2019 American Control Conference (ACC)
Philadelphia, PA, USA, July 10-12, 2019

On the Time-Varying Distributions of Online Stochastic Optimization

Xuanyu Cao, Junshan Zhang, *Fellow, IEEE*, and H. Vincent Poor, *Life Fellow, IEEE*

***Abstract*— This paper studies online stochastic optimization where the random parameters follow time-varying distributions. In each time slot, after a control variable is determined, a sample drawn from the current distribution is revealed as feedback information. This form of stochastic optimization has broad applications in online learning and signal processing, where the underlying ground-truth is inherently time-varying, e.g., tracking a moving target. Dynamic optimal points are adopted as the performance benchmark to define the regret, as opposed to the static optimal point used in stochastic optimization with fixed distributions. Stochastic optimization with time-varying distributions is examined and a projected stochastic gradient descent algorithm is presented. An upper bound on its regret is established with respect to the *drift* of the dynamic optima, which measures the variations of the optimal solutions due to the varying distributions. In particular, the algorithm possesses sublinear regret as long as the drift of the optima is sublinear, i.e., the distributions do not vary too drastically. Finally, numerical results are presented to corroborate the efficacy of the proposed algorithm and the derived analytical results.**

***Index Terms*— Stochastic optimization, online optimization, online learning, time-varying distributions, dynamic benchmark**

I. Introduction

Owing to its broad application in machine learning and signal processing, stochastic programming has been studied extensively over the past decade. In stochastic programs, the objective and/or constraint functions are the expectation of some function of the control variables and random variables (e.g., estimated parameters and data samples in machine learning). The challenge is that the underlying distribution of the random variables is often unknown. For instance, in machine learning, the joint distributions of the input feature and label are unknown to the learner and need to be learned. Even when the distribution is known, the expectations involved in the stochastic programs may be hard to evaluate due to the curse of dimensionality. Alternatively, sequential samples of the random variables are often available as feedback information of the distribution, e.g., the sequential data samples in online learning and adptive filtering. In the literature, various stochastic optimization algorithms have been proposed for both unconstrained optimization [1] and constrained optimization [2], [3] to update the control variables in an online manner by making use of the sequential samples.

Most existing works on stochastic programming presume that the underlying distribution of the random varibales is fixed (i.e., does not vary with time) and independent samples are drawn from this common distribution sequentially. In contrast, in many applications, the distributions involved in the stochastic programs may vary slowly across time. For instance, in online learning, the unknown parameters to be estimated may change over time (e.g., tracking a moving target) which leads to time-varying joint distributions of the feature and label. Therefore, we are motivated to study stochastic optimization with time-varying distributions in this paper.

We consider stochastic optimization with time-varying distributions. Since the stochastic optimization problem herein is inherently time-varying, dynamic optimal points are adopted as the performance benchmark, as opposed to the static optima used in stochastic programming with fixed distribution [1]–[6]. In each time slot, after a control variable is chosen, a sample drawn from the current distribution is disclosed. We note that the control variables will not affect the underlying time-varying distributions. Our goal is to devise an algorithm that can track the dynamic optima by leveraging the sequential samples in an online manner so that it is amenable to online learning tasks. To this end, a projected stochastic gradient descent (SGD) algorithm is presented and an upper bound of its regret is developed with respect to the *drift* of the dynamic optimal points, which measures the temporal variations of the underlying distributions. According to this bound, the advocated algorithm possesses sublinear regret as long as the drift of the optima is sublinear, i.e., the distributions do not vary too drastically. Finally, simulation results are presented to corroborate the efficacy of the proposed algorithm and the established theoretical results.

A. Related Work

In the recent decade, stochastic optimization algorithms have garnered much attention, partly due to their pervasive applications in machine learning. For instance, Adam, a gradient-based stochastic optimization algorithm, has become default optimization tool for training deep neural networks [7], and a dual averaging stochastic optimization method has been proposed for online sparse learning in [8]. There is a vast literature on stochastic optimization methods, which can be roughly divided into two categories. The first catogery of methods is sample average approximation (SAA), in which the expected objective or constraint func-

This research was supported in part by the Army Research Office under Grant W911NF-16-1-0448 and in part by the Defense Threat Reduction Agency under Grant HDTRA1-13-1-0029.

X. Cao and H. V. Poor are with the Department of Electrical Engineering, Princeton University, Princeton, NJ. (email: {x.cao, poor}@princeton.edu)

J. Zhang is with the School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ. (email: junshan.zhang@asu.edu)

978-1-5386-7926-5/$31.00 ©2019 AACC

tions are replaced by their sample averages. The convergence of SAA to the optima can be guaranteed if a sufficient number of samples are used [9], [10]. One limitation of SAA is that one needs to wait until a large number of samples are collected. This renders SAA not suitable for online settings, in which control variables are updated in an online fashion as new samples arrive sequentially. Another category of stochastic optimization methods is stochastic approximation (SA), which is amenable to online implementation [11], [12]. SA, originally proposed in [13], replaces exact first-order information, e.g., gradient, by its noisy versions, which give rise to stochastic algorithms such as stochastic gradient descent (SGD). Accelerated SA was developed in [14] by averaging the iterates and using longer step sizes. Additionally, modified SA based on mirror-descent was proposed in [1], [15] and SGD for non-smooth stochastic optimization was examined in [16], [17]. Incremental stochastic subgradient methods were studied in [18] and [19]. Moreover, stochastic composite optimization was examined in [20], [21] where the objective function was composition of two expected-value functions, and regularized stochastic optimization was studied in [8], [22]. Further, constrained stochastic optimization was investigated by using primal methods [3], primal-dual methods [2], random constraint projection [5], [6], virtual-queue based methods [4], and noisy network utility maximization [23]. All the aforementioned works only considered stochastic optimization with fixed distributions. Time-varying distributions of stochastic programs were taken into account by a recent work [24] for unconstrained optimization problems. Nevertheless, the focus of [24] was to select appropriate number of samples per time slot needed to meet a given tracking accuracy, while the number of samples per time slot in this paper is fixed to be one (the extension to arbitrary fixed amount of samples per time slot is straightforward). Besides, the performance criteria used in [24] are mean tracking and high probability tracking, which are very different from the regret in the current paper.

Another line of research related to this paper is online convex optimization (OCO) [25], [26], which has ubiquitous applications in online learning. For example, the well-known Widrow-Hoff algorithm for online linear regression can be regarded as a stochastic optimization problem minimizing the mean square error [26]. The problem of general unconstrained OCO was originally introduced by the seminal work [27] and was later extended to various settings such as constrained OCO [28]–[30] and bandit feedback [31], [32]. Different from this paper, OCO generally considers deterministic and arbitrary (even adversarial) function sequences, which do not conform to any stochastic models.

The remaining part of this paper is organized as follows. In Section II, we develop and analyze an algorithm for online stochastic optimization with time-varying distributions. Simulation results are presented in Section III. We conclude this paper in Section IV.

II. Online Stochastic Optimization with Time-Varying Distributions

In this section, we study online stochastic optimization problem with time-varying distributions. A projected stochastic gradient descent algorithm is presented. Unlike traditional stochastic optimization with fixed distribution, dynamic optimal points are adopted as performance benchmark to define regret. We develop an upper bound of the regret of the presented algorithm in terms of the drift of the dynamic optimal points, which measures the temporal variations of the underlying distributions. In particular, sublinear regret can be guaranteed as long as the drift of the dynamic optima is sublinear, i.e., the distributions of the stochastic optimization do not vary too drastically.

A. Problem Formulation

1) Online Stochastic Optimization with Fixed Distribution: Consider a cost $f(\boldsymbol{x}, \boldsymbol{\theta})$, in which $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^n$ is a control variable and $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^k$ is a random parameter. $\mathcal{X}$ is the set of admissible control variables and Θ is the set of possible realizations of the random parameter. $f : \mathbb{R}^n \times \mathbb{R}^k \mapsto \mathbb{R}$ is a known cost function. In conventional stochastic optimization, the goal is to minimize the expected cost $F(\boldsymbol{x}) := \mathbb{E}_{\boldsymbol{\theta}\sim\mathcal{P}}[f(\boldsymbol{x}, \boldsymbol{\theta})]$, where $\mathcal{P}$ is an unknown yet *fixed* probability distribution over Θ. Feedback information about the underlying distribution $\mathcal{P}$ is available in an online manner [25]. Specifically, in each time t, after the current control variable $\boldsymbol{x}_t \in \mathcal{X}$ is determined (by some algorithm), a sample $\boldsymbol{\theta}_t \sim \mathcal{P}$ will be revealed, which can be used to determine future control variables $\{\boldsymbol{x}_\tau\}_{\tau\geq t+1}$. It is desired that $F(\boldsymbol{x}_t)$, i.e., the expected cost yielded by the algorithm, is not much worse than the (static) optimal expected cost $\inf_{\boldsymbol{x}\in\mathcal{X}} F(\boldsymbol{x})$, e.g., sublinear regret. This form of online stochastic optimization problem has broad applications in online learning, adaptive signal processing, and online resource allocation [25], [26]. For instance, in online learning, the random parameter $\boldsymbol{\theta}$ can be the input features and output labels/numbers, while the sample $\boldsymbol{\theta}_t$ can be the data revealed sequentially. The control variable $\boldsymbol{x}$ can be some unknown parameters (e.g., the weight vector in linear regression and logistic regression) to be estimated.

2) Online Stochastic Optimization with Time-Varying Distributions: One limitation of the aforementioned online stochastic optimization is that the underlying distribution $\mathcal{P}$ of the random parameter is fixed (although unknown). In contrast, in many applications, the distribution $\mathcal{P}$ may vary with time slowly. For example, in adaptive signal processing, the unknown weight vector that one aims to track (e.g., the channel impulse response in wireless communications) usually varies slowly across time. This renders the joint probability distribution of the input regressor and output response time-varying. Thus, we are motivated to consider the following online stochastic optimization problem with time-varying distributions:

$$\underset{\boldsymbol{x}\in\mathcal{X}}{\text{minimize}} \quad F_t(\boldsymbol{x}) := \mathbb{E}_{\boldsymbol{\theta}\sim\mathcal{P}_t}[f(\boldsymbol{x}, \boldsymbol{\theta})], \tag{1}$$

where $\mathcal{P}_t$ is the unknown distribution of the random parameter $\boldsymbol{\theta}$ at time t. The optimal solution of problem (1) for each time t is denoted as $\boldsymbol{x}_t^*$, which is determinstic and time-varying. Denote the control variable selected by an algorithm at time t as $\boldsymbol{x}_t$. In each time t, after $\boldsymbol{x}_t$ is chosen by the algorithm, a sample $\boldsymbol{\theta}_t \sim \mathcal{P}_t$ will be revealed as feedback. All the samples $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, ...$ are assumed to be independent.

Example. The classical adaptive filtering can be posed as an online stochastic optimization problem with time-varying distributions [33]. Consider a standard linear regression signal model as follows:

$$d_t = \widetilde{\boldsymbol{w}}_t^\mathsf{T} \boldsymbol{u}_t + e_t, \tag{2}$$

where $\boldsymbol{u}_t \in \mathbb{R}^n, d_t \in \mathbb{R}, \widetilde{\boldsymbol{w}}_t \in \mathbb{R}^n, e_t \in \mathbb{R}$ are the input regressor, output response, true weight vector, measurement noise at time t, respectively. We assume that $\boldsymbol{u}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ and $e_t \sim \mathcal{N}(0, \sigma^2)$ are independent, where $\boldsymbol{\Sigma} = \operatorname{diag}(\sigma_1^2, ..., \sigma_n^2)$. The true weight vector $\widetilde{\boldsymbol{w}}_t$ is deterministic, unknown and time-varying. At each time t, we receive an observation pair $(d_t, \boldsymbol{u}_t)$ (i.e., $\boldsymbol{\theta}_t$ in the generic optimization formulation), which, according to the signal model in (2), is drawn from the following time-varying distribution:

$$\mathcal{P}_t = \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \widetilde{\boldsymbol{w}}_t^\mathsf{T} \boldsymbol{\Sigma} \widetilde{\boldsymbol{w}}_t + \sigma^2 & \widetilde{\boldsymbol{w}}_t^\mathsf{T} \boldsymbol{\Sigma} \\ \boldsymbol{\Sigma} \widetilde{\boldsymbol{w}}_t & \boldsymbol{\Sigma} \end{bmatrix}\right). \tag{3}$$

The goal of adaptive filtering is to track the time-varying weight vector $\widetilde{\boldsymbol{w}}_t$. To this end, we minimize (with respect to $\boldsymbol{w}$) the following expected fitting error:

$$F_t(\boldsymbol{w}) := \mathbb{E}\left[\left(d_t - \boldsymbol{w}^\mathsf{T} \boldsymbol{u}_t\right)^2\right] = \mathop{\mathbb{E}}_{(d,\boldsymbol{u}) \sim \mathcal{P}_t}\left[\left(d - \boldsymbol{w}^\mathsf{T} \boldsymbol{u}\right)^2\right], \tag{4}$$

which is clearly in the form of the generic optimization problem (1) if one defines $f(\boldsymbol{w}, d, \boldsymbol{u}) = \left(d - \boldsymbol{w}^\mathsf{T} \boldsymbol{u}\right)^2$.

B. Algorithm and Performance Metric

We first make the following standard assumption.

Assumption 1: $\mathcal{X}$ is closed, convex, and bounded by a constant $R > 0$, i.e., $\|\boldsymbol{x}\|_2 \leq R$ for any $\boldsymbol{x} \in \mathcal{X}$.

Under Assumption 1, the projection onto $\mathcal{X}$ exists and is unique. Thus, the projection operator $\Pi_\mathcal{X}(\boldsymbol{y}) = \arg\min_{\boldsymbol{x} \in \mathcal{X}} \|\boldsymbol{x} - \boldsymbol{y}\|_2$ is well defined for any $\boldsymbol{y} \in \mathbb{R}^n$. In the following, we present a projected stochastic gradient descent algorithm to solve problem (1) in an online fashion. In each time t, after the current control variable $\boldsymbol{x}_t$ is chosen and submitted, the algorithm receives a sample $\boldsymbol{\theta}_t \sim \mathcal{P}_t$. Then, a projected stochastic gradient descent step is conducted to obtain the new control variable $\boldsymbol{x}_{t+1}$ as follows:

$$\boldsymbol{x}_{t+1} = \Pi_\mathcal{X}(\boldsymbol{x}_t - \eta \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t)), \tag{5}$$

where $\eta > 0$ is the step size. The algorithm is summarized in Algorithm 1, in which T is the total horizon of the algorithm. Due to the projection onto $\mathcal{X}$ in the update, the control variable $\boldsymbol{x}_t$ is always in the admissible set $\mathcal{X}$ for all t. We note that $\{\boldsymbol{x}_t\}$ is a random sequence since the sample sequence $\{\boldsymbol{\theta}_t\}$ is random.

Algorithm 1 Projected stochastic gradient descent

```
Initialize x_1 ∈ X arbitrarily.
for t = 1, 2, ..., T do
    Submit the action x_t.
    Receive the sample θ_t ~ P_t.
    Update the control variable according to (5) to obtain
    x_{t+1}.
end for
```

The (dynamic) benchmark is the time-varying optimal point $\boldsymbol{x}_t^*$ of problem (1) and it is desired that the control variable $\boldsymbol{x}_t$ yielded by Algorithm 1 is not much worse than $\boldsymbol{x}_t^*$ in terms of minimizing $F_t(\cdot)$. To measure the discrepancy between $F_t(\boldsymbol{x}_t)$ and $F_t(\boldsymbol{x}_t^*)$, we define the (expected) regret at time T as

$$\operatorname{Reg}(T) := \mathbb{E}\left[\sum_{t=1}^{T} F_t(\boldsymbol{x}_t)\right] - \sum_{t=1}^{T} F_t(\boldsymbol{x}_t^*), \tag{6}$$

which is a prevalent performance metric used in the literature. Unlike conventional stochastic optimization with fixed distribution [1]–[6], the regret adopted in this paper is with respect to the dynamic benchmark sequence $\{\boldsymbol{x}_t^*\}$ instead of a static optimum. Clearly, dynamic benchmark is more meaningful (yet more challenging to handle) than static benchmark when the underlying system is intrinsically time-varying. Further, an ideal sequence of control variables $\boldsymbol{x}_t$ should possess sublinear regret, i.e., $\operatorname{Reg}(T) \leq o(T)$. In such a case, as T goes to infinity, $\frac{\operatorname{Reg}(T)}{T} \leq o(1) \to 0$ so that the performance of $\boldsymbol{x}_t$ is no worse than that of the dynamic benchmark $\boldsymbol{x}_t^*$ in terms of asymptotic time-average costs.

Example cont'd. We revisit the aforementioned adaptive filtering example. Suppose we know that the true weight vector $\widetilde{\boldsymbol{w}}_t$ is in the ball $\mathcal{X} = \{\boldsymbol{x} | \|\boldsymbol{x}\|_2 \leq R\}$ for all t. Thus, the stochastic optimization problem at time t is $\min_{w \in \mathcal{X}} F_t(\boldsymbol{w})$, where $F_t(\cdot)$ is defined in (4). Denote the estimated weight vector at time t as $\boldsymbol{w}_t$. Applying the stochastic projected gradient update in (5) to the adaptive filtering problem, we obtain the following update:

$$\boldsymbol{w}_{t+1} = \Pi_\mathcal{X}\left(\boldsymbol{w}_t + 2\eta\left(d_t - \boldsymbol{w}_t^\mathsf{T} \boldsymbol{u}_t\right) \boldsymbol{u}_t\right), \tag{7}$$

which is indeed the well known LMS filter [33]. Additionally, making use of the signal model (2), we can rewrite the expected fitting error as $F_t(\boldsymbol{w}) = (\boldsymbol{w} - \widetilde{\boldsymbol{w}}_t)^\mathsf{T} \boldsymbol{\Sigma} (\boldsymbol{w} - \widetilde{\boldsymbol{w}}_t) + \sigma^2$. Thus, the optimal point at time t is $\boldsymbol{w}_t^* = \widetilde{\boldsymbol{w}}_t$ with optimal value $F_t(\boldsymbol{w}_t^*) = \sigma^2$, and the regret of the LMS filter becomes

$$\operatorname{Reg}(T) = \mathbb{E}\left[\sum_{t=1}^{T} (\boldsymbol{w}_t - \widetilde{\boldsymbol{w}}_t)^\mathsf{T} \boldsymbol{\Sigma} (\boldsymbol{w}_t - \widetilde{\boldsymbol{w}}_t)\right], \tag{8}$$

which measures the discrepancy between the estimates $\{\boldsymbol{w}_t\}$ and the ground truths $\{\widetilde{\boldsymbol{w}}_t\}$. Generally, conventional convergence analysis of LMS or other adaptive filters (e.g., RLS) presumes that the true weight vector $\widetilde{\boldsymbol{w}}_t$ is time-invariant [33]. Though making the analysis tractable, this time-invariance presumption does not hold in most applications of adaptive filters and indeed contradicts the original intention of adaptive filtering, i.e., tracking the *time-varying* unknown weight vector $\widetilde{\boldsymbol{w}}_t$ in real time. Alternatively, in this paper, we admit the temporal variations of $\widetilde{\boldsymbol{w}}_t$ and quantify the impact of the temporal variations on the algorithm performance through regret analysis, which will be elaborated later.

C. Performance Analysis

Before proceeding to the analysis, we further make the following two standard assumptions.

Assumption 2: For any given $\boldsymbol{\theta} \in \Theta$, $f(\boldsymbol{x}, \boldsymbol{\theta})$ is convex in $\boldsymbol{x}$.

Assumption 3: There exists a constant $H > 0$ such that, for any $\boldsymbol{x} \in \mathcal{X}$ and $t = 1, 2, ...$, we have

$$\mathbb{E}_{\boldsymbol{\theta} \sim \mathcal{P}_t} \left[\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}, \boldsymbol{\theta})\|_2^2 \right] \leq H. \tag{9}$$

Assumption 2 is a standard assumption on convexity satisfied by many loss functions used in machine learning and signal processing. For instance, the quadratic loss function in the example of adaptive filtering $f(\boldsymbol{w}, d, \boldsymbol{u}) = \left(d - \boldsymbol{w}^\mathsf{T}\boldsymbol{u}\right)^2$ is convex in $\boldsymbol{w}$ for any given d and $\boldsymbol{u}$. Under Assumption 2, we know that $F_t(\boldsymbol{x})$ is also convex for any t. Assumption 3 requires the cost function f to have bounded expected squared gradients under each distribution $\mathcal{P}_t$. This assumption is also standard in the literature and is satisfied by many commonly used loss functions. For example, in the adaptive filtering example, a choice of H that satisfies Assumption 3 can be given as

$$H = 16R^2 \left[2\sum_{i=1}^{n} \sigma_i^4 + \left(\sum_{i=1}^{n} \sigma_i^2 \right)^2 \right] + 4\sigma^2 \sum_{i=1}^{n} \sigma_i^2. \tag{10}$$

The derivation of this H is given in the Appendix. Further, we define a notion of the *drift* of the dynamic optimal points as follows.

Definition 1: We define the *drift* of the dynamic benchmark sequence $\{\boldsymbol{x}_t^*\}$ by $\Delta(T) := \sum_{t=2}^{T} \|\boldsymbol{x}_{t-1}^* - \boldsymbol{x}_t^*\|_2$. The drift $\Delta(T)$ measures the temporal variations of the distribution $\mathcal{P}_t$, which lead to the temporal variations of the optimal point $\boldsymbol{x}_t^*$ of problem (1). The more drastically the distribution $\mathcal{P}_t$ evolves across time, the larger the drift $\Delta(T)$ is. In addition, we note the following property of the projection operator [34], which is useful later.

Lemma 1: Suppose $\mathcal{S} \subset \mathbb{R}^n$ is closed and convex. Then, for any $\boldsymbol{x} \in \mathcal{S}$ and $\boldsymbol{y} \in \mathbb{R}^n$, we have

$$\|\boldsymbol{x} - \Pi_{\mathcal{S}}(\boldsymbol{y})\|_2 \leq \|\boldsymbol{x} - \boldsymbol{y}\|_2. \tag{11}$$

Now, we are ready to show the main theorem regarding the performance of Algorithm 1.

Theorem 1: Set the step size as $\eta = \sqrt{\frac{\Delta(T)}{T}}$ in Algorithm 1. Then, under Assumptions 1, 2, 3, the regret (defined in (6)) can be upper bounded as follows:

$$Reg(T) \leq \frac{5R^2}{2}\sqrt{\frac{T}{\Delta(T)}} + \left(R + \frac{H}{2}\right)\sqrt{T\Delta(T)} \tag{12}$$

$$= \mathcal{O}\left(\sqrt{T\Delta(T)}\right). \tag{13}$$

Proof: According to Lemma 1, we have

$$\begin{aligned}
&\|\boldsymbol{x}_t^* - \boldsymbol{x}_{t+1}\|_2^2 && (14)\\
&= \|\boldsymbol{x}_t^* - \Pi_{\mathcal{X}}(\boldsymbol{x}_t - \eta\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t))\|_2^2 && (15)\\
&\leq \|\boldsymbol{x}_t^* - (\boldsymbol{x}_t - \eta\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t))\|_2^2 && (16)\\
&= \|\boldsymbol{x}_t - \boldsymbol{x}_t^*\|_2^2 - 2\eta(\boldsymbol{x}_t - \boldsymbol{x}_t^*)^\mathsf{T}\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t) \\
&\quad + \eta^2\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t)\|_2^2. && (17)
\end{aligned}$$

Rearranging terms, we obtain

$$\begin{aligned}
&(\boldsymbol{x}_t - \boldsymbol{x}_t^*)^\mathsf{T}\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t) \\
&\leq \frac{1}{2\eta}\left(\|\boldsymbol{x}_t - \boldsymbol{x}_t^*\|_2^2 - \|\boldsymbol{x}_{t+1} - \boldsymbol{x}_t^*\|_2^2\right) + \frac{\eta}{2}\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t)\|_2^2.
\end{aligned} \tag{18}$$

Since f is convex in $\boldsymbol{x}$ for any given $\boldsymbol{\theta}$, we have

$$f(\boldsymbol{x}_t^*, \boldsymbol{\theta}_t) \geq f(\boldsymbol{x}_t, \boldsymbol{\theta}_t) + \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t)^\mathsf{T}(\boldsymbol{x}_t^* - \boldsymbol{x}_t). \tag{19}$$

Combining (18) and (19), we get

$$\begin{aligned}
&f(\boldsymbol{x}_t, \boldsymbol{\theta}_t) - f(\boldsymbol{x}_t^*, \boldsymbol{\theta}_t) \\
&\leq \frac{1}{2\eta}\left(\|\boldsymbol{x}_t - \boldsymbol{x}_t^*\|_2^2 - \|\boldsymbol{x}_{t+1} - \boldsymbol{x}_t^*\|_2^2\right) + \frac{\eta}{2}\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t)\|_2^2.
\end{aligned} \tag{20}$$

Further, we note that

$$\begin{aligned}
&\sum_{t=1}^{T}\left(\|\boldsymbol{x}_t - \boldsymbol{x}_t^*\|_2^2 - \|\boldsymbol{x}_{t+1} - \boldsymbol{x}_t^*\|_2^2\right) && (21)\\
&\overset{\text{(a)}}{\leq} \|\boldsymbol{x}_1 - \boldsymbol{x}_1^*\|_2^2 + \sum_{t=2}^{T}\left(\|\boldsymbol{x}_t - \boldsymbol{x}_t^*\|_2^2 - \|\boldsymbol{x}_t - \boldsymbol{x}_{t-1}^*\|_2^2\right) && (22)\\
&= \|\boldsymbol{x}_1 - \boldsymbol{x}_1^*\|_2^2 + \sum_{t=2}^{T} 2\boldsymbol{x}_t^\mathsf{T}(\boldsymbol{x}_{t-1}^* - \boldsymbol{x}_t^*) \\
&\quad + \sum_{t=2}^{T}\left(\|\boldsymbol{x}_t^*\|_2^2 - \|\boldsymbol{x}_{t-1}^*\|_2^2\right) && (23)\\
&\overset{\text{(b)}}{\leq} \|\boldsymbol{x}_1 - \boldsymbol{x}_1^*\|_2^2 + \|\boldsymbol{x}_T^*\|_2^2 + 2\sum_{t=2}^{T}\|\boldsymbol{x}_t\|_2\|\boldsymbol{x}_{t-1}^* - \boldsymbol{x}_t^*\|_2 && (24)\\
&\overset{\text{(c)}}{\leq} 5R^2 + 2R\Delta(T), && (25)
\end{aligned}$$

where (a) is by rearranging terms and the fact $\|\boldsymbol{x}_{T+1} - \boldsymbol{x}_T^*\|_2^2 \geq 0$; (b) is due to telescoping sum, $\|\boldsymbol{x}_1\|_2^2 \geq 0$, and Cauchy inequality; (c) is because of $\boldsymbol{x}_t^*, \boldsymbol{x}_t \in \mathcal{X}$ ($\mathcal{X}$ is bounded by R) and the definition of the drift $\Delta(T)$. Summing (20) for t from 1 to T, making use of (25), and taking expectation, we obtain

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}[f(\boldsymbol{x}_t, \boldsymbol{\theta}_t) - f(\boldsymbol{x}_t^*, \boldsymbol{\theta}_t)] \\
&\leq \frac{1}{2\eta}(5R^2 + 2R\Delta(T)) + \frac{\eta}{2}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t)\|_2^2\right].
\end{aligned} \tag{26}$$

We note that $\boldsymbol{x}_t$ and $\boldsymbol{\theta}_t$ are independent according to Algorithm 1. Hence,

$$\mathbb{E}[f(\boldsymbol{x}_t, \boldsymbol{\theta}_t)] = \mathbb{E}_{\boldsymbol{x}_t}[\mathbb{E}_{\boldsymbol{\theta}_t}[f(\boldsymbol{x}_t, \boldsymbol{\theta}_t)|\boldsymbol{x}_t]] \tag{27}$$

$$= \mathbb{E}_{\boldsymbol{x}_t}[\mathbb{E}_{\boldsymbol{\theta}\sim\mathcal{P}_t}[f(\boldsymbol{x}_t, \boldsymbol{\theta})|\boldsymbol{x}_t]] \tag{28}$$

$$= \mathbb{E}[F_t(\boldsymbol{x}_t)]. \tag{29}$$

Moreover, according to Assumption 3, we have

$$\mathbb{E}\left[\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t)\|_2^2\right] = \mathbb{E}_{\boldsymbol{x}_t}\left[\mathbb{E}_{\boldsymbol{\theta}_t}\left[\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta}_t)\|_2^2 | \boldsymbol{x}_t\right]\right] \tag{30}$$

$$= \mathbb{E}_{\boldsymbol{x}_t}\left[\mathbb{E}_{\boldsymbol{\theta}\sim\mathcal{P}_t}\left[\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_t, \boldsymbol{\theta})\|_2^2 | \boldsymbol{x}_t\right]\right] \tag{31}$$

$$\le \mathbb{E}_{\boldsymbol{x}_t}[H] = H. \tag{32}$$

Substituting (29) and (32) into (26) and making use of $\mathbb{E}[f(\boldsymbol{x}_t^*, \boldsymbol{\theta}_t)] = F_t(\boldsymbol{x}_t^*)$, we obtain

$$\sum_{t=1}^{T}(\mathbb{E}[F_t(\boldsymbol{x}_t)] - F_t(\boldsymbol{x}_t^*)) \le \frac{1}{2\eta}\left(5R^2 + 2R\Delta(T)\right) + \frac{\eta H T}{2}. \tag{33}$$

Substituting the step size $\eta = \sqrt{\frac{\Delta(T)}{T}}$, we get

$$\mathrm{Reg}(T) = \sum_{t=1}^{T}(\mathbb{E}[F_t(\boldsymbol{x}_t)] - F_t(\boldsymbol{x}_t^*)) \tag{34}$$

$$\le \frac{5R^2}{2}\sqrt{\frac{T}{\Delta(T)}} + \left(R + \frac{H}{2}\right)\sqrt{T\Delta(T)} \tag{35}$$

$$= \mathcal{O}\left(\sqrt{T\Delta(T)}\right), \tag{36}$$

in which the last step is due to $\Delta(T) \ge \|\boldsymbol{x}_1^* - \boldsymbol{x}_2^*\|_2 = \Omega(1)$, i.e., $\Delta(T)$ is at least of constant order. ■

According to Theorem 1, if the drift $\Delta(T)$ is sublinear, so is the regret of Algorithm 1. In other words, Algorithm 1 possesses sublinear regret as long as the distribution $\mathcal{P}_t$ does not vary too drastically. Conversely, if the drift is allowed to be of linear order, i.e., $\Delta(T) = \Theta(T)$, then no online algorithm can guarantee sublinear regret. To see this, we consider a cost function $f(x, \theta) = (x - \theta)^2$ and action set $\mathcal{X} = [0, 1]$. Thus, the expected cost at time t is

$$F_t(x) = \mathbb{E}_{\theta\sim\mathcal{P}_t}\left[(x-\theta)^2\right] = (x - \mathbb{E}_{\mathcal{P}_t}[\theta])^2 + \mathrm{Var}_{\mathcal{P}_t}[\theta], \tag{37}$$

where $\mathcal{P}_t$ is some distribution over the interval $[0, 1]$. It is easy to check that Assumptions 1, 2, 3 are all satisfied. The optimal point at time t is clearly $x_t^* = \mathbb{E}_{\mathcal{P}_t}[\theta]$. Consider an arbitrary online algorithm and denote its iterate at time t as x_t. Then, its regret is

$$\mathrm{Reg}(T) = \mathbb{E}\left[\sum_{t=1}^{T}(x_t - \mathbb{E}_{\mathcal{P}_t}[\theta])^2\right]. \tag{38}$$

Recall that, in the online setting, only after x_t is determined, a sample from $\mathcal{P}_t$ is revealed. Thus, an adversary can choose $\mathcal{P}_t$ based on the algorithm iterate x_t. Since $\mathcal{P}_t$ is a distribution over $[0, 1]$, $\mathbb{E}_{\mathcal{P}_t}[\theta]$ can achieve arbitrary value in $[0, 1]$. We can choose $\mathcal{P}_t$ such that

$$\left|\mathbb{E}_{\mathcal{P}_t}[\theta] - x_t\right| \ge \frac{1}{4}, \ \left|\mathbb{E}_{\mathcal{P}_t}[\theta] - \mathbb{E}_{\mathcal{P}_{t-1}}[\theta]\right| \ge \frac{1}{4}. \tag{39}$$

Then, from (38), we know $\mathrm{Reg}(T) \ge \frac{T}{16}$, i.e., the regret is not sublinear. Additionally, the drift satisfies $(T-1)/4 \le \Delta(T) \le T - 1$ and is of linear order.

In Theorem 1, the step size is chosen as $\eta = \sqrt{\frac{\Delta(T)}{T}}$, which depends on the drift $\Delta(T)$. We note that $\Delta(T)$ may not be known precisely in advance. Nevertheless, as long as $\eta = \Theta\left(\sqrt{\frac{\Delta(T)}{T}}\right)$, the order bound $\mathcal{O}\left(\sqrt{T\Delta(T)}\right)$ of the regret in Theorem 1 will hold. So, when selecting the step size, one only needs an estimate of the *order* of $\Delta(T)$ with a possible constant factor error. Further, even if such an estimate of the order of $\Delta(T)$ is not available, we can still choose step size η to ensure sublinear regret whenever $\Delta(T)$ is sublinear. We presume that a sublinear upper bound of $\Delta(T)$ is known. That is, we know a sublinear positive sequence $\widetilde{\Delta}(T)$, such that $\Delta(T) \le \widetilde{\Delta}(T)$ for T large enough (there can be a positive constant scaling factor on either side of the inequality, which does not affect order statements). Since $\Delta(T)$ is sublinear, such a sublinear upper bound $\widetilde{\Delta}(T)$ must exist and can be known in advance in many scenarios. If little knowledge about $\Delta(T)$ is known besides sublinearity, we can choose a very conservative sublinear upper bound $\widetilde{\Delta}(T)$, e.g., T^{ζ}, where $\zeta < 1$ is very close to 1. With such a sublinear upper bound $\widetilde{\Delta}(T)$ known, we can choose the step size to be $\eta = \sqrt{\frac{\widetilde{\Delta}(T)}{T}}$. Then, after minor adaption of the proof of Theorem 1, the regret bound becomes $\mathcal{O}\left(\sqrt{T\widetilde{\Delta}(T)}\right)$. Since $\widetilde{\Delta}(T)$ is sublinear, so is the regret. We summarize the above points in the following corollary.

Corollary 1: Suppose we know a sublinear positive sequence $\widetilde{\Delta}(T)$ such that $\Delta(T) \le \widetilde{\Delta}(T)$ for T large enough. Set the step size as $\eta = \sqrt{\frac{\widetilde{\Delta}(T)}{T}}$ in Algorithm 1. Then, under Assumptions 1, 2, 3, there is a sublinear regret bound as follows:

$$Reg(T) \le \mathcal{O}\left(\sqrt{T\widetilde{\Delta}(T)}\right). \tag{40}$$

Example cont'd. We revisit the example of adaptive filtering again. In this case, the drift of the dynamic benchmark is $\Delta(T) = \sum_{t=2}^{T}\|\widetilde{\boldsymbol{w}}_{t-1} - \widetilde{\boldsymbol{w}}_t\|_2$, which measures the temporal variations of the true weight vector $\widetilde{\boldsymbol{w}}_t$. According to Theorem 1, the regret of the LMS given in (8) is upper bounded by $\mathcal{O}\left(\sqrt{T\Delta(T)}\right)$. In particular, if the drift $\Delta(T)$ of the true weight vector is sublinear, so is the regret of the LMS. Conversely, if the drift $\Delta(T)$ is linear or superlinear, the true weight vector $\widetilde{\boldsymbol{w}}_t$ evolves in constant rate at least and virtually no adaptive filter can track it well due to lack of information.

III. Simulation Results

In this section, numerical experiments are conducted to validate the efficacy of the proposed algorithm and the

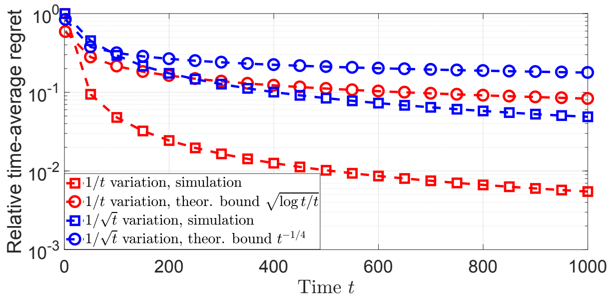

Fig. 1: Relative time-average regret $\frac{\text{Reg}(t)}{t\cdot\text{Reg}(1)}$ for adaptive filtering

theoretical results in Theorems 1.

We apply Algorithm 1 to the adaptive filtering example, which is a stochastic optimization problem with time-varying distributions. The dimension of the true weight vector $\widetilde{\boldsymbol{w}}_t$ is set to be $n = 5$ and the radius of the admissible set $\mathcal{X}$ is $R = 10$. The variance of each entry of the input $\boldsymbol{u}_t$ is 1 and the variance of the noise e_t is 0.5. The sequence of $\widetilde{\boldsymbol{w}}_t$ is generated according to $\widetilde{\boldsymbol{w}}_{t+1} = \Pi_{\mathcal{X}}(\widetilde{\boldsymbol{w}}_t + \boldsymbol{\alpha}_t)$, where each entry of $\boldsymbol{\alpha}_t$ is an independent Gaussian random variable with zero mean and standard deviation $t^{-\beta}$. Here $\beta > 0$ characterizes the temporal variation speed of the distribution $\mathcal{P}_t$ (c.f. (3)) and is related to the drift $\Delta(T)$ of the dynamic optima. The larger β is, the slower $\mathcal{P}_t$ varies and the smaller $\Delta(T)$ is. In fact, we have $\|\widetilde{\boldsymbol{w}}_{t+1} - \widetilde{\boldsymbol{w}}_t\|_2 = \Theta\left(t^{-\beta}\right)$ and $\Delta(T) = \Theta\left(\sum_{t=1}^{T} t^{-\beta}\right)$. In the adaptive filtering example, we consider a slow variation scenario of $\beta = 1$ and a fast variation scenario of $\beta = 0.5$. The orders of $\Delta(T)$ in these two scenarios are $\Theta(\log T)$ and $\Theta(\sqrt{T})$, respectively, which are both sublinear. The total time horizon is chosen to be $T = 1000$. According to Theorem 1, the order of the step size η should be $\Theta\left(\sqrt{\Delta(T)/T}\right)$, which is 0.083 for $\beta = 1$ and 0.1778 for $\beta = 0.5$. Thus, a reasonable choice of the step size is $\eta = 0.1$, which we use for both values of β. The relative time-average regrets $\frac{\text{Reg}(t)}{t\cdot\text{Reg}(1)}$ of Algorithm 1 (the regret in the adaptive filtering example is given by (8)) are shown in Fig. 1 for $\beta = 1$ (i.e., $1/t$ variation) and $\beta = 0.5$ (i.e., $1/\sqrt{t}$ variation). The results are averaged over 1000 independent trials. The theoretical time-average regret (order) bounds (c.f. Theorem 1), i.e., $\mathcal{O}\left(\sqrt{\log t/t}\right)$ for $\beta = 1$ and $\mathcal{O}\left(t^{-1/4}\right)$ for $\beta = 0.5$, are also plotted. We observe that, in accordance with Theorem 1, the time-average regret converges to zero, i.e., the regret is sublinear, for both values of β. Further, for both values of β, the simulated relative time-average regret is smaller than the corresponding theoretical bound, confirming Theorem 1. In addition, the relative time-average regret for $\beta = 1$ is smaller than that for $\beta = 0.5$ by an order of magnitude, which highlights the impact of the temporal variations of the distribution $\mathcal{P}_t$. This impact is reflected in the theoretical regret bound $\mathcal{O}\left(\sqrt{T\Delta(T)}\right)$ in Theorem 1, while it is mostly ignored in the classical analysis of LMS or other adaptive filters [33].

IV. Conclusion

In this paper, we have studied online stochastic optimization problem with time-varying distributions. Due to the temporal variations of the underlying problem, dynamic optimal points have been adopted as performance benchmark to define the regret. A stochastic projected gradient descent algorithm has been presented and an upper bound of its regret has been established in terms of the drift of the dynamic benchmark. Sublinear regret of the algorithm can be ensured as long as the drift of the dynamic optima is sublinear, i.e., the underlying distributions do not vary too drastically across time. Finally, numerical results have been presented to corroborate the efficacy of the proposed algorithm.

Appendix: Derivation of H for Adaptive Filtering

In the example of adaptive filtering, we have

$$\|\nabla_{\boldsymbol{w}} f(\boldsymbol{w}, d, \boldsymbol{u})\|_2^2 = 4\left(d - \boldsymbol{w}^\mathsf{T}\boldsymbol{u}\right)^2 \|\boldsymbol{u}\|_2^2. \tag{41}$$

Thus, for any $\boldsymbol{w} \in \mathcal{X}$, $t = 1, 2, ...,$

$$\begin{aligned}
&\mathbb{E}_{(d,\boldsymbol{u})\sim\mathcal{P}_t}\left[\|\nabla_{\boldsymbol{w}} f(\boldsymbol{w}, d, \boldsymbol{u})\|_2^2\right] && (42)\\
&= 4\mathbb{E}\left[(d_t - \boldsymbol{w}^\mathsf{T}\boldsymbol{u}_t)^2\|\boldsymbol{u}_t\|_2^2\right] && (43)\\
&= 4\mathbb{E}\left[\left(\left(\widetilde{\boldsymbol{w}}_t - \boldsymbol{w}\right)^\mathsf{T}\boldsymbol{u}_t + e_t\right)^2 \|\boldsymbol{u}_t\|_2^2\right] && (44)\\
&\overset{\text{(a)}}{=} 4\mathbb{E}\left[\left(\left(\widetilde{\boldsymbol{w}}_t - \boldsymbol{w}\right)^\mathsf{T}\boldsymbol{u}_t\right)^2 \|\boldsymbol{u}_t\|_2^2\right] + 4\mathbb{E}\left[e_t^2\right]\mathbb{E}\left[\|\boldsymbol{u}_t\|_2^2\right] && (45)\\
&\overset{\text{(b)}}{\le} 4\mathbb{E}\left[\|\widetilde{\boldsymbol{w}}_t - \boldsymbol{w}\|_2^2\|\boldsymbol{u}_t\|_2^4\right] + 4\sigma^2\sum_{i=1}^{n}\sigma_i^2 && (46)\\
&\overset{\text{(c)}}{\le} 16R^2\mathbb{E}[\|\boldsymbol{u}_t\|_2^4] + 4\sigma^2\sum_{i=1}^{n}\sigma_i^2 && (47)\\
&\overset{\text{(d)}}{=} 16R^2\left[2\sum_{i=1}^{n}\sigma_i^4 + \left(\sum_{i=1}^{n}\sigma_i^2\right)^2\right] + 4\sigma^2\sum_{i=1}^{n}\sigma_i^2, && (48)
\end{aligned}$$

where (a) is due to the independence between $\boldsymbol{u}_t$ and e_t; (b) is an application of Cauchy inequality; (c) results from the facts that $\widetilde{\boldsymbol{w}}_t, \boldsymbol{w} \in \mathcal{X}$ and $\mathcal{X}$ is bounded by R; (d) can be derived by using the fourth-order moment of Gaussian random variable $\mathbb{E}\left[u_{t,i}^4\right] = 3\sigma_i^4$.

References

[1] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, "Robust stochastic approximation approach to stochastic programming," *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1574–1609, 2009.

[2] M. Mahdavi, T. Yang, and R. Jin, "Stochastic convex optimization with multiple objectives," in *NIPS*, pp. 1115–1123, 2013.

[3] G. Lan and Z. Zhou, "Algorithms for stochastic optimization with expectation constraints," *arXiv:1604.03887*, 2016.

[4] H. Yu, M. Neely, and X. Wei, "Online convex optimization with stochastic constraints," in *NIPS*, pp. 1427–1437, 2017.

[5] M. Wang and D. P. Bertsekas, "Stochastic first-order methods with random constraint projection," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 681–717, 2016.

[6] M. Wang, Y. Chen, J. Liu, and Y. Gu, "Random multi-constraint projection: Stochastic gradient methods for convex optimization with many constraints," *arXiv:1511.03760*, 2015.

[7] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *ICLR*, 2015.

[8] L. Xiao, "Dual averaging methods for regularized stochastic learning and online optimization," *Journal of Machine Learning Research*, vol. 11, no. Oct, pp. 2543–2596, 2010.
[9] A. J. Kleywegt, A. Shapiro, and T. Homem-de Mello, "The sample average approximation method for stochastic discrete optimization," *SIAM Journal on Optimization*, vol. 12, no. 2, pp. 479–502, 2002.
[10] W. Wang and S. Ahmed, "Sample average approximation of expected value constrained stochastic programs," *Operations Research Letters*, vol. 36, no. 5, pp. 515–519, 2008.
[11] H. Kushner and G. G. Yin, *Stochastic approximation and recursive algorithms and applications*. Springer Science & Business Media, 2003.
[12] A. Benveniste, M. Métivier, and P. Priouret, *Adaptive algorithms and stochastic approximations*. Springer Science & Business Media, 2012.
[13] H. Robbins and S. Monro, "A stochastic approximation method," *Annals of Mathematical Statistics*, pp. 400–407, 1951.
[14] B. T. Polyak and A. B. Juditsky, "Acceleration of stochastic approximation by averaging," *SIAM Journal on Control and Optimization*, vol. 30, no. 4, pp. 838–855, 1992.
[15] A. Nedich and S. Lee, "On stochastic subgradient mirror-descent algorithm with weighted averaging," *SIAM Journal on Optimization*, vol. 24, no. 1, pp. 84–107, 2014.
[16] O. Shamir and T. Zhang, "Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes," in *ICML*, pp. 71–79, 2013.
[17] A. Rakhlin, O. Shamir, K. Sridharan, *et al.*, "Making gradient descent optimal for strongly convex stochastic optimization.," in *ICML*, 2012.
[18] S. S. Ram, A. Nedić, and V. V. Veeravalli, "Incremental stochastic subgradient algorithms for convex optimization," *SIAM Journal on Optimization*, vol. 20, no. 2, pp. 691–717, 2009.
[19] A. Nedich and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization," *SIAM Journal on Optimization*, vol. 12, no. 1, pp. 109–138, 2001.
[20] M. Wang, J. Liu, and E. Fang, "Accelerating stochastic composition optimization," in *NIPS*, pp. 1714–1722, 2016.
[21] M. Wang, E. X. Fang, and H. Liu, "Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions," *Math. Programm.*, vol. 161, no. 1-2, pp. 419–449, 2017.
[22] S. Ghadimi and G. Lan, "Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework," *SIAM Journal on Optimization*, vol. 22, no. 4, pp. 1469–1492, 2012.
[23] J. Zhang, D. Zheng, and M. Chiang, "The impact of stochastic noisy feedback on distributed network utility maximization," *IEEE Transactions on Information Theory*, vol. 54, no. 2, pp. 645–665, 2008.
[24] C. Wilson, V. V. Veeravalli, and A. Nedich, "Adaptive sequential stochastic optimization," *IEEE Transactions on Automatic Control*, 2018.
[25] E. Hazan *et al.*, "Introduction to online convex optimization," *Foundations and Trends® in Optimization*, vol. 2, no. 3-4, pp. 157–325, 2016.
[26] S. Shalev-Shwartz *et al.*, "Online learning and online convex optimization," *Foundations and Trends® in Machine Learning*, vol. 4, no. 2, pp. 107–194, 2012.
[27] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *ICML*, pp. 928–936, 2003.
[28] M. Mahdavi, R. Jin, and T. Yang, "Trading regret for efficiency: online convex optimization with long term constraints," *Journal of Machine Learning Research*, vol. 13, no. Sep, pp. 2503–2528, 2012.
[29] S. Paternain and A. Ribeiro, "Online learning of feasible strategies in unknown environments," *IEEE Transactions on Automatic Control*, vol. 62, no. 6, pp. 2807–2822, 2017.
[30] T. Chen, Q. Ling, and G. B. Giannakis, "An online convex optimization approach to proactive network resource allocation," *IEEE Transactions on Signal Processing*, vol. 65, no. 24, pp. 6350–6364, 2017.
[31] A. D. Flaxman, A. T. Kalai, and H. B. McMahan, "Online convex optimization in the bandit setting: gradient descent without a gradient," in *SODA*, pp. 385–394, 2005.
[32] A. Agarwal, O. Dekel, and L. Xiao, "Optimal algorithms for online convex optimization with multi-point bandit feedback," in *COLT*, pp. 28–40, 2010.
[33] S. S. Haykin, *Adaptive Filter Theory*. Pearson Education India, 2008.
[34] R. T. Rockafellar, *Convex Analysis*. Princeton University Press, 2015.