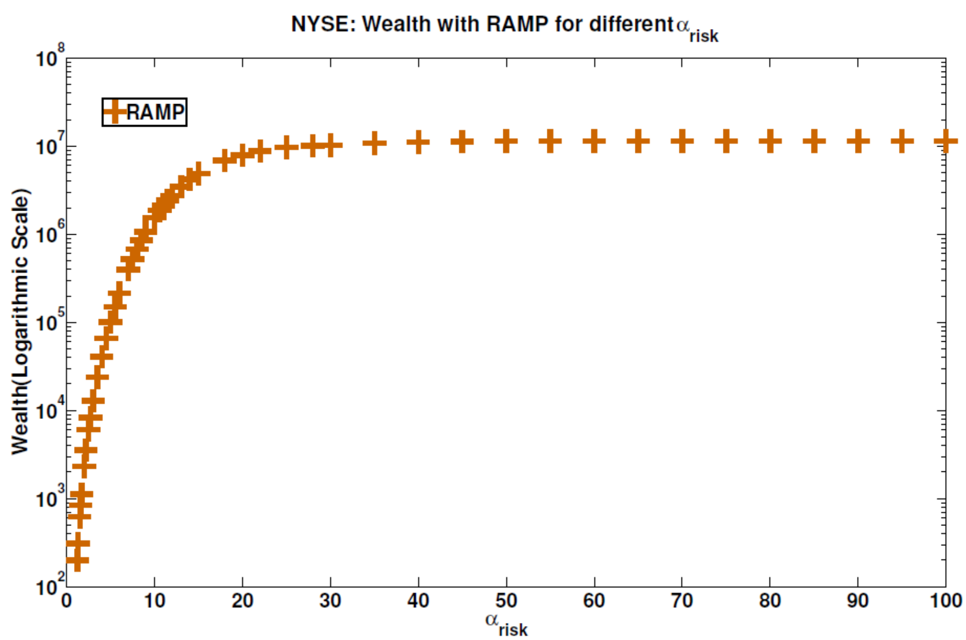
Methodology: We ran a pool of Base Portfolio (BP) selection algorithms and the Meta Portfolio (MP) selection algorithms on the datasets (NYSE and S&P500). RAMP is the only risk adjusted Meta Portfolio. We start by briefly describing the Base Portfolio selection algorithms and the Meta Portfolio selection algorithms before we go on to discuss the results from our experiments.

4.1 Algorithms

Base Portfolio (BP): We have used Universal Potfolios(UP) [14], Exponentiated-Gradient(EG) [22], Online Newton Step method(ONS) [1] as our Base Portfolios. These portfolios are competitive with the best Constant Rebalanced Portfolio (Best CRP) in hindsight. Additionally we have a heuristic Anticor [5] and its variation Anticor$_{BAH}$ [5, 16] in our base pool.



(a) Monetary returns on NYSE.



(b) Wealth VS α_{risk} for NYSE.

Figure 1: Monetary returns of RAMP on the NYSE dataset, for \$1 investment, with different values of α_{risk} used for Risk Minimization. The wealth accumulated grows with increase of α_{risk} (best viewed in color).

Meta Portfolio (MP): The MPs used for our experiments run with identical pool of BPs. MPs can be categorized into 1)risk adjusted and 2)non-risk adjusted. MP_{EG} which uses exponentiated gradient method [24, 25] and MP_{GD} which uses Gradient Descent method [30] are non-risk adjusted. In Online Learning, the objective function includes a penalty term which in the context of meta algorithms is used to keep the weight vectors w_{t+1} and w_t close. While MP_{EG} [16] uses Kullback-Lieblar divergence (relative entropy) as the distance function, MP_{GD}, uses squared euclidean distance. RAMP, our new Risk Adjusted Meta Portfolio algorithm introduced in Section 3 uses squared euclidean distance and behaves like MP_{GD} when we set α_{risk} to be a large number that is when the quadratic risk constraint is relaxed.

Risk: As described in section 4, as opposed to Markowitz's work, RAMP maximizes the wealth for a given α_{risk}. The risk of a portfolio on day t is computed as given in equation (28). We use the Uniform Constant Rebalanced Portfolio [13] (U-CRP also referred to as uniform portfolio) as our basis for risk minimization. Like U-CRP, EWI (Equally Weighted Index) is a popular weighting schemes for S&P500 and like the name suggests equally weighs the constituent S&P500 stocks. However, it should be noted that we could use any other portfolio (eg. Buy-and-Hold) as a standard basis for risk minimization.

We call α_{risk}, the 'permissible risk' with which we run RAMP and we call $\gamma_{risk} = p_t^T \Sigma_t p_t / u^T \Sigma_t u$, the 'actual risk' incurred by the portfolio where u is a uniform constant rebalanced portfolio (U-CRP).

For our experiments we do a warm start, i.e. we start running RAMP, after we have d days of price relatives for constructing the first covariance matrix. This makes our estimated covariance matrix well-conditioned. The offset d depends on the number of stocks n in the dataset and as such should be in the least greater than n. We choose $d = 50$ for the NYSE dataset and $d = 260$ for S&P500.

Parameter choices: Of the BPs, EG runs with a single parameter η_{EG} which was set to a value of 0.5. ONS algorithm has parameters β_{ONS}, η_{ONS} and δ_{ONS} which were set to 1, 0 and $\frac{1}{8}$ respectively for optimal performance [1]. Anticor was run with a window size of 30 days and Anticor_{BAH} with window sizes ranging from 2 to 30. MP_{EG} was run with its single parameter η_{MP} set to 20 to reproduce results observed in [16]. Since, MP_{GD}, is essentially RAMP without the risk constraint, we ran MP_{GD} with different η_t values and found that $\eta_t = 0.35$ does well for NYSE where as $\eta t = 0.21$ works for S&P500. This ηts and $\beta = 0.5$ was then used to run RAMP for the two datasets. Changes in η_t did not change the order of the wealth. It is important to note that as β increases the step size and hence the accuracy for each update for ADMM decreases.

Table 1: APY and mean γ_{risk} of RAMP, MPs and BPs for NYSE and S&P500 dataset.

Dataset		U-CRP	EG	$\text{RAMP}_{1.2}$	ONS	$\text{RAMP}_{7/15}$	Anticor_{30}	Anticor_{BAH}	MP_{EG}	MP_{GD}	RAMP_{100}
NYSE	APY	15.86	15.89	**27.19**	22.99	**62.03**	83.25	125.95	124.25	108.17	**109.35**
	γ_{risk}	1	1.0331	1.0595	2.6762	2.6218	12.6681	5.4344	5.4798	4.8357	4.8012
S&P500	APY	18.19	18.14	**25.79**	43.58	**82.24**	93.99	131.01	125.35	111.01	**113.09**
	γ_{risk}	1	1.0116	1.0818	4.9690	4.6493	29.1620	8.6481	11.9793	9.5793	8.50680

4.2 Results

We ran experiments with the Base Portfolios, Meta Portfolios and RAMP on the two datasets. For RAMP we tried a wide range of α_{risk} values, to observe the nature of the increase of wealth return as the permissible risk increases. We then computed the γ_{risk} on each day for corresponding α_{risk} values. The discussion of the results can be broadly categorized as follows:

No Risk No Gain: One of the essential factors that comes into play for investors in portfolio selection is their degree of risk aversion. Some people are looking for 'safer' investments and will stick to low risk portfolios and hence low returns. Others might be willing to explore riskier portfolios and will reap the benefits of higher rate of returns. Figures 1 and 2 show the wealth accumulated by RAMP for the two datasets with different α_{risk} values. It is evident that RAMP manages to make more money than the U-CRP for both the datasets, with a very slight increase of risk ($\alpha_{risk} = 1.2$).

Figures 1(b) and 2(b) provide supplementary information with the risk(α_{risk}) versus wealth curves that highlight the trade-off between total return and risk. Investing with RAMP on the NYSE dataset and increasing the α_{risk} value from 1 to 25, investors will see a likewise monotonic increase in return. However, for α_{risk} values beyond 25, there is no significant increase in wealth for NYSE. Figure 1 shows that the wealth gain from $\alpha_{risk} = 50$ and $\alpha_{risk} = 100$ are within the same orders of magnitude. Setting α_{risk} to high values essentially means that no heed is paid to the risk incurred. For S&P500 dataset, we see a steep growth of wealth curve between α_{risk} values 1.2 to 30. For $\alpha_{risk} > 30$, the wealth continues to increase but at a much slower rate. Hence ,we see that α_{risk}controls the wealth earned with RAMP and increasing this permissible risk increases the return in wealth.

Permissible and Actual Risk: Figure 3 compares the actual risk (γ_{risk}) observed for each day for different values of permissible risk (α_{risk}) for both the NYSE and S&P500 dataset.

We observe that portfolios obtained with small values of α_{risk} also have small values of γ_{risk}. Figures 3 (b) and (c) show that for α_{risk} set to 100, the γ_{risk} can peak to large values on certain days (going up to 100 for S&P500), while for low α_{risk} values, the γ_{risk} remains very much constrained. Figure 3(a) shows the mean and standards deviation for γ_{risk} for the corresponding α_{risk} values. We notice two things. The mean γ_{risk} grows up to a certain point as α_{risk} increases, and then stabilizes. From the error rates it is quite evident that the greater the permissible risk, the more the γ_{risk} can fluctuate.

Comparison with BPs and MPs: We now discuss how the BPs and MPs compare to RAMP in terms of wealth gained and risk encountered. It is but common knowledge that an investor when faced with two portfolios with similar risk, he will prefer the one with the higher return. Alternatively if he is faced with two portfolios with similar return, will prefer the one with the lower risk. Figure 4 along with Table 1 shows how, when either of these two scenarios occur, RAMP is the natural winner. Figure 4 shows the wealth accumulated by the BPs, MPs and RAMP (with a range of α_{risk} values observed on the top X-axis) and the mean of the corresponding γ_{risk} on the bottom X-axis. Table 1 shows the APY and mean γ_{risk} for the Algorithms. We first look at the performance of the BPs. We observe the following trend. EG and UP both seem to be comparable to the U-CRP in terms of wealth and γ_{risk}. With, almost similar γ_{risk}, RAMP has a larger APY (almost double in the case of NYSE) than EG, UP or U-CRP. ONS has higher risk and also achieves more wealth than EG, UP or U-CRP. But RAMP with γ_{risk} close to that of ONS has almost triple the APY of ONS for NYSE and almost double in case of S&P500 (as seen in Figure 4 and Table 1). Of the BPs, Anticor and Anticor$_{BAH}$ make the most wealth which is in the order of 10^6 and 10^7 respectively for NYSE and S&P500. Anticor has the largest mean γ_{risk} amongst all the portfolio selection algorithms and is not plotted in Figure 4. RAMP with $\alpha_{risk} = 100$ is competitive with Anticor$_{BAH}$ in terms of wealth and has lower risk. MA$_{EG}$ also makes equivalent wealth but has much higher risk. As expected with high values of α_{risk}, we observe that RAMP behaves very similar to MA$_{GD}$. Hence we see that RAMP with equivalent γ_{risk} outdoes the Base online portfolio selection algorithms in wealth. For heuristics like Anticor$_{BAH}$, RAMP is competitive and has lower risk for the same order of wealth return.

RAMP's weight distribution on BPs: Figure 5 shows how RAMP's weight distribution over the 3 BPs (EG, ONS and Anticor) change with α_{risk}. With very low α_{risk}, RAMP runs with EG which we have observed in Figure 4 to have the lowest risk amongst the BPs . As we in crease α_{risk}, to 4 and 6, the weight distribution favors ONS which has medium risk and medium wealth gain amongst all the BPs. As we further relax the risk constraint and set α_{risk} to 95 and 100, RAMP almost concentrates its entire wealth on Anticor, because it makes the most wealth amongst the three BPs shown here. Hence depending on the value of α_{risk} that we set and the independent performance of the BPs in terms of risk , RAMP will adjust its weight on the BPs accordingly. This trend is clearly evident here.

5 Conclusions

We have presented an efficient and scalable algorithm for solving Quadratically Constrained Convex Optimization problems in the Online setting. Using a primal-dual approach based on ADMM, our algorithm overcomes the computational bottleneck that arises in the projection step of Online Convex Optimization when faced with ellipsoidal constraints. We extend our work to portfolio selection, where the existing online algorithms do not have a way of accounting for risk. We adopt Markowitz's mean-variance framework for risk and propose

RAMP, the risk adjusted meta portfolio which combines base portfolios and is adept in satisfying the risk constraint. Through extensive experiments over the NYSE and S&P500 datasets, we observe that RAMP for a given risk level outperforms existing online portfolio selection algorithms. RAMP is also competitive with the best heuristic in its pool of base algorithms and has lower risk. For our future work we plan to look at other existing notions of risk in finance as additional constraints.

Acknowledgements: The research was supported by NSF CAREER award IIS-0953274, and NSF grants IIS-0916750, IIS-0812183, IIS-1029711, and NetSE-1017647.

References

[1] A. Agarwal, E. Hazan, S. Kale, and R. Schapire. Algorithms for portfolio management based on the newton method. *Proceedings of the 23rd International Conference on Machine Learning*, pages 9–16, 2006.

[2] A. Beck and M. Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

[3] D. Bertsekas. *Nonlinear Programming.* Athena Scientific, 1999.

[4] Z. Bodie, A. Kane, and A. Marcus. *Portfolio Performance Evaluation, Investments.* Irwin McGraw-Hill, 1999.

[5] A. Borodin, R. El-Yaniv, and V. Gogan. Can we learn to beat the best stock. *Journal of Artificial Intelligence Research*, 21:579–594, 2004.

[6] L. Bottou. Stochastic gradient learning in neural networks. In *Proceedings of Neuro-Nîmes*, 1991.

[7] L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of the 19th International Conference on Computational Statistics*, pages 177–187, 2010.

[8] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

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

[10] R. Burden and J. D. Faires. *Numerical Analysis.* Thompson, 1985.

[11] Y. Censor and S. Zenios. *Parallel Optimization: Theory, Algorithms and Applications.* Oxford University Press, 1997.

[12] N. Cesa-Bianchi, Y. Freund, D. Haussler, D. P. Helmbold, R. E. Schapire, and M. K. Warmuth. How to use expert advice. *Journal of the ACM*, 44(3):427–485, 1997.

[13] N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games.* Cambridge University Press, 2006.

[14] T. Cover. Universal portfolios. *Mathematical Finance*, 1:1–29, 1991.

[15] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* Wiley-Interscience, 1991.

[16] P. Das and A. Banerjee. Meta optimization and its application to portfolio selection. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2011.

[17] V. DeMiguel, L. Garlappi, F. J. Nogales, and R. Uppal. A generalized approach to portfolio optimization: Improving performance by constraining portfolio norms. *Management Science*, 55(5):798–812, 2009.

[18] J. Duchi, Y. Singer, and T. Chandra. Efficient projections onto the ?1-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, 2008.

[19] D. Goldfarb and G. Iyengar. Robust convex quadratically constrained programs. *Mathematical Programming*, 97:495–515, 2002.

[20] E. Hazan. *Optimization for Machine Learning*, chapter The convex optimization approach to regret minimizatio. MIT Press, 2011.

[21] E. Hazan, A. Agarwal, and S. Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.

[22] D. Helmbold, E. Scahpire, Y. Singer, and M. Warmuth. Online portfolio setection using multiplicative weights. *Mathematical Finance*, 8(4):325–347, 1998.

[23] S. Kakade and S. Shalev-Shwartz. Mind the duality gap: Logarithmic regret algorithms for online optimization. In *Neural Information Processing Systems*, 2008.

[24] J. Kivinen and M. Warmuth. Additive versus exponentiated gradient updates for linear prediction. In *Proceedings of the 27th Annual ACM Symposium on the Theory of Computing*, pages 209–218, 1995.

[25] J. Kivinen and M. Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *Information and Computation*, 132(1):1–64, 1997.

[26] J. Langford, L. Li, and A. Strehl. Vowpal wabbit online learning project. http://hunch.net/?p=309, 2007.

[27] H. Markowitz. Portfolio selection. *Journal of Finance*, 7:77–91, 1952.

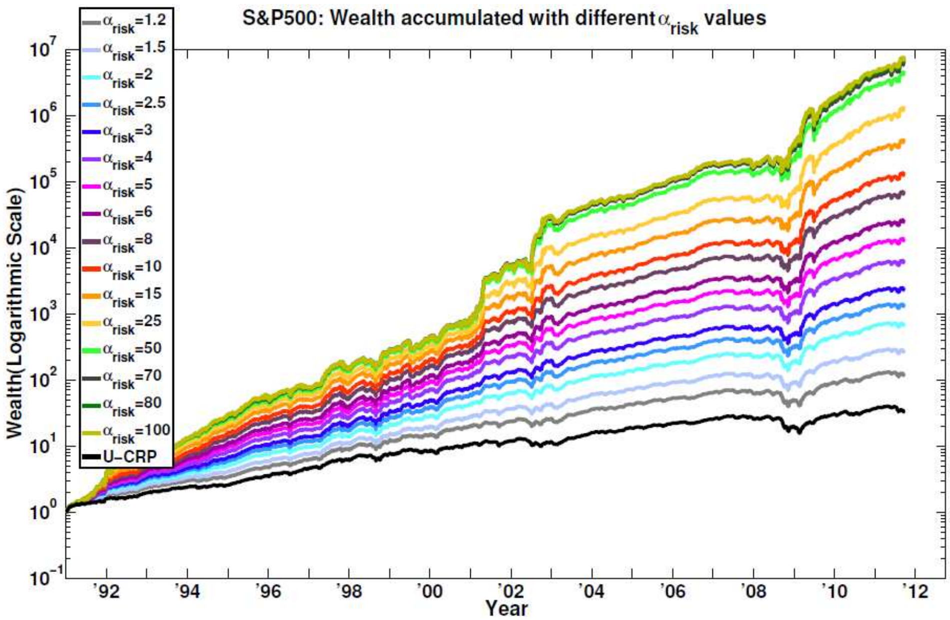
[28] W. Sharpe. Mutual Fund Performance. *The Journal of Business*, 39(1):119–138, 1966.

[29] M. Steinbach. Markowitz revisited: Mean-variance models in financial portfolio analysis. *SIAM Review*, 43(1):31, 2001.

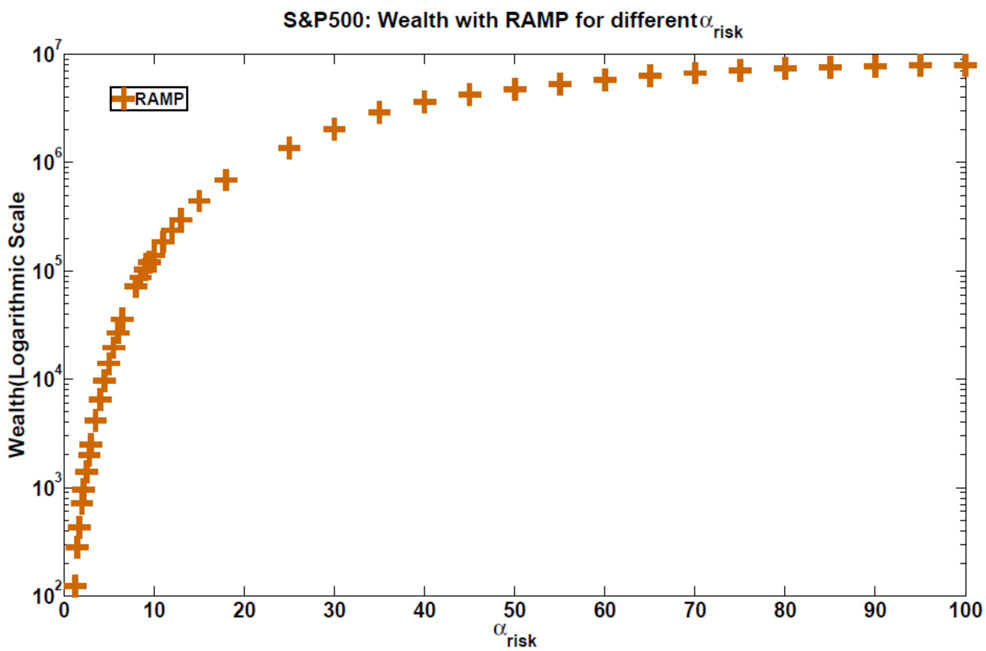
[30] B. Widrow and S. Stearns. *Adaptive signal processing*. Prentice-Hall, Inc., 1985.

[31] S. Zhou, J. D. Lafferty, and L. A. Wasserman. Time varying undirected graphs. *Machine Learning*, 80(2-3):295–319, 2010.

[32] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. *Proceedings of the 20th International Conference on Machine Learning*, 2003.

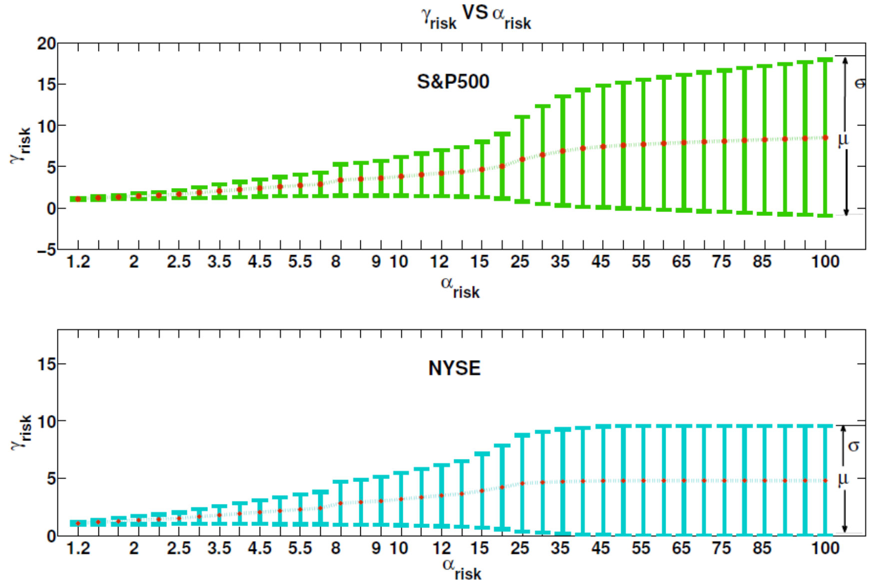


(a) Monetary returns on S&P500.

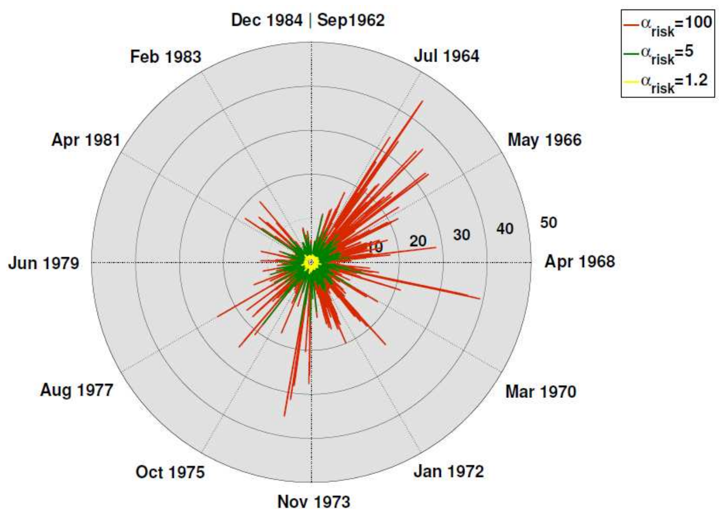


(b) Wealth VS α_{risk} for S&P500:.

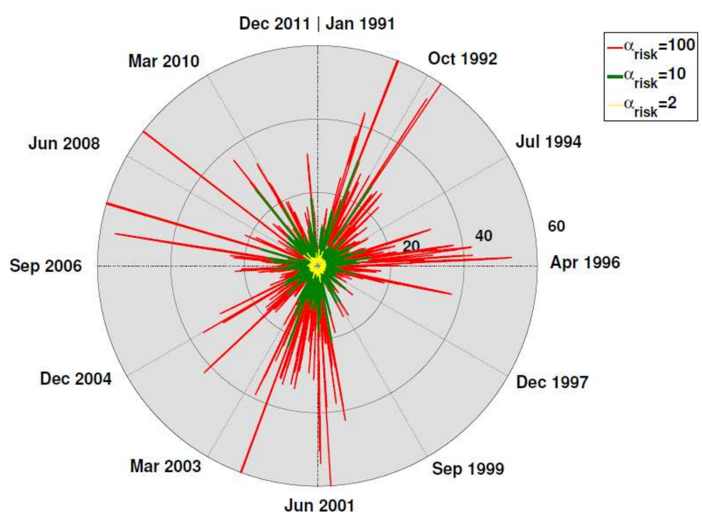
Figure 2: Monetary returns of the RAMP on the S&P500 dataset, for $1 investment, with different values of α_{risk} used for Risk Minimization. The wealth accumulated grows with increase of α_{risk} (best viewed in color).



(a) Mean and standard deviation of γ_{risk} for different α_{risk}.

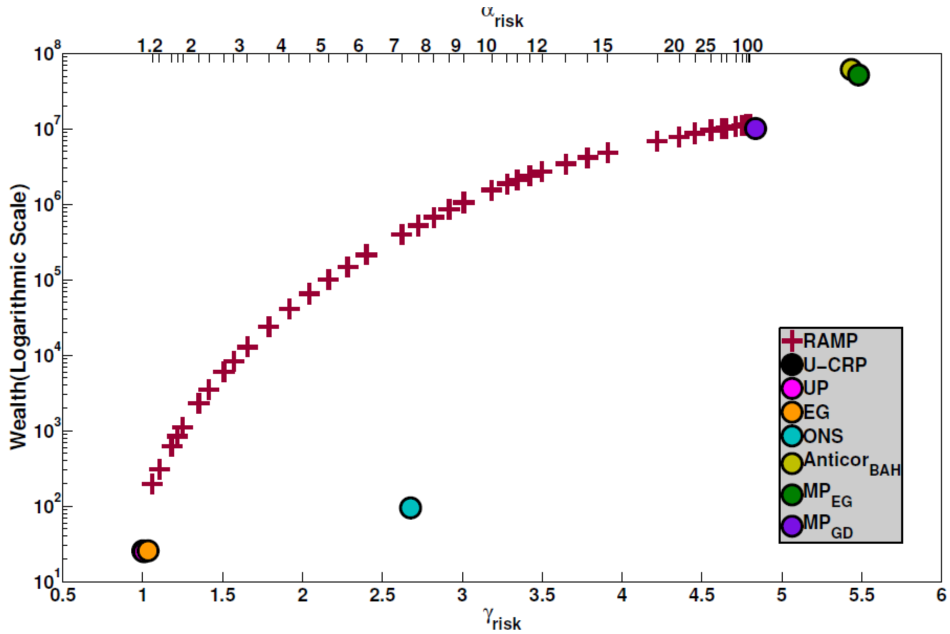


(b) γ_{risk} for NYSE.

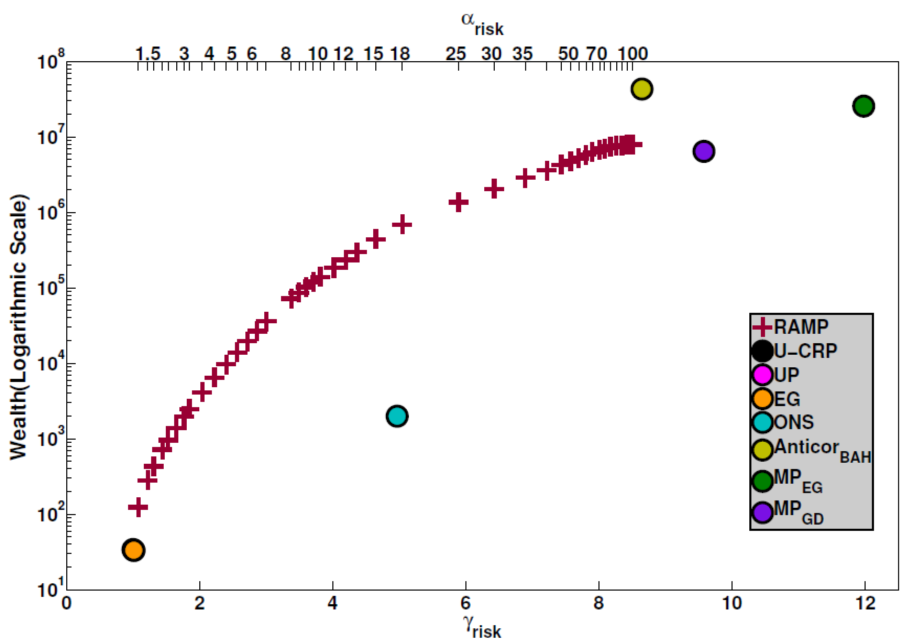


(c) γ_{risk} for S&P500.

Figure 3: RAMP: Comparison of γ_{risk} with their corresponding α_{risk} values(best viewed in color).



(a) Monetary returns VS γ_{risk} and α_{risk} for NYSE.



(b) Monetary returns VS γ_{risk} and α_{risk} S&P500.

Figure 4: For equivalent γ_{risk}, RAMP is either competitive or does better in terms of monetary returns (for $1 investment shown here) with the BPs, MP $_{EG}$ and MP$_{GD}$ (best viewed in color).

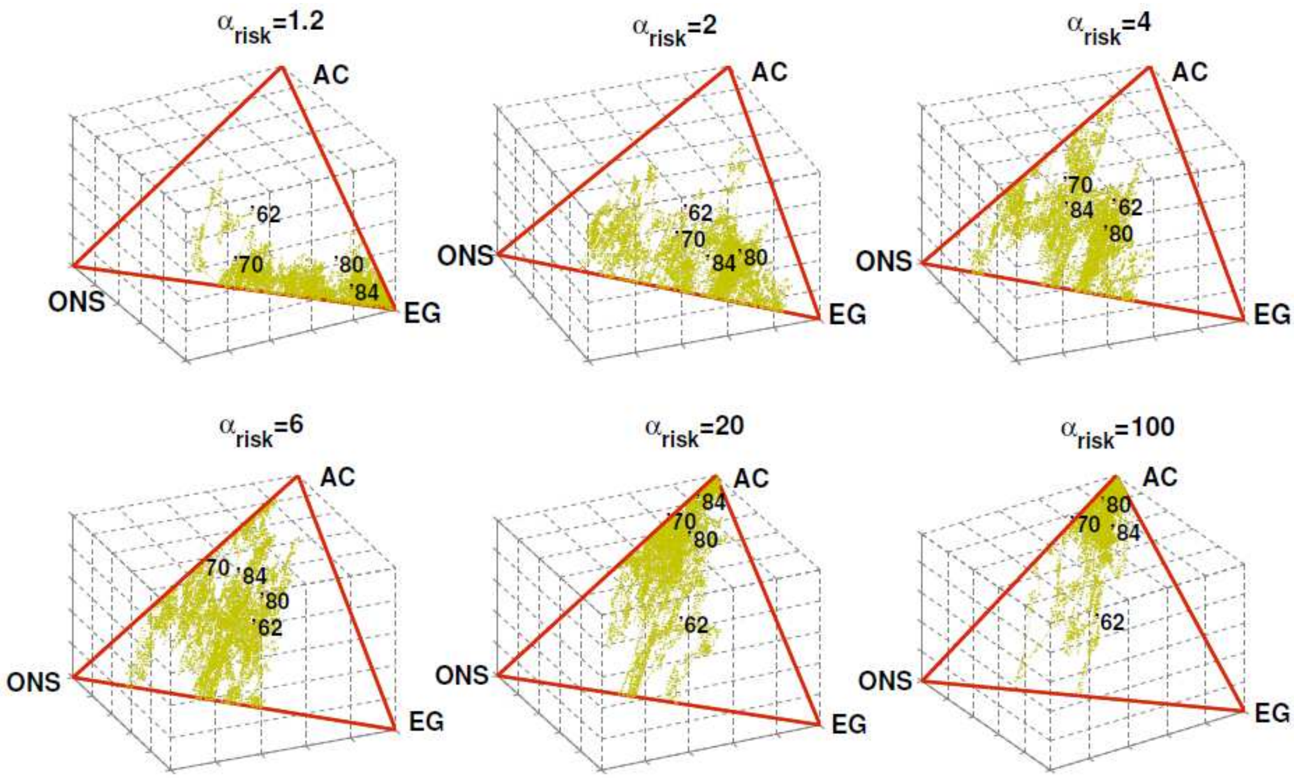


Figure 5: RAMP's weight distribution on the BPs: EG, ONS and Anticor. With low α_{risk} values, most of RAMP's weight is concentrated on EG (has inherent low γ_{risk}) , as α_{risk} increases, the weight shifts first to ONS (moderate wealth and moderate γ_{risk}) and with high values of α_{risk}, weight shifts entirely on to Anticor (greater wealth amongst the BPs with large γ_{risk})(best viewed in color).