AFRL-RQ-WP-TR-2019-0196

OPTIMAL ONLINE DATA-DRIVEN OPTIMIZATION WITH MULTIPLE TIME-VARYING NON-CONVEX OBJECTIVES

Arthur R. Calderbank, Robert Ravier, and Vahid Tarokh

Duke University
Division Office of Research Administration

OCTOBER 2019
Final Report

DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited.

STINFO COPY

AIR FORCE RESEARCH LABORATORY
AEROSPACE SYSTEMS DIRECTORATE
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7542
AIR FORCE MATERIEL COMMAND
UNITED STATES AIR FORCE

NOTICE AND SIGNATURE PAGE

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
October 2019	Final	11 April 2018 – 11 October 2019

4. TITLE AND SUBTITLE
OPTIMAL ONLINE DATA-DRIVEN OPTIMIZATION WITH MULTIPLE TIME-VARYING NON-CONVEX OBJECTIVES

5a. CONTRACT NUMBER FA8650-18-1-7837
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER 61101E

6. AUTHOR(S)
Arthur R. Calderbank, Robert Ravier, and Vahid Tarokh

5d. PROJECT NUMBER 1000 (DARPA)
5e. TASK NUMBER
5f. WORK UNIT NUMBER Q1YQ

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Duke University
Division Office of Research Administration
2200 West Main St., Suite 710
Durham, NC 27705

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory
Aerospace Systems Directorate
Wright-Patterson Air Force Base, OH 45433-7542
Air Force Materiel Command
United States Air Force

DARPA/DSO
675 North Randolph Street
Arlington, VA 22203-2114

10. SPONSORING/MONITORING AGENCY ACRONYM(S)
AFRL/RQVC

11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S)
AFRL-RQ-WP-TR-2019-0196

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited.

13. SUPPLEMENTARY NOTES
This report is the result of contracted fundamental research, which is deemed exempt from Public Affairs Office security and policy review in accordance with Deputy Assistant Secretary of the Air Force (Science, Technology, Engineering) (SAF/AQR) memorandum dated 10 Dec 08 and Air Force Research Laboratory Executive Director (AFRL/CA) policy clarification memorandum dated 16 Jan 09.

This report contains .pdf attachments related to the research effort. Click the paperclip icon in the left vertical toolbar, and double click the referenced attachment.

14. ABSTRACT
This report presents the development of optimization research methods and concepts for time-varying, real-world situations, which include: factoring in more realistic assumptions on the dynamical systems underlying the objective functions and constraints, nonconvexity of objectives, reasonability of currently existing performance measures, time-varying constraints, and situations where the true objective function is unknown. Two scenarios were considered for training of statistical models for optimization: 1.) where changes in the objective function were smooth, and 2.) where the dynamical system of objective functions was adversarial (non-predictable). The problems considered in the body of work span a large number of subfields of optimization and are summarized in the remainder of the report.

15. SUBJECT TERMS
non-convex optimization, online data-driven, adaptive algorithms

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT:	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON (Monitor)
a. REPORT	b. ABSTRACT	c. THIS PAGE			Edwin E. Forster
Unclassified	Unclassified	Unclassified	SAR	37	19b. TELEPHONE NUMBER (Include Area Code) N/A

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Optimization of multiple time-varying objective functions was investigated, with a goal of adapting currently known theoretical results and numerical algorithms to better fit real-world scenarios. A significant body of mathematical frameworks and algorithms designed to handle multiple different scenarios based on as additional structural information as possible were formulated. More specifically, novel research directions that will be of significant interest for the foreseeable future were proposed, multiple new performance measures for optimization in time-varying scenarios were developed, and new algorithms as modifications of previously existing algorithms to handle such scenarios were developed. All numerical methods developed were tested on both synthetic and real data. By the end of the performance period, it was made clear that time-varying optimization remains a rich area with significant theoretical and practical contributions for scientific interests inside and outside the Department of Defense remaining to be made.

1 SUMMARY

The Duke University group for the Defense Advanced Research Projects Agency (DARPA) Lagrange program was tasked with investigating optimization problems in time-varying scenarios in situations where multiple objective functions may be present. Their literature came to a number of conclusions concerning previously developed theoretical and practical methods in the literature: outside of, e.g. [1], there was a surprising lack of work related to situations where time-varying objective functions came from stochastic dependent processes (though both stochastic independent processes [2, 3, 4] and deterministic dependent processes [5, 6, 7] were considered); performance measures previously studied were focused on situations and goals not sufficiently broad enough to encompass all situations of practical interest; the theoretical performance guarantees of the online distributed optimization literature [8] did not match those for online centralized optimization [9]; and, that the data-driven distributionally robust optimization literature had very little work in terms of adaptations to time-varying scenarios (though a model based solution was considered [10]). The goal of this group was to address each of these issues.

In addressing the lack of stochastic dependence in the time-varying optimization literature, the group approached the problem from two standpoints: when the objective functions are over a continuous domain, and when the objective functions are over a discrete domain. In the case that the objective functions were continuous, they investigated the manner in which accurate prediction of future events affected total algorithmic performance, where performance is measured in a quantity known as dynamic regret. They came to the conclusion that potentially significant increases in performance could be made by predicting correctly in contrast to traditional approaches in the literature which do not consider predictable scenarios [9, 11]. Furthermore, the performance guarantees are proved in a method that is readily adaptable to those of other optimization, thus can easily be adapted for further work. Considering specific time-varying scenarios, where objective functions either correspond to conditional expectations or smoothly varying functions results in specific theoretical bounds that are easily interpretable both theoretically and practically. Since the stochastic process governing the objective functions was assumed to be stochastic and dependent, an algorithm was developed using Gibbs posterior inference/expert learning in a manner similar to those in meta-learning, where the algorithm would learn the future objective function before making an optimization step [12, 13].

From the standpoint of continuous objectives over a discrete domain, the team also investigated bandit optimization in the case of stochastically dependent availability, specifically if the available options depended on the immediate past, whereas then previously existing literature only considered the case that the actions available at each point and time were independent of the past [2, 3]. They developed an adaptation of the state-of-the-art algorithm for the dependent case, where the goal was to learn marginals of the Markov chain, and then base algorithmic decisions on those marginals. They chose to learn marginals rather than the entire Markov chain due to memory constraints. They showed via numerical experiment over a short amount of time steps, their algorithm outperformed that of the optimal algorithm for the adversarial case, which is the only one that could be immediately adapted in this scenario.

The team's work on performance measures focused on two areas. In online scenarios, performance measures are usually given by the performance of an online optimization algorithm by considering the cumulative objective value of all points versus some optimal choice. This does not adapt well to

nonconvex problems or problems with constraints. The team proposed a new performance measure known as the Karush-Kuhn-Tucker (KKT) Regret, named after the corresponding optimality conditions on the time-invariant setting [14], that adapts known optimality conditions in the time-invariant setting to the time-varying one. As a sample calculation, we evaluated the performance of a standard adaptation of the well-studied primal dual algorithm for constrained optimization under the general prediction regime mentioned above, and showed that the KKT regret depends on a number of different regularities of the dynamical system governing the objective function.

Then, going back to the stochastic case, they considered a practical problem for time-varying optimization problems: if the true process governing the objective function is not known, what is a good performance measure that reflects the best possible performance based on a number of candidate models and not the true process? This led them to formulate the notion of asymptotically efficient optimization, directly analogous to that due to Shibata in the time series literature [15]. The definition then immediately adapts to a measurable performance measure provided sufficient computational resources. They showed that a minor adaptation of the proposed expert learning algorithm mentioned above achieves this in practice on a multi-regime process.

Their focus in online distributed optimization concerned the lack of similar performance guarantees for online optimization algorithms if dynamic regret is the desired benchmark. Previously, the best known bound in the literature for general convex problems depended explicitly on time [8]. Taking motivation from other works in the online optimization literature [9], they wished to see if there were additional assumptions such that an algorithm would have performance bounds without this explicit time dependence. They successfully showed that such an algorithm existed but removed the time dependence at the cost of introducing a new term that is negligible in many practical situations. Furthermore, they showed via a target tracking example that this quantity cannot be improved on for this algorithm in general.

Finally, they focused on robust optimization, where the focus is on minimizing the maximal possible loss if many different objective functions are feasible. Recent work developed a feasible, data-driven way to solve this problem under several different practical scenarios, though the methodology developed did not focus on time-varying scenarios [16]. One work released in early 2019 proposed a solution [10], but it was based on choosing a model class that may not hold in practice. To avoid this model dependence, combining well-understood methodologies in statistics for bootstrapping time dependent data to accurately estimate a sample of future data without explicitly choosing a model class; such methodology is guaranteed to sample the true distribution in a wide variety of cases. They then showed similar performance of this procedure to that if they would have known the true distribution in the future.

2 INTRODUCTION

Optimization research exploded in recent years due to increased computational power allowing for training of statistical models never before possible. Much recent work focused on the case where the objective function does not change over time, though some work focused on the time-varying setting. The researchers in this setting considered two particular settings. The first scenario assumed that the changes in objective functions were smooth in time. Using this smoothness, researchers developed techniques for predicting future stationary points by using predictor-corrector methods on a constructed differential equation that approximated the evolution of a stationary point; this allowed for reasonable prediction of future stationary points if already close to one. The second scenario investigated concerned the situation where the dynamical system of objective functions was adversarial (non-predictable). In this setting, researchers focused on deriving worst-case performance guarantees using adaptations of standard optimization algorithms. Works in both directions were extended to cover cases involving time-varying constrained problems (though often operated under the assumption that the constraints and problems were convex), as well as multiple objective functions through the online distributed optimization literature

The Duke University team was tasked with continuing the time-varying optimization literature. They focused on the development of methods and concepts that better dealt with real-world situations, which included: factoring in more realistic assumptions on the dynamical systems underlying the objective functions and constraints, nonconvexity of objectives, reasonability of currently existing performance measures, time-varying constraints, and situations where the true objective function is unknown. The problems considered in the body of work span a large number of subfields of optimization and are summarized in the remainder of the report. As applicable, all numerical methods developed were tested on financial data obtained from the New York Stock Exchange.

3 MAJOR GOALS

The goals of the Duke University team were quite ambitious and are summarized as:

1. Determine the minimal number of assumptions needed in order obtain theoretical and practical performance improvements using algorithms that were built by minimally adapting previous ones in the literature.
2. Define performance metrics for online optimization that were more suitable for optimization problems encountered in practice that received little to no attention in the literature
3. Construct algorithms for online optimization that provided performance guarantees where few assumptions are reasonable.

4 METHODS, ASSUMPTIONS, AND PROCEDURES

The general methods, assumptions, and procedures of the work are not dissimilar to other standard procedures utilized in the development of computational mathematics. The primary tool used to read and derive new mathematics consisted of standard writing implements and surfaces, as well as a personal computer and tablet. Given the subject matter, no additional facilities were used. Computational experiments were developed in MATLAB and Python developed on an Acer Predator laptop running the Windows 10 operating system. Financial data obtained was obtained from historical daily New York Stock Exchange returns.

4.1 Goal 1

4.1.1 Parametrizability and Predictability of Continuous Objectives in Online Optimization

4.1.1.1 Problem Statement and Assumptions

The following online optimization problem was considered: given a sequence of objective functions $f_1, \ldots, f_T$ defined on a continuous domain, construct a sequence of candidate optima $x_1, \ldots, x_T$ such that x_T is only selected based on the objective functions received up until time $t-1$. The goal was to minimize the following quantity known as dynamic regret, where the algorithm's performance was compared to that of the optimal point of the objective at each time.

$$R_T^d = \sum_{t=1}^{T}(f_t(x_t) - \min_{x_t^*} f_t(x_t^*)) \tag{1}$$

The team assumed that the objective function at each time was of the form $f_t(x) = f(x, \theta_t)$, where θ_t is some finite dimensional parameter governing the objective function of interest. To understand the role of accurate prediction of θ_t in minimizing the dynamic regret, the following assumptions were made: the domain of the objective functions was assumed to be convex, the objectives were assumed to be strongly convex in x with strong convexity parameter λ, and the objective functions and both their gradients in both x and θ were assumed to be Lipschitz continuous with constants G, L_x, L_θ respectively.

4.1.1.2 Performance of Predictive Gradient Descent

Under the above assumptions, the dynamic regret obtained by performing k gradient descent steps with step size η at each time based on the predicted values of θ_t satisfies the following bound

$$R_T^d \leq \frac{G\,\|x_1 - x_1^*\|}{1-C^k} + \frac{GC^k}{1-C^k}\sum_{t=2}^{T}\|x_{t-1}^* - x_t^*\| + \frac{G\eta L_\theta}{1-C^k}\sum_{t=2}^{T}\|\hat{\theta}_t - \theta_t\| \tag{2}$$

where C is a positive constant depending on L_x and λ that is smaller than 1.

4.1.1.3 Simultaneous Modeling and Descent (SMAD) Algorithm

Input: Parameters $\beta, \eta, \gamma > 0$, initial data $\theta_{-i_0}, ..., \theta_0$
Output: $p_t = [p_{t,1}, ..., p_{t,N}]$ (predictive distribution over the collection of N models), x_t (predictive points for optimization)
Initialize $w_{1,0} = ... = w_{k_0,0} = 1$, $w_{k,0} = 0$ for $k > k_0$
Initialize $M = k_0$
for $t = 1 \to T$ **do**
    if Initializing new model **then**
        Compute $w_{i,t-1} = (1-\beta)w_{i,t-1}$ for $i \le M$
        Compute $w_{M+1,t-1} = \beta$
        Compute $p_{i,t-1} = \frac{w_{i,t}}{\sum_{i=1}^{N} w_{i,t}}$
        Initialize $x_{t-1}^{M+1} = x_{t-1}$
        Compute $M = M + 1$
    end if
    Observe θ_t
    Initialize $x_t = 0$
    for $1 \le i \le M$ **do**
        Receive $v_i = \Pi_{\mathcal{X}}(x_{t-1}^i - \eta \nabla_x f(x_{t-1}^i, \hat{\theta}_t^i))$ from expert i
        Compute $x_t = x_t + p_{i,t-1} v_i$
        Compute $l_i = \exp[-\gamma f(\mathcal{X}(v_i), \theta_t)]$
        Compute $w_{i,t} = p_{i,t-1} l_i$
    end for
    Output $x_t = \Pi_{\mathcal{X}}(x_t)$
    Compute $w_{i,t} = p_{i,t-1} l_i$
    Compute $p_{i,t} = \frac{w_{i,t}}{\sum_{i=1}^{N} w_{i,t}}$
end for

Figure 1: Details of the SMAD Algorithm

4.1.1.4 Performance of SMAD

If the if-statement is ignored in Figure 1, SMAD has the following upper bound on dynamic regret

$$R_T^d \le \frac{G\,\|x_1 - x_1^*\|}{1-C^k} + \frac{GC^k}{1-C^k}\sum_{t=2}^{T}\|x_{t-1}^* - x_t^*\| + \frac{G\eta L_\theta}{1-C^k}\min_{1\le n\le N}\sum_{t=2}^{T}\|\hat{\theta}_t^n - \theta_t\| + \frac{D\sqrt{2T}}{4}(1+\ln N) \quad (3)$$

where D is the difference between the largest and smallest possible values of the objective function and $\hat{\theta}_t^n$ denotes the prediction of θ_t by model n.

4.1.2 Markovian Availability in Bandit Optimization

4.1.2.1 Problem Statement and Assumptions

Assume that in total, a user has access to a collection $b_1, ..., b_n$ of possible actions. At each time $t = 1, ..., T$, a nonempty subset of this total collection of actions is available. From this subset of actions, a user must select one of them to perform at each time. Denoting the selected action by a_t, the user receives a reward $r_t(a_t)$. No assumptions on the structure of r_t were made. The actions available at each time were assumed to be Markovian in the following sense: the probability that a given subset of actions is available at time t depends directly on the actions available at time $t-1$ and nothing else. The goal in this setting was to minimize the following notion of regret.

$$R_T^d = \sum_{t=1}^{T}(r_t(a_t) - \mathbb{E}_t r_t(a_t^*)) \tag{4}$$

Here, $\mathbb{E}_t$ denotes expectation with respect to the distribution of available subsets of actions at time t, and a_t^* is the best action in hindsight at time t with respect to the available set of actions (note that a_t^* is a random variable with respect to the distribution of subsets of available actions at time t).

4.1.2.2 Algorithm for Optimizing Regret for Bandits with Markovian Availability

Input: Parameters $1 > \alpha, \beta_{\text{on}}, \beta_{\text{off}}, \gamma > 0$, $\varepsilon > 0$
Input: Distribution p on four groups
Initialize r a vector of zeros with length k equal to number of bandits
for $t = -T_0 \to 0$ **do**
  Observe available bandits
  for each bandit b **do**

$$(N^b_{\text{on}}, N^b_{\text{off}}) = (N^b_{\text{on}}, N^b_{\text{off}}) + \begin{cases} (1,0) & \text{if } b \text{ is on} \\ (0,1) & \text{otherwise} \end{cases}$$

$$(N^b_{\text{on,on}}, N^b_{\text{on,off}}) = (N^b_{\text{on,on}}, N^b_{\text{on,off}}) + \begin{cases} (1,0) & \text{if } b \text{ is on when previously on} \\ (0,1) & \text{if } b \text{ is off when previously on} \\ (0,0) & \text{otherwise} \end{cases}$$

$$(N^b_{\text{off,on}}, N^b_{\text{off,off}}) = (N^b_{\text{off,on}}, N^b_{\text{off,off}}) + \begin{cases} (1,0) & \text{if } b \text{ is on when previously off} \\ (0,1) & \text{if } b \text{ is off when previously off} \\ (0,0) & \text{otherwise} \end{cases}$$

  end for
end for
for each bandit b **do**
  Compute $\bar{M}^b = \begin{pmatrix} \frac{N^b_{\text{on,on}}}{N_{\text{on}}} & \frac{N^b_{\text{on,off}}}{N_{\text{on}}} \\ \frac{N^b_{\text{off,on}}}{N_{\text{off}}} & \frac{N^b_{\text{off,off}}}{N_{\text{off}}} \end{pmatrix}$
end for
Partition bandits into four groups depending on values of $\bar{M}^b_{1,1}$, $\bar{M}^b_{2,2}$, β_{on} and β_{off}
for $t = 1, \ldots, T$ **do**
  Observe available actions
  Compute best ordering $\hat{\sigma}$ of observed actions via Follow the Perturbed Leader on r
  Flip a coin that is heads with probability γ
  if heads **then**
    $r = r + \alpha\mathbf{1}$
    Sample group from p
    if If at least one bandit from sampled group available **then**
      Sample bandit b from sampled group
      Pull bandit and receive reward R_t
      $r = r + \frac{kR(\bar{M}^b_{1,1} + \bar{M}^b_{2,2})}{\gamma \bar{M}^b_{1,1}}\mathbf{e^b}$
    else
      Pull best action with respect to $\hat{\sigma}$ and receive reward R_t
    end if
  else
    Pull best action with respect to $\hat{\sigma}$ and receive reward R_t
  end if
end for

Figure 2: Exploration-Exploitation Algorithm for Bandits with Markovian Availability

4.1.3 Improved Dynamic Regret in Distributed Optimization

4.1.3.1 Problem Statement and Assumptions

The following distributed optimization scenario was considered. There are n agents whose goal is to optimize a sequence of time-varying objective functions $f_0, \ldots, f_T$ over a continuous domain. The objective functions are of the form $f_t = \sum_{i=1}^{n} f_{i,t}$ The agents are decentralized and agent i only has access to $f_{i,0}, \dots, f_{i,t-1}$ at time t. Each agent must choose an action $x_{i,t}$ such that the following notion of dynamic regret is minimized.

$$R_T^d = \frac{1}{n}\sum_{i=1}^{n}\sum_{t=0}^{T}(f_t(x_{i,t}) - \sum_{t=0}^{T}\min_{x_t^*} f_t(x_t^*)) \tag{5}$$

For the purposes of theoretical analysis and algorithm development, the following assumptions were made: the f_t are strongly convex over a continuous, convex domain with strong convexity parameter μ, the $f_{i,t}$ and their gradients are Lipschitz continuous with constants L_f and L_g respectively for all i and t, and the matrix W referenced below is symmetric and doubly stochastic. This implies that $\sigma_W = \left\| W - \frac{1}{n}\mathbf{1}\mathbf{1}^T \right\| < 1$, where $\mathbf{1}$ denotes the column vector of all 1's.

4.1.3.2 Online Push-Pull Algorithm

Require: The primal variable $x_{i,0}$, the local gradient $\nabla f_{i,0}(x_{i,0})$ and global gradient estimate $y_{i,0} = \nabla f_{i,0}(x_{i,0})$ for all i, and the doubly stochastic matrix $W := (w_{i,j})$.

for $1 \le t \le T$ **do**

Agent i computes

$$x_{i,t+1} = \sum_{j \in N_i} W_{ij}(x_{j,t} - y_{j,t}).$$

Agent i computes

$$y_{i,t+1} = \sum_{j \in N_i} W_{ij} y_{j,t} + \alpha_j(\nabla f_{i,t+1}(x_{i,t+1}) - \nabla f_{i,t}(x_{i,t}))$$

end for

Figure 3: Push-Pull Gradient Tracking Algorithm for Online Distributed Optimization

4.1.3.3 Performance of Online Push-Pull Algorithm

Define the following quantities:

$$P_T = \sum_{t=0}^{T-1} \|x_{t+1}^* - x_t^*\| \tag{6}$$

$$g_t(x) = [\nabla f_{1,t}(x), \dots, \nabla f_{n,t}(x)] \tag{7}$$

$$V_T = \sum_{t=0}^{T-1} \|g_{t+1}(x_{t+1}^*) - g_t(x_t^*)\| \tag{8}$$

Under the assumptions above, if the α_j are chosen uniformly such that

$$\alpha_j = \min\left(\frac{(1-\sigma_W)^2}{3\sigma_W\left(\frac{L_g}{\mu}n+1\right)+(1-\sigma_W)\sigma_W}\frac{1}{L_g}, \frac{1}{L_g}\right) \tag{9}$$

Then the dynamic regret is bounded by

$$R_T^d \le O(C_1 + C_2 + C_3 + P_T + V_T) \tag{10}$$

where the C_i are all constants depending on the initial guesses x_0 and y_0.

4.2 Goal 2

4.2.1 KKT Regret: An Alternative Performance Measure

4.2.1.1 Definition of KKT Regret

For objectives f_t with inequality constraints $g_t \leq 0$ and equality constraints $h_t = 0$ (in the vector sense), the KKT regret for an algorithm outputting a sequence $\{(x_t, \xi_t, \mu_t)\}_{t=0}^{T}$ is:

$$R_{KKT}^{T}(\{(x_t, \xi_t, \mu_t)\}_{t=0}^{T}) = \sum_{t=0}^{T} \left(\|\nabla L(x_t, \xi_t, \mu_t)\|^2,\ \lambda_{min}\left(\nabla^2 L(x_t, \xi_t, \mu_t)\right) \right) \quad (11)$$

Here, $L(x_t, \xi_t, \mu_t) = f_t(x_t) + \xi_t \cdot g_t(x_t) + \mu_t h_t(x_t)$ and λ_{min}denotes the minimum eigenvalue of a matrix.

For theoretical analysis, only the case of linear equality constraints was considered, implying that $g_t = 0$ and $h_t = A_t$ is a matrix. The restriction to equality constraints only implies that $\lambda_{min}\left(\nabla^2 L(x_t, \xi_t, \mu_t)\right)$ can be safely ignored as this condition (i.e. minimum eigenvalues only matter for inequality constraints). The same assumptions as in 4.1.1 were made. Specifically, the team assumed that the objective function at each time was of the form $f_t(x) = f(x, \theta_t)$, where θ_t is some finite dimensional parameter governing the objective function of interest. To understand the role of accurate prediction of θ_t in minimizing the dynamic regret, the following assumptions were made: the domain of the objective functions was assumed to be convex, the objectives were assumed to be strongly convex in x with strong convexity parameter λ, and the objective functions and both their gradients in both x and θ were assumed to be Lipschitz continuous with constants G, L_x, L_θ respectively. In addition, the possible values of $\mu_t$at each time were assumed to be bounded for all time, and the A_t were assumed to have bounded operator norm for all time.

4.2.1.2 Predictive Primal-Dual Algorithm

Input: Step size $\eta > 0$, and $x_1 \in X$
for $t = 1 \to T$ **do**
  Receive parameter θ_t
  Predict $\hat{\theta}_{t+1}$ from $\theta_t, ..., \theta_1$
  Predict $\hat{A}_{t+1}$ from $A_t, ..., A_1$
  Compute $x_{t+1} = x_t - \eta \nabla_x f(x, \hat{\theta}_{t+1}) - \eta \mu_t \hat{A}_{t+1}$
  Compute $\mu_{t+1} = \mu_t + \eta \hat{A}_{t+1} x_{t+1}$
end for

Figure 4: Predictive Primal-Dual Algorithm

4.2.1.3 Analysis of KKT Regret for Equality Constrained Case

For η sufficiently small, the proposed predictive primal-dual algorithm achieves KKT regret bounded by sums of constant multiples (depending on η and the other constants in the assumption) of the following terms

$$\|x_1 - x_1^*\|, \|\mu_1^*\|, \sum \| \theta_t - \hat{\theta}_t \|, \sum \| \mu_t - \hat{\mu}_t \|, \sum \| \mu_{t-1} \| \|A_t - \hat{A}_t\|, \sum \| \theta_t - \widehat{\theta_t} \|^2,$$
$$\sum \| \theta_t - \hat{\theta}_t \| \| (\|\mu_{t-1} - \hat{\mu}_{t-1}\| + \|\mu_{t-1}\| \|A_t - \hat{A}_t\|),$$
$$\sum \| \mu_{t-1} \|^2 \|\hat{A}_t - A_t\|, 2 \sum \| \mu_{t-1} \| \|\mu_{t-1} - \hat{\mu}_{t-1}\| \|A_t - \hat{A}_t\|, \sum \|\mu_{t-1} - \hat{\mu}_{t-1}\|^2,$$
$$\sum \|x_t^* - x_{t-1}^*\|, \sum \|x_t^* - x_{t-1}^*\|^2, \sum \| \mu_t^* - \mu_{t-1}^* \|, \sum \| \mu_t^* - \mu_{t-1}^* \|^2 \tag{12}$$

In the above, $\hat{\mu}_t$ is the value of the μ parameter assuming perfect knowledge of the future at time t, and both x_t^* and μ_t^* are the optimal x and μ at time t

4.2.2 Asymptotically Efficient Optimization

4.2.2.1 Definition of Feasible Regret

Let g_t be a sequence of possibly nonconvex objective functions. Given a sequence of decisions $x_1, \dots, x_{t-1}$ and a subset $D_t(x_{t-1})$ of possible decisions depending on x_{t-1}, the ***feasible regret at time t*** is given by

$$R_t^{D_t(x_{t-1})}(x_t) = g_t(x_t) - \min_{x_t^* \in D_t(x_{t-1})} g_t(x_t^*) \tag{13}$$

4.2.2.2 Definition of Asymptotically Efficient Optimization

Given a collection of objective functions $g_t(x) = h(x, \theta_t)$, indexed starting from 1, where θ_t is a stochastic process, let x_1 be an initial guess of a solution, and let $D_t(x_{t-1})$ be a (defined) sequence of possible values of x_t given x_{t-1}. Letting $\boldsymbol{x_{t-1}}$ and $\boldsymbol{\theta_{t-1}}$ be the vectors of the entire history of x and θ up until time $t-1$. Given a statistical model α of θ_t given $\boldsymbol{\theta_{t-1}}$, let $\boldsymbol{A}$ be an algorithm that maps the formal quadruple $(h, \boldsymbol{x_{t-1}}, \boldsymbol{\theta_{t-1}}, \alpha)$ to some point $x_t \in D_t(x_{t-1})$. Let $\alpha_1, \dots, \alpha_n$ be a collection of statistical models and let $x_t^i = \boldsymbol{A}(h, \boldsymbol{x_{t-1}}, \boldsymbol{\theta_{t-1}}, \alpha)$. A model selection procedure $\boldsymbol{M}$ for θ_t is an ***asymptotically efficient optimizer*** for the sequence of optimization problems of minimizing g_t via the algorithm $\boldsymbol{A}$ if, as $t \to \infty$, we have the following convergence in probability.

$$\frac{\mathbb{E}\left(R_t^{D_t(x_{t-1})}\left(x_t^{M(\boldsymbol{\theta_{t-1}})}\right) | \boldsymbol{\theta_{t-1}}\right)}{\min_{1 \le k \le n} \mathbb{E}\left(R_t^{D_t(x_{t-1})}(x_t^k) | \boldsymbol{\theta_{t-1}}\right)} \to 1 \tag{14}$$

4.3 Goal 3

4.3.1 Bootstrapping for Time-Varying Distributionally Robust Optimization

4.3.1.1 Problem Statement and Assumptions

Assume that at each time t, a user receives samples $\theta_t^1, \dots, \theta_t^k$ of a stochastic parameter θ_t governing an objective function of interest. It is assumed that the θ_t are not independent of one another and come from an ultimately unknown distribution. The original goal is to solve the following optimization problem of the future.

$$\min_x \mathbb{E} f(x, \theta_{t+1}) \tag{15}$$

If no reasonable models for the future parameter θ_{t+1} known a priori, an alternative is to solve the following robust optimization problem for a set $\mathbb{B}$ of potential probability distributions:

$$\min_x \max_{\theta_{t+1} \in \mathbb{B}} \mathbb{E} f(x, \theta_{t+1}) \tag{16}$$

Work prior to the Lagrange period of performance established a principled way to do this if one did not require predicting the future: if one received samples $\theta_{t+1}^1, \dots, \theta_{t+1}^k$ of θ_{t+1}, one could consider the empirical distribution of θ_{t+1}given the samples, denoted $\hat{\theta}_{t+1}$, and solve the following robust optimization for fixed $\varepsilon > 0$:

$$\min_x \max_{\{\theta: d_{\mathbb{W}}(\theta, \hat{\theta}_{t+1}) \leq \varepsilon\}} \mathbb{E} f(x, \theta) \tag{17}$$

Here, $d_{\mathbb{W}}$ denotes the Wasserstein distance between two probability distributions. In this setting, optimization problems have a tractable reformulation due to a combination of standard duality conditions and properties of Wasserstein distance. The goal is to develop an algorithm to adapt this to the proposed problem by using available data to generate an approximate sample of $\theta_{t+1}$with minimal assumptions, and then employ the methodologies for the non-predictive case.

The only assumption on distributions we assume are that the θ_t are infinite order autoregressive processes, i.e. for ε_t independent and identically distributed with mean zero and finite variance.

$$\theta_t = \sum_{i=1}^{\infty} a_i \theta_{t-i} + \varepsilon_t \tag{18}$$

4.3.1.2 Bootstrap Algorithm for Time-Varying Distributionally Robust Optimization

Input: Wasserstein radius $\varepsilon > 0$, initial samples $\theta_{-t_0}, \dots, \theta_0$,
size of bootstrap sample $k \geq 1$
for $t = 1 \to T$ **do**
  Construct sieve bootstrap sample $\theta_t^1, \dots, \theta_t^k$ from $\theta_{-t_0}, \dots, \theta_{t-1}$
  Compute $x_t = \text{argmin}_x \max_{\{\theta: d_{\mathbb{W}(\theta, \hat{\theta}_t)} \leq \varepsilon\}} \mathbb{E} f(x, \theta)$
  Observe θ_t
end for

Figure 5: Bootstrap Algorithm for Time-Varying Distributionally Robust Optimization

5 ACCOMPLISHMENT UNDER GOALS

5.1 Goal 1

Invented adaptation of gradient tracking methods to online distributed optimization problems that gave improved theoretical dynamic regret bounds under additional but standard assumption on curvature of objectives, more robust performance with respect to step-size choices than previous state of the art, and empirically showed that achieved bounds cannot be significantly improved in general.

Proposed combining online optimization research with time series research, showed theoretically and empirically by simple but powerful example analyses that performance gains can be significantly improved, and developed quickly adaptable optimization algorithms for both discrete and continuously parametrized cases to simultaneously predict and optimize.

5.2 Goal 2

Developed the KKT Regret as an alternative performance measure suitable for general non-convex, time-varying constrained optimization problems, analyzed standard primal-dual algorithm to illustrate the types of quantities that will come up in theoretical analysis, and showed that predictability of objective functions can significantly improve performance of algorithms with respect to KKT Regret

Defined the notion of asymptotically efficient optimization, extending that of Shibata originally proposed for model selection, and showed that a standard adaptation of the prominent expert learning algorithm can be used to achieve this in practice.

5.3 Goal 3

Adapted methodology developed for distributionally robust, fully data-driven optimization, that provided worst-case performance guarantees for stochastic optimization problems in time-invariant, independent data setting to time-varying, dependent data setting valid for a large number of stochastic processes encountered in practice by novel application of bootstrap

6 RESULTS AND DISCUSSION

The results below were in general not developed with respect to satisfying one particular goal. Applicable goal will be mentioned in each subsection.

6.1 Goal 1

6.1.1 Parametrizability and Predictability of Continuous Objectives in Online Optimization

As discussed above, the literature prior to the beginning of the contract was concerned with two main ways in which objective functions could change: smoothly in time, and adversarially in time. The team found this surprising given that many data-generating processes observed do not fit either assumption. Furthermore, given that objective functions in time-varying of practical interest depended on these processes (e.g. target tracking, finance), the team was interested in understanding the extent at which one could improve results in the adversarial setting if one assumed that the objective functions were generated from a stochastic dependent process, which allowed for considering the ability to predict the future in a more general case than one given by the smoothly time-varying setting.

The main performance measure of interest was dynamic regret: the sum of the algorithm's performance versus that of the true optimal value of the objective at each time. In the general case of prediction (i.e. not appealing to expectations), we proved that any algorithm developed in the adversarial setting could have its bounds on dynamic regret lowered if they factored in prediction provided that the prediction of the future objective function was cumulatively sufficiently accurate, specifically by decreasing values of constants in the bounds in the adversarial setting. The bounds in question suggested a lack of ability to further improve these bounds in general given that the bounds mimicked those found in time-invariant settings. The assumptions used for these bounds were standard in the optimization literature: the objectives were differentiable, and both the objectives and their gradients were assumed to be Lipschitz continuous. The objectives were also assumed to be strongly convex: convexity or something very close to convexity was required as otherwise dynamic regret would not have made sense (see below for a subsection addressing this); therefore, strong convexity was chosen to provide better intuition and allow for comparison with other bounds in the literature.

The result above was not of sufficient practical interest on its own accord as objective functions naturally live in infinite dimensional spaces. However, as previously discussed, the objective function of interest at each time step generally only depends on a finite dimensional process (such as stock returns). This allowed an additional minor assumption. More specifically, they assumed that the time-varying portion of the objective function is contained within an explicit parameter that is finite-dimensional in practice. Provided that the objective function was uniformly dependent on the parameter, the above bounds could be written explicitly in terms of the data, suggesting that the performance of an optimization algorithm could be directly estimated from the quality of estimation for the data generating process. Though seemingly strong, it was found to be difficult to relax this assumption without further complicating the bounds, as the natural language for optimization in terms of epiconvergence required some degree of uniformity. Connections with these bounds to the smoothly time-varying case and the case that the objective functions of interest were conditional expectations were discussed, strengthening the idea that optimal optimization was given by optimal prediction. The assumption that the data generating process was finite dimensional allowed for the development of a practical algorithm for learning how to optimize. Specifically, an algorithm based on expert learning was proposed. Given a collection of candidate models of the data generating process, each expert represented the performance of some model class. For a new sequence of optimization problems, each model is initially assigned uniform weight, reflective of an adversarial setting. An initial point in the domain of the objectives was also given. At each time, each model attempted to predict the future.

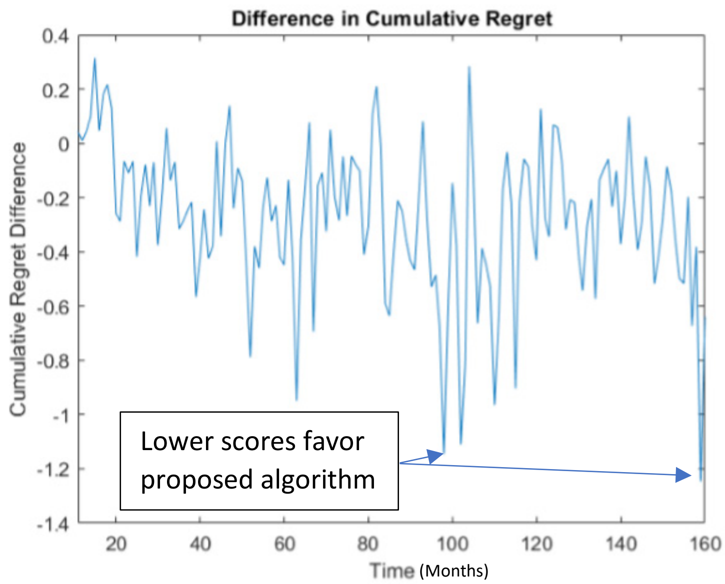


Figure 6: Performance of the proposed expert learning method in portfolio optimization using incorrect prediction models versus no prediction

Based on the prediction, the expert ran a prescribed optimization based on the prediction and the previous sequence of candidate optimal points in the domain. After the actual data value was observed, each model was reweighted to reflect its performance on optimization, which allowed the algorithm to automatically tune itself based on data observations.

An experiment in portfolio optimization was performed and output in Figure 6. This experiment considered the prediction of someone's optimal portfolio in the sense of Markowitz using the proposed algorithm with incorrect models. Daily stock returns from the NYSE were used. Their risk was observed each month and was modeled as autoregressive process of lags 1 month through six months; the true process was assumed to be the same as the previous month with probability 0.9 and a uniformly chosen random integer between 1 and 20 otherwise after an initial period of 8 months being 4 plus a Gaussian random vector with mean zero and covariance 0.64. The average stock performance used to evaluate their optimal portfolio was assumed to be between 15 and 90 days in increments of 15; the actual window was 50 days. The average difference in cumulative regret between the proposed algorithm and an adversarial, non-predictive algorithm in Figure 6 showed that the proposed algorithm could provide benefits despite the lack of correctness of the models used.

6.1.2 Markovian Availability in Bandit Optimization

The team also considered the problem of bandit optimization: an algorithm was tasked with choosing one of several abstract actions known as bandits. The bandits rewarded the algorithm with some numeric award, where larger positive numbers were favorable. The goal was to maximize total reward. In contrast to the previous subsection, the domain of possible options was discrete, so the above was not applicable. The problem of time-varying awards had been discussed in varying degrees within the literature for a number of years.

The problem of time-varying availability of bandits, however, had been very limited. At the beginning of the program, works had considered time-varying availability of bandits, but assumed that the collection of bandits available at each individual round was adversarial, which was not reflective of many real world scenarios where availability of future actions depended directly on the availability of actions in the past. Analysis was restricted to the case that the dynamics of the available actions were given by a Markov chain defined on the collection of all subsets of available actions. By Markov chain, we mean that the probability that a given collection of actions is available depends only on the actions that were available in the immediate past. We assumed no structure on available rewards, though experimentally they were randomly generated between 0 and 1.

The naïve adaptation of previous methods, which involved learning the dynamics of the change in availability before making any choices, became computationally infeasible due to the size of the number of subsets of available actions; in the case of 14 total bandits, the number of possible subsets exceeded the population of Earth as of 2017. Adaptations were needed to make any such algorithm feasible. We proposed *improper estimation* of the Markov chain; rather than concern ourselves with the whole chain, our algorithm only kept track of marginals of each individual bandit being available or unavailable rather than the actual transition probabilities. This process was equivalent to modeling each bandit's availability as an independent 2 x 2 Markov chain, which drastically reduced the computational resources required. After sufficient time passed, upon which the algorithm would deem that the marginals had been learned, each bandit was classified into one of four categories based on their marginal probabilities. The algorithm then chose bandits in an exploration-exploitation fashion via coin flip. If exploitation was chosen, the algorithm chose the best-known available action based on average reward received prior. If it chose exploration, the algorithm chose one of the four classifications based on predefined weights, and then chose uniformly at random one of the bandits belonging to the chosen class.

An experiment testing the algorithm was conducted, with results in Figure 7, to investigate the performance of this method. The awards were generated for each bandit (five in total) by independent Brownian motions that, for each bandit, started at some uniformly random point chosen on the unit interval that proceeded as Brownian motions with variance 0.02. The bandits were independently available with transition matrices given in Table 1. The performance of this algorithm was compared to that of EXP4, the only other available algorithm capable of handling the Markov case, but suffers from high memory costs and results. The experiment occurred for 500 time iterations after learning the Markov chain marginals for 100 time steps, and the per round regret with respect to the post-factor best decision was measured. The results in Figure 7 showed the superiority of the proposed method by plotting average results after repeating the experiment 1000 times.

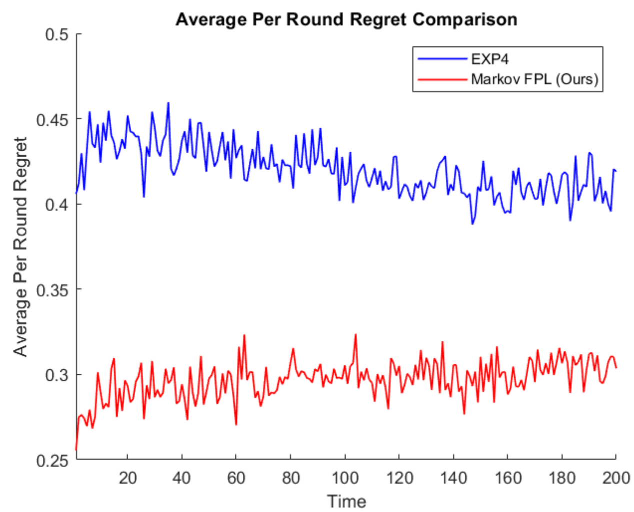


Figure 7: Performance of the proposed learn then exploit/explore algorithm versus EXP4

Table 1: Transition Probabilities for Experiment in Figure 7

	Bandit 1	Bandit 2	Bandit 3	Bandit 4	Bandit 5
Available-Available	0.1	0.15	0.5	0.7	0.9
Available-Unavailable	0.9	0.85	0.5	0.3	0.1
Unavailable-Available	0.9	0.1	0.5	0.3	0.7
Unavailable-Unavailable	0.1	0.9	0.5	0.7	0.4

6.1.3 Improved Dynamic Regret in Distributed Optimization

The team also studied distributed optimization problems, where multiple agents participated in optimizing a joint function, which was assumed to be the sample averages of individual functions the agents had access to, in a decentralized measure. This meant that agents could communicate with one another as to their predicted optimal point, but had no access to the other agents' objective functions. Previous work in this area before the period of performance showed that a distributed adaptation of the mirror descent algorithm for time-varying convex functions achieved a theoretical regret bound that explicitly depended on the time horizon of the problem at hand, as was true of the centralized setting. However, in the centralized setting, it was also known that such explicit dependence could be removed provided that the objective functions at each time were strongly convex, a condition giving a lower bound on the curvature of the objectives at each point. It was thus natural to ask whether such result held in the distributed setting, or whether the required communication intrinsically prevented that.

A modified version of the push-pull algorithm in time-invariant distributed optimization answered this question in the affirmative. At each time, each agent performed a descent step on their individual objective functions, and then communicated time-corrected gradients to the other agents. It was shown that this algorithm had theoretical regret bounds without explicit dependence on the time horizon, though at the cost of adding a term involving the sum of the magnitude of the differences in the gradients at the optimal points at each time. This quantity did not appear in the centralized bound, however, it turned out to be negligible in practice, as the quantity was provably zero in practice provided that the optimal points were inside the constraint set. Furthermore, empirical results validated that this gradient term could not in general be removed in any bound for the proposed algorithm, which suggested that the bound for the proposed algorithm was in a sense optimal. The key in generating this example involved creating a situation where the optimal point at each time was the same, but the constraints were time-varying.

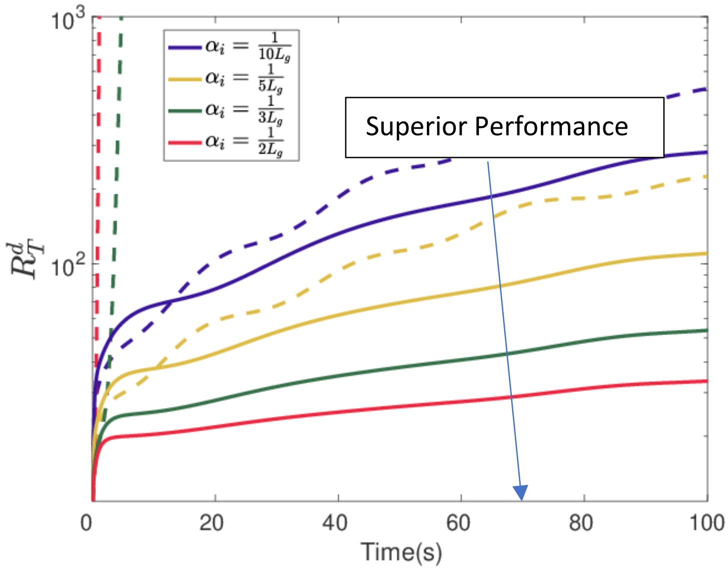


Figure 8: Comparison of the cumulative regret of the proposed push-pull online distributed optimization method with online mirror descent.

It was natural to compare the proposed push-pull algorithm with the previously proposed mirror descent-based algorithm. This was done for a gradient tracking example detailed in the paper section of the Appendix. The results are posted in Figure 8, where the solid lines indicate the performance of the proposed algorithm and the dashed lines indicate the performance of the mirror descent method for various step sizes. Not only was the proposed method superior at each step size, it was also more stable for larger step sizes, which was not true of the mirror descent algorithm.

6.2 Goal 2

6.2.1 *KKT Regret: An Alternative Performance Measure*

As previously mentioned, almost every paper on online optimization measured algorithm performance with respect to that of a particular optimal value. Though time-varying constraints were factored in if necessary, the analysis of the theoretical performance of these algorithms treated objective and constraint regret as two separate quantities, which ignored well-known mathematical language developed in the past on these subjects. Furthermore, these performance measures were unable to adapt to many scenarios of practical interest, where functions of interest were non-convex (analysis had been performed on certain cases of non-convexity but could not be adapted in general).

The team formulated the notion of KKT regret, based on previously developed measures for non-convexity. This regret measured performance in a manner based on the well-known KKT conditions in classical optimization theory, and allowed for the performance of any algorithm that outputted candidate evaluation points as well as dual coefficients for the KKT conditions to be evaluated. At each time, the KKT regret collects a vector of nonnegative numbers that details exactly how much the output of the algorithm deviated from those required of the KKT conditions at each individual time. The total KKT regret is the sum of these conditions. In addition, the proposed regret also measured the deviation from satisfaction of the well-known second order sufficient conditions that the Hessian of the Lagrangian is positive definite by observing the deviation from positivity of the smallest eigenvalue of the Hessian.

As in the section on predictability, analysis could be conducted on any algorithm that solved a time-varying constrained optimization problem. For purposes of illustration of the types of quantities present in the resulting performance bounds on the proposed measure, a simple case of time-varying linear constraints was analyzed under a parameter predictive scenario as was mentioned in the previous sections. The resulting bounds obtained were complicated but the dependencies on quantities pertaining to the dynamical system were unsurprising based on previous quantities considered in the online optimization literature and included: changes in location of optimal point, changes in linear constraint, and quality of prediction of the true optimal values.

It was difficult to conduct any experiment illustrating that the KKT regret proposed was superior to previous performance metrics; its benefits lie in its generalizability, not that it necessarily captured something better than other proposed metrics (at least for convex and nearly convex objectives as previously discussed). Nevertheless, relating back to a previous section, it was useful to determine whether prediction of future objective functions made a difference in terms regret for practical purposes; given that the performance metric changed from objective values to norm of gradients, this was not immediately clear. To test this, the following portfolio optimization problem was considered.

The same NYSE data as previously mentioned was used. At each time, the objective function used was the squared norm generated by the stock covariance matrix, which was estimated using a 30 day window. The time-varying constraints used were lower bounds on the expected return by constants, with the average return of each stock again estimated using a 30 day window. The constants in the lower bound alternated between -0.1, -0.05, 0.05, and 0.1 every 90 days, simulating desires to be greedy and focus more so on returns than safety before ultimately having a change of heart and returning to safe again, and repeating this as long as desired. Two models of the constraint were considered: one that was aware of the cyclic nature of the constraints and accurately knows future constraints, and one that never picks up on this and always used the currently known constraints. The difference in KKT regret for this experiment was plotted in Figure 9. It was easily shown that accurate prediction was beneficial for minimizing the KKT regret.

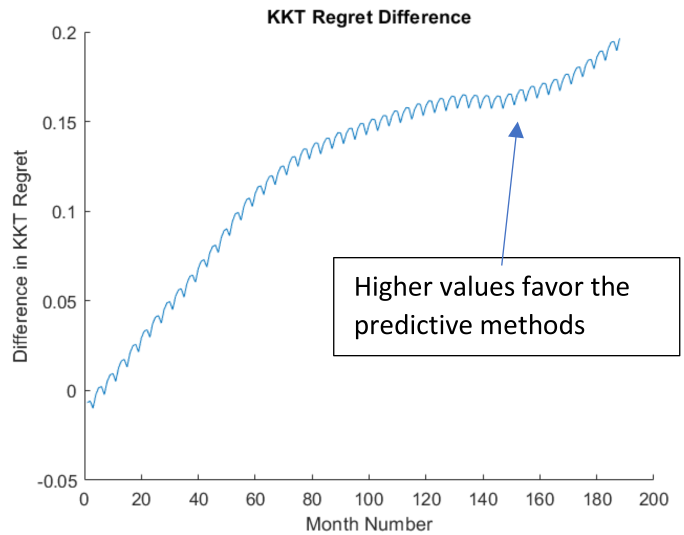


Figure 9: Difference in cumulative KKT regret between perfect predictive and nonpredictive methods.

6.2.2 Asymptotically Efficient Optimization

Even though the KKT regret proposed helped address issues related to performance metrics in terms of generalizability, there remained intuitive properties to be satisfied. It has been well observed that models rarely captured actual processes; they just approximated them well. Given a collection of dynamical models that approximated (but not fully) a particular process governing objective functions, it was natural to ask which of the models led to the best optimizers of the dynamical process given whatever constraints on computational resources existed. Such optimizers could have involved full optimizations of a predicted function if possible but did not have to; it was also feasible to only have a few (gradient) descent steps at each iteration. It was natural to ask whether it was possible to formulate such a question rigorously, as well as whether a practical method existed in order to solve it.

To make this formulation, it was natural to recall past work on time series concerning asymptotically efficient model selection. A model selection process for a given (possibly dependent) stochastic process was asymptotically efficient in the sense of Shibata if, given reasonable stationarity of the process, the model selection process eventually chose the model that achieved minimal expected loss with probability 1 as the amount of observed data points became infinite. This definition was naturally generalizable to the problem of online optimization provided that the process was not-adversarial: the definitions readily generalize to functional data, hence can generalize to optimization. It was clear that this property was desirable, though as with time series, difficult to show theoretically due to various algebraic properties used in proofs of the classical case.

From a practical perspective, it was interesting to determine whether an algorithm existed that was an asymptotically efficient optimizer in practice. It was natural to consider the same basic algorithm considered in the first section of this report in this more general context: the modelling aspect stayed the same, but optimization algorithms other than gradient steps were allowed, such as full optimizations via standard algorithms. As a demonstration, portfolio optimization was once again considered in an example very similar to that in the first section of this report. This experiment considered the prediction of someone's optimal portfolio in the sense of Markowitz when all models considered were incorrect.

Daily stock returns from the NYSE were used. Their risk was observed each month and was modeled as autoregressive process of lags 1 month through six months; the true recursion was generated in this simulation by letting the risk parameter be .5 for the first 240 days followed by a recursion generated by multiplying a uniformly randomly generated point on the eight dimensional unit simplex by 0.75. To simulate uncertainty in risk at any particular time, each computed risk parameter is perturbed by independent exponentially distributed noise with parameter 10. The average stock performance used to evaluate their optimal portfolio was assumed to be between 15 and 90 days in increments of 15; the actual window was 150 days. The results plotted in Figure 10 showed that the proposed exponential weight algorithm in the first section achieved asymptotically efficient optimization even though the governing process of the stock prices was multiregime.

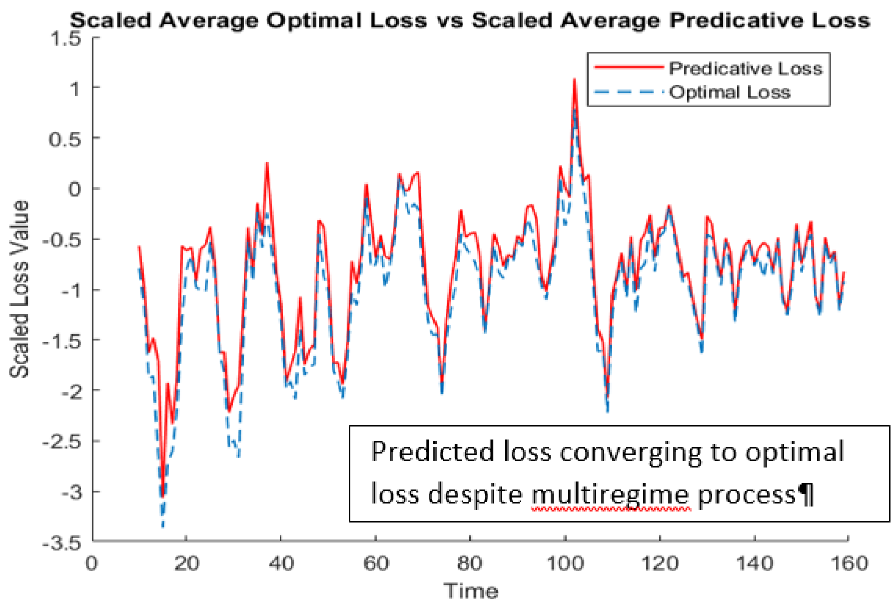


Figure 10: An example of asymptotically efficient optimization on portfolio optimization.

6.3 Goal 3

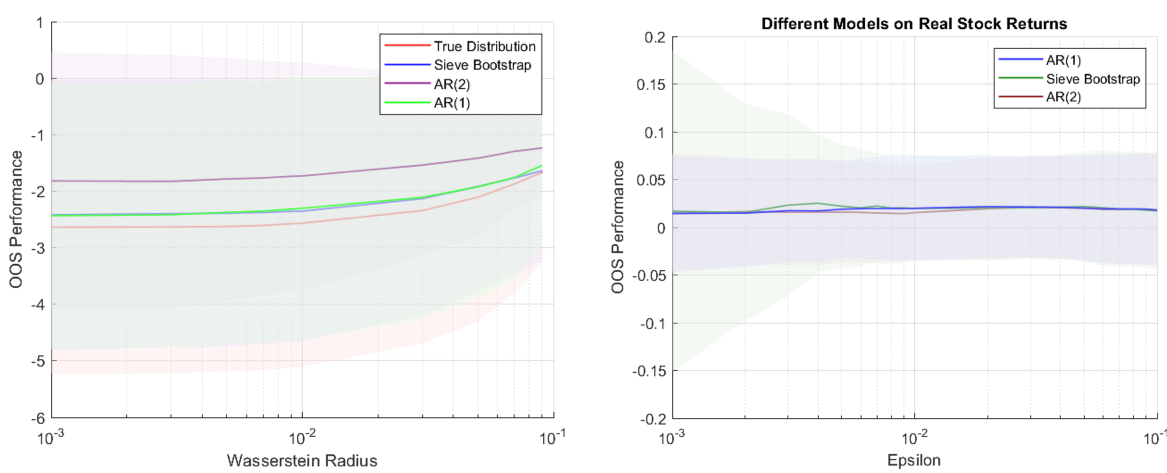
6.3.1 Bootstrapping for Time-Varying Distributionally Robust Optimization

The Duke University team finally considered the case where the process generating the observed data was not only unknown, but was also dfficult or near impossible to model. Without a model for the data generating process, all of the above no longer applied. Nevertheless, good performance was still desirable; this naturally led to the notion of robustness with respect to the distribution of the data, which could be thought of as optimizing with respect to the worst case of a potentially infinite number of objective functions. This problem was studied under the notion of data-driven distributionally robust optimization.

Though traditionally studied via divergence measures between distributions, recent work investigated distributionally robust optimization with respect to Wasserstein balls centered at the empirical distribution of the observed data. Wasserstein distances had the advantage of being able to meaningfully compare discrete empirical distributions with continuous distributions, whereas divergence measures were not able to by definition. It was shown that robust programs in the Wasserstein sense could be reformulated in tractable manners that were applicable to a wide variety of objectives occurring in practice, hence allowing for a way to robustly optimize without assuming any model of distribution for the observed data.

The programs previously studied had by and large only been applicable for situations where available data were assumed to be independently sampled from an identical distribution, an assumption well known to be restrictive in practice. Furthermore, this independent, identically distributed (i.i.d.) assumption was not directly applicable for problems in practice where the distribution of the data, and hence the optimization problem of interest, was time-varying. It was thus of interest to determine if this framework was somehow adaptable to the time-varying setting. The literature review conducted revealed only one work that had considered applying this Wasserstein methodology to the time-varying setting specifically for vector autoregressive models. Though the use of a particular dynamical model was understandable, as some amount of structure was required in order to adapt the methodology, it was natural to ask whether the i.i.d. formulations could be utilized under reasonably general assumptions of the data distribution without directly incorporating the model assumptions into the optimization problems.

A natural solution to this problem came via the sieve bootstrap. The sieve bootstrap was originally proposed as one of a number of methods to bootstrap statistics involving time series data. The sieve bootstrap was immediately adaptable to predicting future distributions given only currently available data: under reasonable assumptions concerning the moments of the innovations of the time series, it was known that the sieve bootstrap procedure could be used to estimate a sample of the future distribution of a time series representable by an autoregressive process of infinite order provided enough data points of the time series were previously observed. This case covered a large number of models studied in practice, such as autoregressive moving average models. This led to a natural algorithm that could be utilized under reasonable assumptions: given realizations of the observed time series, bootstrap a sample of the future, and use this sample to solve the applicable robust program for the time-invariant setting already developed.



(a) (b)

Figure 11: The performance of time-varying robust optimization for different estimations of the future distribution using: (a) the synthetic experiment; and, (b) real stock data

The proposed algorithm was evaluated by considering a synthetic and real data example based off of those in the original paper *Data-driven Distributionally Robust Optimization Using the Wasserstein Metric: Performance Guarantees and Tractable Reformulations* by Eshafani and Kuhn (2018). The objective function considered was the conditional value at risk with the same parameters detailed in the above paper. For the synthetic example, ten assets were used with returns independently following autoregressive processes of lag 2 with average returns between 2% and 20% in increasing increments of 2%. The lag 2 coefficient was equal to 0.1 for all assets, and the lag 1 coefficient was equal to, assuming the assets are ordered by their average return -0.8, -0.6, -0.4, -0.2, 0, 0, 0.2, 0.4, 0.6, 0.8. The following experiment was repeated 100 times. Given 15 observations of the returns, the process for the returns of each asset were modeled separately in four ways: using the Yule-Walker equations to estimate lag 1 and lag 2 autoregressive models, the sieve bootstrap, and assuming the true distribution of the future was known. 200 samples of the future were generated for each asset using each model. These samples were then used to solve the program given in Eshafani and Kuhn (2018) given a value of the Wasserstein radius, and out of sample performance was measured using the next realization of the time series of returns. The average out of sample performance plus and minus one standard deviation was plotted for each Wasserstein radius used for each method used to estimate the future distribution. The lag 2 autoregressive estimation performed the worst (here, more negative is preferable) as expected whereas the other three methods performed similarly, with the sieve bootstrap generally performing the same or better than the lag 1 autoregressive model on average for Wasserstein balls of sufficiently large radius.

A similar experiment was performed on the same NYSE dataset with the same objective function as previous using the Yule-Walker equations with lags 1 and 2 as well as the sieve bootstrap. 300 samples of the estimated future were generated based on 15 days of stock data to solve the robust program, and out of sample performance is based on the returns the following day. The 15 day intervals used were separated by 30 day periods. The results were plotted on the right side of Figure 11 in the same manner as the synthetic example. All estimation methods used generated similar results on average, with the sieve bootstrap having higher variance for small Wasserstein radius but similar overall performance for sufficiently large enough radius but still reasonably small radius.

7 CONCLUSION

Over the course of the performance period, the Duke University team investigated optimization of (potentially multiple) time-varying objective functions, where methods were developed based on available data as much as possible. The main contributions can be fit into three categories: improvements to theoretical performance guarantees were made provided sufficient assumptions concerning the regularity of the objectives, constraints, and dynamics of the sequence of optimization problems at hand, new performance measures more suitable for real-world problems were proposed and analyzed, and interpretable algorithms for solving real-world optimization problems of interest not previously considered in the literature were developed.

Though the investigations made over the course of the period of performance offered a lot of insight into potential directions, there remains a plethora of items left to be investigated. From a theoretical perspective, dynamical systems of optimization problems are still poorly understood, in large part thanks to the natural topology of the space of objective functions being not amenable to analogous constructions commonly used in understanding other dynamical systems of practical interest. It is not immediately obvious as to what implications such theory would be applicable towards the goals of the Department of Defense. Nevertheless, several directions are of clear interest to the DoD, DARPA, and many branches of science as a whole. These include and are not limited to: computational methods for learning dynamical systems and a theoretical understanding of any fundamental limits of learning in this regard, leveraging knowledge of the structure of dynamical systems to develop new optimization methods that achieve superior performance guarantees in theory and practice, time-varying optimization problems with additional structural assumptions, and data-driven distributionally robust optimization in time-varying scenarios. In particular, the latter is of immense interest, as such optimization problems provide theoretical guarantees regard using only observed data and no other assumptions by virtue of the use of Wasserstein distance in their formulation. Significant work outside of this realm, however, would need to be done in better understanding the spaces of observed data before one could readily make progress in this area.

8 RESULTS DISSEMINATION

Results were presented at

1- 2019 International Conference on Stochastic Programming (ICSP) in Trondheim, Norway.
2- 2019 International Conference on Continuous Optimization (ICCOPT) in Berlin, Germany.
3- Future presentations at MIDAS Data Science Symposium in Ann Arbor, MI (11/13/19-11/15/19)
4- 2019 IEEE Conference on Decision and Control (CDC) in Nice, France.

9 PUBLICATIONS

The following works were accepted to the 2019 IEEE Conference on Decision and Control in Nice, France and are attached:

- Appendix A "Prediction in Online Convex Optimization for Parametrizable Objective Functions"
- Appendix B "A Distributed Online Convex Optimization Algorithm with Improved Dynamic Regret"

The following paper (attached) was submitted to IEEE Transactions on Automatic Control, and additional of preprints are in preparation to be sent to journals to be determined:

- Appendix C "Distributed Online Convex Optimization with Improved Dynamic Regret"

10 HONORS AND AWARDS

Arthur. R. Calderbank

- Named International Francqui Chair – Multimodal Data Analytics in the Age of Big Data
- Gave Vincent Meyer Colloquium at Technion-Israel Institute of Technology

Robert Ravier

- Work performed (prior to start of this grant) during PhD studies was directly used in arguments before the U.S. Supreme Court in *Rucho v. Common Cause.*

Vahid Tarokh

- Elected to National Academy of Engineering
- Named Rhodes Family Distinguished Professor
- Named Bass Connections Endowed Professor
- Visited Caltech through May 15, 2018 as a Gordon Moore's Distinguished Scholar
- Named Microsoft Data Science Investigator at Duke Innovation Hub

11 REFERENCES

[1] Chen, N., Agarwal, A., Wierman, A., Barman, S., and Andrew, L. L. H., "Online Convex Optimization Using Predictions," *Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, June, 2015, pp. 191-204.

[2] Kanade, V., McMahan, H. B., and Bryan, B., "Sleeping Experts and Bandits with Stochastic Action Availability and Adversarial Rewards," *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2009, pp. 272-279.

[3] Neu, G., and Valko, M., "Online Combinatorial Optimization with Stochastic Decision Sets and Adversarial Losses," *Proceedings of the 27th International Conference on Neural Information Processing Systems*, December, 2014, pp. 2780-2788.

[4] Yu, H., Neely, M. J., and Wei, X., "Online Convex Optimization with Stochastic Constraints," *Proceedings of the 31st International Conference on Neural Information Processing Systems*, December, 2017, pp. 1427-1437.

[5] Simonetto, A., Mokhtari, A., Koppel, A., Leus, G., and Ribeiro, A., "A Class of Prediction-Correction Methods for Time-Varying Convex Optimization," *IEEE Transactions on Signal Processing*, Vol. 64, No. 17, 2016, pp. 4576-4591.

[6] Fazlyab, M., Paternain, S., Preciado, V. M., and Ribeiro, A., "Prediction-Correction Interior-Point Method for Time-Varying Convex Optimization," *IEEE Transactions on Automatic Control*, Vol. 63, No. 7, 2017, pp. 1973-1986.

[7] Simonetto, A., and Dall'Anese, E., "Prediction-Correction Algorithms for Time-Varying Constrained Optimization," *IEEE Transactions on Signal Processing*, Vol. 65, No. 20, 2017, pp. 5481-5494.

[8] Shahrampour, S., and Jadbabaie, A., "Distributed Online Optimization in Dynamic Environments Using Mirror Descent," *IEEE Transactions on Automatic Control*, Vol. 63, No. 3, 2017, pp. 714-725.

[9] Mokhtari, A., Shahrampour, S. Jadbabaie, A., and Ribeiro, A., "Online Optimization in Dynamic Environments: Improved Regret Rates for Strongly Convex Problems," *Proceedings of the 55th IEEE Conference on Decision and Control (CDC)*, 2016, pp. 7195-7201.

[10] Dou, X., and Anitescu, M., "Distributionally Robust Optimization with Correlated Data from Vector Autoregressive Processes," *Operations Research Letters*, Vol. 47, No. 4, 2019, pp. 294-299.

[11] Zhang, L., Yang, T., Yi, J., Jin, R., and Zhou, Z.-H., "Improved Dynamic Regret for Non-degenerate Functions," *Proceedings of the 31st International Conference on Neural Information Processing Systems*, December, 2017, pp. 732-741.

[12] Cesa-Bianchi, N., and Lugosi, G., *Prediction, Learning, and Games*, Cambridge University Press, New York, NY, January, 2006.

[13] Finn, C., Abbeel, P., and Levine, S., "Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks," *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017, pp. 1126-1135.

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

[15] Shibata, R., “Asymptotically Efficient Selection of the Order of the Model for Estimating Parameters of a Linear Process,” *The Annals of Statistics*, Vol. 8, No. 1, 1980, pp. 147-164.

[16] Esfahani, P. M., and Kuhn, D., “Esfahani, "Data-driven Distributionally Robust Optimization using the Wasserstein Metric: Performance Guarantees and Tractable Reformulations," Mathematical Programming, Vol. 171, No. 1-2, 2015, pp. 115-166.

APPENDIX: PRODUCED PAPERS

Attached are the PDFs of the papers mentioned in the conclusion.