Network Resource Allocation via Stochastic Subgradient Descent: Convergence Rate

Amrit S. Bedi, Student Member, IEEE and Ketan Rajawat, Member, IEEE

Abstract

This paper considers a general stochastic resource allocation problem that arises widely in wireless networks, cognitive radio, networks, smart-grid communications, and cross-layer design. The problem formulation involves expectations with respect to a collection of random variables with unknown distributions, representing exogenous quantities such as channel gain, user density, or spectrum occupancy. We consider the constant step-size stochastic dual subgradient descent (SDSD) method that has been widely used for online resource allocation in networks. The problem is solved in dual domain which results in a primal resource allocation subproblem at each time instant. The goal here is to characterize the non-asymptotic behavior of such stochastic resource allocations in an almost sure sense. It is well known that with a step size of $\epsilon$, SDSD converges to an $O(\epsilon)$-sized neighborhood of the optimum. In practice however, there exists a trade-off between the rate of convergence and the choice of $\epsilon$. This paper establishes a convergence rate result for the SDSD algorithm that precisely characterizes this trade-off. Towards this end, a novel stochastic bound on the gap between the objective function and the optimum is developed. The asymptotic behavior of the stochastic term is characterized in an almost sure sense, thereby generalizing the existing results for the stochastic subgradient methods. For the stochastic resource allocation problem at hand, the result explicates the rate with which the allocated resources become near-optimal. As an application, the power and user-allocation problem in device-to-device networks is formulated and solved using the SDSD algorithm. Further intuition on the rate results is obtained from the verification of the regularity conditions and accompanying simulation results.

Index Terms
Stochastic subgradient, constant step-size, stochastic resource allocation, D2D communication.

I. INTRODUCTION

Resource allocation is a fundamental problem in economic theory that finds application in the design of wireless communication protocols [1], smart grid systems [2], and scheduling algorithms [3]. From an optimization perspective, the goal is to find the optimal allocation variables such as transmit power, bandwidth, operational schedule, or facility locations, so
as to maximize the user satisfaction, minimize the cost, and satisfy all system constraints. The stochastic resource allocation problem arises in scenarios where the optimization problem includes random parameters with unknown distributions [4]. For such problems, the goal is to find an allocation policy that is feasible and optimal, on average [5] or with high probability [6]. Since the policy variable may be infinite dimensional, the problem is more tractable in the dual domain due to finite number of constraints, an aspect exploited by a number of algorithms; see e.g., [7]–[9] and references therein.

This paper focuses on the so-called online algorithms, where the allocation must occur every time the random parameter is realized and revealed. For each realization, the resource allocation adheres to the operational or “box” constraints, while the overall allocation policy is only asymptotically feasible and optimal. The dual problem may then be solved using the stochastic subgradient descent method, whose asymptotic behavior is well-known [10], [11]. Further justification for solving the problem in the dual domain was provided in [5], [7], where it was shown that such stochastic problems have zero duality gap under some mild conditions. The asymptotic feasibility and optimality of the allocated resources via primal averaging was also established in [8]. In a similar vein, the relationship between the stochastic and ‘averaged’ dual iterates for the power and subcarrier allocation problem in OFDM was established in [9].

In resource allocation problems, it is possible for the environmental variables to change abruptly. This motivates the use of constant step sizes in stochastic algorithms, that obviate the need to restart the iterations whenever such a change occurs [12]. With a constant step size of \( \epsilon \) however, it is well known that the stochastic iterates converge only to an \( O(\epsilon) \)-sized neighborhood of the optimum [8]. On the other hand, making \( \epsilon \) arbitrarily small is also impractical, since it results in a slow convergence rate [13], [14]. The aforementioned trade-off between the rate of convergence and the asymptotic performance of the constant step-size stochastic dual subgradient descent (SDSD) algorithm is an important aspect that has not been studied explicitly.

The goal of this paper is to rigorously characterize the convergence rate of the SDSD algorithm in an almost sure sense. The key contribution of the paper is the development of stochastic bounds on the iterates produced by SDSD method, that explicate the role played by \( \epsilon \) in forgetting the initial conditions, and coming close to the optimum. To this end, the iterations are divided into epochs of duration \( 1/\epsilon \), and the optimality gap is analyzed for both fixed and arbitrarily small \( \epsilon \). The main result of the paper is that the stochastic component of this gap goes to zero almost surely, either as the number of epochs go to infinity with fixed \( \epsilon > 0 \), or as \( \epsilon \) itself
goes to zero. The bounds developed here specialize to the known asymptotic results, and are generally applicable to any problem solved via the SDSD method. To the best of our knowledge, these are also the first such convergence rate results for the constant step size SDSD algorithm. Corresponding results for the diminishing step size stochastic subgradient exist, but cannot be readily extended to the present case [15]. Likewise, the analysis in [16] can be extended to yield rate results that hold on average, but does not yield almost sure bounds. The analysis in the present work makes use of the strong law of large numbers directly, and is completely different from that in [15], [16].

As the second contribution, it is shown that the convergence rate results are readily applicable to the stochastic resource allocation problem of interest here. To further demonstrate the usefulness of the bounds, the paper details a contemporary application that uses mobile caching for improving service via device-to-device (D2D) communication [17], [18]. To this end, we consider the D2D edge caching framework where willing users offer data connectivity to highly mobile users experiencing spotty coverage. The problem is well-motivated for vehicular users who may download data from other users residing near the highway. The corresponding resource allocation problem is shown to satisfy the required regularity conditions, thereby demonstrating the flexibility afforded by the SDSD algorithm.

This paper is organized as follows. Sec. I-A lists some of the related work in this area, providing context to the current work. Sec. II starts with detailing the D2D edge caching problem and formulates the general network resource allocation problem. Sec. III discusses the various solution methodologies in the literature, including the stochastic subgradient descent (SSD) framework. Sec. IV provides the main results of the paper, stating the convergence rate results for both primal and dual problems. Sec. V further develops the D2D examples introduced in Sec. II, and verifies the different conditions required for the convergence results to hold. Simulation results on D2D example are provided in Sec. VI and Sec. VII concludes the paper.

The notation used here is as follows. Boldface letters denote column vectors, for which the inequalities and equalities are defined component-wise. The set of all real numbers is denoted by $\mathbb{R}$, and likewise the sets of non-negative reals, positive reals, and $K$-dimensional real vectors are denoted by $\mathbb{R}_+$, $\mathbb{R}_{++}$, and $\mathbb{R}^K$, respectively. Time indices are denoted by the subscripts $t$ and $\tau$. For a vector $\mathbf{x}$, $[\mathbf{x}]_i$ denotes its $i$-th entry, $\|\mathbf{x}\|$ denotes its $\ell_2$ norm, $\|\mathbf{x}\|_p$ denotes its $p$-th norm, for $p \in \mathbb{R}_+$, and $\mathbf{x}^T$ denotes the transposed row vector. The expectation operator is denoted by $\mathbb{E}[\cdot]$ and the inner product is denoted by $\langle \cdot, \cdot \rangle$. Finally, $[c]_b^a = \min(\max\{c, b\}, a)$ and
\[ [c]_+ := [c]_0^\infty. \]

A. Related work

Stochastic approximation algorithms have a long history, going back to the prototypical adaptive filtering algorithms by Robbins and Monro [19] and Widrow and Stearns [20], and have been studied extensively in the context of least mean square (LMS) and recursive least-squares (RLS) algorithms [21]. Stochastic gradient and subgradient methods have since been applied to neural networks [22], parameter tracking [23], large-scale machine learning [15], [24], and resource allocation problems [4]. Convergence of these algorithms is well known for various choices of the step-size parameter [25]. Convergence rate of the stochastic subgradient descent algorithm has been established for diminishing step size rules via non-asymptotic analysis [15]. However, not much is known about the convergence rate of the constant step-size counterpart, except for the fact that it exhibits linear convergence when far from the optimum, if the objective function is strongly convex [10]. The rate analysis presented here fills this gap for a class of convex problems that satisfy certain regularity conditions; see Sec. IV.

The use of dual subgradient algorithms for deterministic resource allocation was first popularized in [26]. Recovery of near-optimum allocations via primal averaging was proposed in [27], and the result was extended to stochastic resource allocation problems in [8], [28]. The corresponding convergence rate analysis for primal recovery was provided in [16], which also serves as a starting point for the analysis presented here. Note however that the extension of the rate results to stochastic problems is not trivial, and does not follow immediately from the result in [16]. The specific assumptions required to develop the bounds in this paper are inspired from those used in the context of stochastic approximation and averaging [29].

From a broader perspective, the work in this paper is also related to the backpressure algorithm, first proposed in the context of stochastic network optimization [30]. As shown in [31], the dual subgradient algorithm when applied to deterministic resource allocation problems, may also be viewed as the so-called drift-plus-penalty algorithm. The analysis in [31] however does not translate to convergence rate results for the SDSD algorithm.

The wireless caching framework utilizing D2D communications was first proposed in [17], [18], where the fundamental limits were analyzed. The system model described here builds upon the basic framework of [18] by formulating the problem within the resource allocation fabric, and adding some implementation details. The results presented here may also be applied to other stochastic resource allocation formulations, such as those in broadcast power allocation [4],
OFDM [9], beamforming [32], cognitive radio networks [6], network utility maximization [7], [8], [33], demand-response in the smart grid [34], smart grid powered green communications [35], [36] and energy harvesting [37].

II. Problem Formulation

This section formulates the general stochastic network resource allocation problem. We begin with detailing a D2D caching example that is used to motivate the general problem.

A. Motivation: D2D Mobile Caching

The D2D framework enables direct communication between nearby user equipments (UE), enabling greater spectrum utilization, higher energy efficiency, and increased overall throughput. The technology also allows unique solutions to connectivity problems that arise at the network edge or as a result of cellular congestion at overcrowded events. As an example, the D2D architecture proposed in [18] considers caching of popular content on mobile devices with excess storage. The content files are then available for download over a D2D link, allowing users to reach higher data rates, avoid congestion, and overcome coverage issues. By directly involving the smartphone equipped users into the process of content distribution, such an edge-caching solution not only cuts down the hardware provisioning costs but also promises better user experience.

This example builds upon the mobile caching framework proposed in [18]. Specifically, the mobile user equipment (MoUE) seeks to download a large file or stream media for a sufficiently long duration, while maintaining a reasonable download rate or quality of experience. Let $\mathcal{M} = \{1, \ldots, M\}$ be the set of mobile caches in the network and at time $t$, the requested chunk be available at $\mathcal{M}_t \subset \mathcal{M}$ unique mobile caches (devices) that are at close proximity to the user. The potential download rate $R_i(p^i_t, \gamma^i_t)$ depends on the power allocation $p^i_t$ at the $i$-th mobile cache, as well as on the channel gain $\gamma^i_t$ of the D2D links. The downloads also incur a cost $c^i_t \in \mathbb{R}_{++}$ per unit of transmit power $p^i_t$ for slot $t$. The costs could be in form of incentives provided to the mobile caches by the content delivery network (CDN) company, and/or directly charged to the MoUE in form of an “enhanced coverage” fee. At each time $t$, the MoUE selects a cache $i_t$ to download from, and obtains an average throughput of $r$ over time. Finally, the user satisfaction for the achieved average throughput $r$ is quantified through the concave utility function $U(r)$. Fig. 1 depicts an example scenario, where a MoUE connects to different UEs in order to download cached data, that would otherwise be available only from the base stations.

The resulting stochastic resource allocation problem is formulated as
Fig. 1: System model for mobile caching in D2D networks.

\[
\begin{align*}
\max_{r, \{p^t_i\}} & \quad U(r) - \mathbb{E} \left[ \sum_{i \in \mathcal{M}_t} c_i^t p^t_i \right] \\
\text{s. t.} & \quad r \leq \mathbb{E} \left[ \sum_{i \in \mathcal{M}_t} R_i(p^t_i, \gamma_i^t) \right] \\
\{p^t_i\}_{i \in \mathcal{M}} \in \mathcal{P}, & \quad r_{\text{min}} \leq r \leq r_{\text{max}}
\end{align*}
\] (1)

where the expectations are with respect to the random variables \(\varphi_t := (\mathcal{M}_t, \{c_i^t\}_{i \in \mathcal{M}_t}, \{\gamma_i^t\}_{i \in \mathcal{M}_t})\). The optimization variables consist of the power allocation function \(p^t_i\) and the rate variable \(r\). The formulation of (1) follows the classical “utility minus penalty” maximization format common to network resource allocation problems [33]. The set of functions \(\mathcal{P}\) is such that only one out of \(\{p^t_i\}_{i \in \mathcal{M}}\) is non-zero for each \(t\) (cf. Sec. V). Consequently, the summations in (1a) and (1b) consist of only one term for each \(t\). The set \(\mathcal{P}\) also specifies the maximum and minimum values of \(p^t_i\) for each \(t\). Finally, the constraint in (1b) ensures that the power allocated per-time slot is sufficient to satisfy the average rate requirement.

The specific form of the rate function depends on the wireless technology used by the users. For instance, under slow fading scenarios, the power allocation and user selection can occur every coherence interval. Since channel state information can be acquired easily, the users may employ adaptive modulation and coding in order to achieve a rate close to the ergodic capacity. Specifically, for the mobile device \(i\), the potential transmission rate is of the form \(R_i(p^t_i, \gamma_i^t) := W \log_2(1 + p^t_i \gamma_i^t / \alpha)\) where \(W\) is the bandwidth of the channel and \(\alpha\) includes the effect of noise and interference as well as other impairments, such as the use of finite block length codes and imperfect channel state information at the transmitter.

More realistically, under fast fading scenarios, the power allocation must occur over intervals
that are significantly longer than the coherence time. In this case, it is more reasonable to consider the average rate over several coherence intervals as $R_i(p_i^t, \gamma_i^t) := W \log_2(1 + p_i^t \gamma_i^t h_i / \alpha)$ where $h_i$ is the small-scale fading gain [38]. In this case, the power allocation and user selection occur only on the basis of the average channel gain $\gamma_i^t$, which changes slowly. It is remarked that the system model considered here allows other forms of the rate function as well.

B. Stochastic Resource Allocation

This section considers the more general stochastic resource allocation problem where the formulation involves expectations with respect to a collection of $q$ random variables with unknown distributions, denoted by $\varphi \in \mathbb{R}^q$. Of particular interest are the problems arising in the context of wireless communications and networks, where $\varphi$ captures the state of the system, and the formulation takes the form [6]–[9]

$$\begin{align*}
(P_1) \quad (x^*, p^*) &= \arg \max f_0(x) \\
\text{s. t. } u(x) + \mathbb{E}[v(\varphi, p_{\varphi})] &\geq 0, \\
x &\in \mathcal{X}, \ p \in \mathcal{P}.
\end{align*}$$

The optimization variables in $(P_1)$ comprise of the resource allocation variable $x \in \mathbb{R}^d$ and the policy functions $p : \mathbb{R}^q \rightarrow \mathbb{R}^p$. The objective function $f_0 : \mathbb{R}^d \rightarrow \mathbb{R}$ is a concave function, while the set $\mathcal{X}$ is compact and convex. The vector-valued constraint function is defined as $u(x) := [u_1(x) \cdots u_K(x)]^T$, where $\{u_i(x) : \mathbb{R}^d \rightarrow \mathbb{R}\}_{i=1}^K$ are concave functions. In contrast, no such restriction is placed on the vector-valued function $v : \mathbb{R}^q \times \mathbb{R}^p \rightarrow \mathbb{R}^K$ and the compact set of functions $\mathcal{P}$. The rate analysis in Sec. IV however does require the overall problem to satisfy certain regularity properties, such as Slater’s constraint qualification and differentiability of the subgradient error; see (A1)–(A4).

It can be seen that the D2D edge caching problem in (1) is a special case of (2)-(4). Introducing a scalar variable $z \in \mathbb{R}$, it is possible to write (1) equivalently as

$$\begin{align*}
\max_{r, \{p_i^t\}} \quad &U(r) + z \\
\text{s. t. } \quad &-z - \mathbb{E} \left[ \sum_{i \in \mathcal{M}_t} c_i^t p_i^t \right] \geq 0 \quad (5a) \\
&-r + \mathbb{E} \left[ \sum_{i \in \mathcal{M}_t} R_i(p_i^t, \gamma_i^t) \right] \geq 0 \quad (5b) \\
&\{p_i^t\}_{i \in \mathcal{M}} \in \mathcal{P}, \quad r_{\text{min}} \leq r \leq r_{\text{max}} \quad (5d)
\end{align*}$$
Comparing (5) with (2)-(4), we see that \( \mathbf{x} = [r, z]^T \) and \( \varphi_t := (\mathcal{M}_t, \{c_i^t\}_{i \in \mathcal{M}_t}, \{\gamma_i^t\}_{i \in \mathcal{M}_t}) \). Likewise the forms of vector functions \( \mathbf{u} \) and \( \mathbf{v} \) can be readily inferred.

Since the distribution of \( \varphi \) is also not known in advance, it is generally not possible to solve \((P_1)\) in an offline manner. The goal here is to solve \((P_1)\) in an online fashion by observing the realizations of the independent identically distributed (i.i.d.) process \( \{\varphi_t\}_{t \in \mathbb{N}_0} \), where \( \mathbb{N}_0 \) is the set of non-negative integers. For most problems of interest, such a framework also entails online allocation of resources for each time \( t \). To this end, the class of algorithms considered here will output the sequence of vector pairs \( \{x_t, \tilde{p}_t\} \) for each \( t \), for the purpose of allocating resources. For the sake of brevity, we will subsequently denote policy function \( p_t := p_{\varphi_t} \) and \( s_t(x, p_t) := u(x) + v(\varphi_t, p_{\varphi_t}) \), so that (3) can equivalently be written as \( \mathbb{E}[s_t(x, p_t)] \geq 0 \). Here, it is understood that the expectation is with respect to the random vector \( \varphi_t \). Having introduced the problem at hand, we detail an example formulation in the context of D2D mobile caching.

III. Solution via Dual Descent

This section details the SDSD algorithm for solving \((P_1)\) in an online fashion. To this end, the basic assumptions are first stated (Sec. III-A), followed by the SDSD algorithm (Sec. III-B), and a discussion of the known results (Sec. III-C).

A. Basic assumptions

The following assumptions are commonly utilized by different dual algorithms proposed in the literature. None of these assumptions are too restrictive, and they can be easily verified for most resource allocation problems of interest.

A1. Non-atomic probability density function: The random variable \( \varphi_t \) has a non-atomic probability density function (pdf).

A2. Slater’s condition: There exists strictly feasible \( (\bar{x}, \bar{p}) \), i.e., \( \mathbb{E}[s_t(\bar{x}, \bar{p}_t)] > 0 \).

A3. Bounded subgradients: The function \( s_t(\cdot, \cdot) \) takes bounded values, i.e., there exists a constant \( G < \infty \) such that \( \|s_t(\cdot, \cdot)\| \leq G \) for all \( t \in \mathbb{N}_0 \).

In (A1), for \( \varphi_t \) to have a non-atomic pdf, it should not have any point masses or delta functions. Note that this requirement is not restrictive for a number of applications arising in wireless communications; see e.g. [8]. The Slater’s condition is also not restrictive, since a strictly feasible resource allocation can often be found for most real-world problems; see Sec. V, [7], [8] for examples. Finally, the bound in (A3) also holds for most resource allocation problems where the functions \( s_t(\cdot, \cdot) \) represent natural quantities such as instantaneous rate (cf. (1b)), indicator
function for channel outage [6], or household power consumption [39]. Having introduced the basic assumptions, we are ready to state the SDSD algorithm.

B. The stochastic dual subgradient algorithm

Towards solving \((P_1)\), consider the more tractable dual formulation, which has a finite number of optimization variables. Introducing a dual variable \(\lambda \in \mathbb{R}^K_+\) corresponding to the constraint (3), the Lagrangian is given by

\[
L(\lambda, x, p) = f_0(x) + \left\langle \lambda, \mathbb{E} [s_t(x, p_t)] \right\rangle
\]

(6)

where the constraints in (4) are kept implicit. The dual function is obtained by maximizing \(L(\lambda, x, p)\) subject to (4), that is,

\[
g(\lambda) = \max_{x \in X, p \in P} L(\lambda, x, p).
\]

(7)

Finally, the dual problem of \((P_1)\) is given by

\[
D = \min_{\lambda \geq 0} g(\lambda).
\]

(8)

In general, since \((P_1)\) may be non-convex, it holds that \(D \geq P\). It was shown in [7, Prop. 6] however, that under \((A1)-(A3)\), it holds that \(P = D\). The proof utilizes the Lyapunov convexity theorem, and holds even if at least one element of \(\varphi_t\) has an absolutely continuous cumulative distribution function (cdf) [40]. It is remarked that Lyapunov convexity has previously yielded similar results in control theory [41], economics [42], and wireless communications [5], [43].

The result on zero duality gap legitimizes the dual descent approach, since the dual problem is always convex, and the resultant dual solution can be used for primal recovery. To this end, similar problems in various contexts have been solved via the classical dual descent algorithm [5]–[7], [28], wherein the primal updates utilize various sampling techniques.

This paper considers the ergodic stochastic optimization (ESO) algorithm proposed in [8] for a similar problem\(^1\). Applied to \((P_1)\), the ESO algorithm starts with an arbitrary \(\lambda_0\), and utilizes the following iterations for \(t \in \mathbb{N}_0\),

\[
(x_t(\lambda_t), p_t(\lambda_t)) = \arg\max_{x \in X, p \in \Pi_t} f_0(x) + \left\langle \lambda_t, s_t(x, p) \right\rangle
\]

(9a)

\[
\lambda_{t+1} = [\lambda_t - \epsilon s_t(x_t(\lambda_t), p_t(\lambda_t))]_+.
\]

(9b)

Here, \(\Pi_t := \{p_{\varphi_t} \in \mathbb{R}^p | p \in \mathcal{P}\}\) is the set of all legitimate values of the vector \(p_{\varphi_t}\) and \(\bar{p}\) denotes all feasible vectors \(p_t(\lambda_t)\). The ESO algorithm is motivated from the fact that for any

\(^1\)The ESO algorithm is a stochastic dual subgradient descent algorithm applied to a resource allocation problem in [8].
\( \lambda \in \mathbb{R}_+^K \), \( s_t(x_t(\lambda), p_t(\lambda)) \) is a stochastic subgradient of the dual function \( g(\lambda) \). Consequently, the updates in (9) amount to solving (8) via the SSD algorithm with a constant step-size. The use of a constant step size is motivated from classical short memory adaptive algorithms such as the least mean squares algorithm. As stated earlier, the constant step-size algorithms can even handle abrupt changes in the problem parameters, without being restarted.

This paper considers the projected variant of the SSD algorithm for dual updates. Specifically, the updates in (9b) are projected on to a compact set \( \mathcal{L} \subset \mathbb{R}_+^K \), and take the following form

\[
\lambda_{t+1} = P_{\mathcal{L}}(\lambda_t - \epsilon s_t(x_t(\lambda_t), p_t(\lambda_t)))
\]

where for any \( \lambda \in \mathbb{R}^K \),

\[
[P_{\mathcal{L}}(\lambda)]_i = \begin{cases} 
0 & [\lambda]_i < 0 \\
\lambda_{\text{max}} & [\lambda]_i > \lambda_{\text{max}} \\
\lambda_i & 0 \leq [\lambda]_i \leq \lambda_{\text{max}} 
\end{cases}
\]

for \( 1 \leq i \leq K \). In other words, large values of \( [\lambda]_i \) are truncated to \( \lambda_{\text{max}} \), where \( \lambda_{\text{max}} \gg \|\lambda^*\|_\infty \). Such a modification is already applicable to any practical implementation of the SSD algorithm, where \( \lambda_t \) is not allowed to take arbitrarily large values. Since \( \lambda^* \) is not known in advance, a bound on \( \|\lambda^*\|_\infty \) is derived in Appendix A using (A2). Consequently, the following rule can be used for choosing \( \lambda_{\text{max}} \) in practice:

\[
\lambda_{\text{max}} \gg \frac{1}{\chi(\tilde{x}, \tilde{p})}(g(\tilde{\lambda}) - f_0(\tilde{x})) \geq \frac{(g(\tilde{\lambda}) - f_0(\tilde{x}))}{G} \tag{12}
\]

where \( \tilde{\lambda} \in \mathbb{R}_+^K \), \((\tilde{x}, \tilde{p})\) is a strictly feasible solution to (P1) (cf. (A2)), \( G \) is the subgradient bound (cf. A3) and \( \chi(\tilde{x}, \tilde{p}) := \min_{1 \leq k \leq K} \mathbb{E}[s_t(\tilde{x}, \tilde{p}_t)]_k \). In general, the quantity \( \chi(\tilde{x}, \tilde{p}) \) may be calculated empirically. However, for many problems of interest, a bound on \( \lambda_{\text{max}} \) may arise naturally (cf. Sec. VI). The projected SSD proposed in (10) ensures that the iterates \( \lambda_t \) stay bounded for all \( t \in \mathbb{N}_0 \). The boundedness condition is required for carrying out the rate analysis in Sec. IV.

C. Known results

The asymptotic properties of the SDSD algorithm with constant step-size are well-known [10], [11]. Asymptotic convergence results for the ESO algorithm, applied to slightly different resource allocation problem, were established in [8]. The results in [8] can readily be extended to (P1) solved via projected SDSD, and take following form:

\[
\lim_{t \to \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} s_r(x_r(\lambda_r), p_r(\lambda_r)) \geq 0 \quad \text{a. s.} \tag{13a}
\]
\[
\lim_{t \to \infty} f(\bar{x}_t) \geq P - (\epsilon \bar{G}^2 / 2)
\]  

where the running average \( \bar{x}_t \) is defined as \( \bar{x}_t = \frac{1}{t} \sum_{r=0}^{t-1} x_r(\lambda_r) \) and \( \bar{G}^2 \) is the bound on \( \mathbb{E} \left[ \|s_t(x_t(\lambda_t), p_t(\lambda_t))\|^2 \right] \). An important feature of the stochastic algorithm is that the primal updates in (9a) can be used for allocating resources in real-time. Further, such allocations will be asymptotically feasible and near-optimal for almost every realization of the random process \( \{\varphi_t\}_{t \in \mathbb{N}_0} \).

IV. CONVERGENCE RATE RESULTS

This section develops various results regarding the rate of convergence of the SDSD algorithm. In contrast to the asymptotic results in (13), the goal here is to quantify the rate at which the allocations specified by (9a) become optimal. Such results are of practical significance to the protocol designers, since they can be used to estimate the number of iterations required for the primal and dual objectives to be near-optimal. In the case of the constant step-size SDSD, the convergence rate also depends on the step-size parameter \( \epsilon \). For instance, it is well-known that the choice \( \epsilon \to 0 \), motivated from the result in (13b), leads to slow convergence in all constant step-size (sub-)gradient descent algorithms. The results presented here provide a precise characterization of the trade-off between \( \epsilon \) and the convergence rate for the SDSD algorithm.

As in [8], the results in this section make use of the strong law of large numbers, and thus hold for almost every realization of the i.i.d. process \( \{\varphi_t\}_{t \in \mathbb{N}_0} \). It is emphasized that the analysis presented here is quite different from the standard convergence analysis carried out for SSD algorithm and its variants [10], [11], [22], [24]. It is also different from the non-asymptotic analysis for the case of diminishing step-size SSD algorithms, that only applies to ensemble averages [15]. Furthermore, the rate results presented in [15] apply only to the unconstrained stochastic subgradient algorithm, and cannot be extended to constrained problems (cf. (P1)) or to the projected subgradient algorithm (cf. (10)).

The results are first developed for the general SSD algorithm (Sec. IV-A), and subsequently specialized to the resource allocation problem at hand (Sec. IV-B).

A. Convergence rate for the SSD algorithm

This section considers the generic optimization problem
\[
\lambda^* = \arg \min_{\lambda \in \Lambda} g(\lambda)
\]

where \( \Lambda \subset \mathbb{R}^K \) is a closed, compact, and convex set, and \( \max_{\lambda \in \Lambda} \|\lambda - \lambda^*\| \leq \Lambda_{\text{max}} < \infty \). Similar to (8), the optimum function value is denoted by \( D = g(\lambda^*) \). Given \( \lambda \in \Lambda \), let \( f(\lambda) \in \mathbb{R} \).
∂g(λ) be a subgradient of g(λ). Similar to (P_1), let f_t(λ) := f_{φ_t}(λ) for all t ∈ N_0 be the corresponding stochastic subgradients that depend on the i.i.d. process \{φ_t\}_{t∈N_0} and satisfy E[f_t(λ)] = f(λ) for any λ ∈ Λ. For instance, in the simplest case, the stochastic gradient could be of the form f_t(λ) = f(λ) + ζ_t, where ζ_t is a zero mean i.i.d. random variable. The optimization problem (14) is solved via the projected SSD algorithm.

\[
λ_{t+1} = P_Λ (λ_t - ef_t(λ_t))
\]

where P_Λ (·) denotes the projection operation. The algorithm is initialized with an arbitrary λ_0 ∈ Λ such that B_0 := ∥λ_0 − λ⋆∥ < ∞. Next, we make certain assumptions specific to (14). To this end, define the stochastic error e_t(λ) := f_t(λ) − f(λ), and observe that for any λ ∈ Λ, the sequence \{e_t(λ)\} is also i.i.d.

**A3' Bounded subgradients:** There exists constant G < ∞ such that ∥f_t(λ)∥ ≤ G for all λ ∈ Λ.

**A4. Continuously differentiable error:** The error function e_t(λ) is continuously differentiable on Λ, and the gradient with respect to λ satisfies ∥∇λe_t(λ)∥ < G_e < ∞.

The requirement for bounded subgradient in (A3') is analogous to that in (A3) for (P_1). Here, (A3') is stated separately because the problem in (14) is more general than the dual of (2)-(4). In practice, applying the results of this section to (8) entails substituting f_t(λ) = s_t(x(λ), p_t(λ)) for all t ∈ N_0, which makes (A3') the same as (A3). The error function e_t(λ) may not always be continuous or differentiable for the problem at hand, and the same must be verified explicitly. It is emphasized that (A4) need only be checked for e_t(λ) and not for f_t(λ), which is still allowed to be non-differentiable; see Sec. V.

As an example, consider the class of problems where the convex objective function takes the form g(λ) = E[ℓ_t(λ)] + r(λ), where ℓ_t(λ) is a twice-differentiable loss function that depends on the ‘data index’ t, and r(λ) is a possibly non-differentiable regularizer. For such problems, the error function becomes e_t(λ) = ∇λℓ_t(λ) − E[∇λℓ_t(λ)], which is clearly differentiable. Further, the ‘loss-plus-regularizer’ problem structure is quite general, and includes well-known formulations such as LASSO [24] and nuclear norm regularized matrix least squares [44]. Specifically, given regressors \{y_t\} and regressors \{x_t\}, the objective function in the LASSO formulation takes the form \[\sum_t (y_t - ⟨x_t, λ⟩) + ∥λ∥_1\] and thus adheres to (A4). The first result is regarding the objective function values obtained from (15), and holds for all T.
Theorem 1. Under (A1)-(A4) and for $T = n/\epsilon$, the minimum dual function value is bounded as
\begin{equation}
\min_{0 \leq \tau \leq T-1} g(\lambda_{\tau}) \leq \frac{1}{T} \sum_{\tau=0}^{T-1} g(\lambda_{\tau}) \leq D + \frac{B_0}{2n} + \frac{\epsilon G^2}{2} + C_T(n, \epsilon) \tag{16}
\end{equation}
where the random variable $C_T(n, \epsilon)$ holds for $\zeta > 0$,
\begin{align*}
\epsilon^{\zeta-1/2} C_T(n, \epsilon) &\to 0 \quad \text{a.s. as } \epsilon \to 0 \text{ for fixed } n < \infty \quad (17a) \\
n^{1/2-\zeta} C_T(n, \epsilon) &\to 0 \quad \text{a.s. as } n \to \infty \text{ for fixed } \epsilon > 0. \quad (17b) \\
\mathbb{P}(C_t(n, \epsilon) > \nu^{\zeta-1/2}) &< A \exp (-n^{2\zeta}) \quad (17c)
\end{align*}
where $\nu := \max\{\frac{1}{\epsilon}, n\}$ and $A < \infty$ is a constant that does not depend on $n$ or $\epsilon$.

It is remarked that since $g(\cdot)$ is convex, the bound in (16) also holds for $g(\bar{\lambda}_t)$, where $\bar{\lambda}_t := \frac{1}{t} \sum_{\tau=0}^{t-1} \lambda_{\tau}$ is the running average of the iterates. Theorem 1 characterizes the manner in which the minimum objective function value approaches $D$ for large $t$. Of the three terms in this optimality gap, the first one depends on the initialization and decays as $O(1/n)$. The second term depends on the subgradient bound, and decays linearly with the step-size $\epsilon$. Finally, the third term is random, and decays almost surely as $O((\epsilon/n)^{1/2-\zeta})$ for any $\zeta > 0$ (cf. (17)). Alternatively, the probability of the third term being non-zero decays exponentially as either $n \to \infty$ or $\epsilon \to 0$ (cf. (17c)). Indeed, for a given run of (15) with a fixed $\epsilon$, the probability of the third term being non-negligible starts to decrease only beyond $n > 1/\epsilon$ or equivalently, $T > 1/\epsilon^2$.

Further intuition on the convergence rate can be obtained by considering the two cases in (17). When $\epsilon > 0$ is fixed, it can be seen that the asymptotic results in [8], [10], [11] follow directly from Theorem 1 as $n \to \infty$. That is, while the initial condition is “forgotten” for $t \gg 1/\epsilon$, the optimality gap does not necessarily approach zero, but is eventually bounded by $\epsilon G^2/2$. At the same time, the fluctuations due to the stochastic term subside exponentially fast; see (17c).

On the other extreme, consider the case when $n$ is kept fixed, while the algorithm is run for different values of $\epsilon$. For the scenarios when $\epsilon$ is arbitrarily small, the asymptotic optimality gap is clearly negligible. However, for such small step-sizes, the algorithm takes a long time to forget the initial conditions, since the first term decays only as $O(1/t)$. Consequently, for all runs when $\epsilon$ is taken to be small, the algorithm will appear to converge slowly. Likewise, the probability of the stochastic term being non-negligible starts to decrease exponentially only for $T > 1/\epsilon^2$ (cf. (17c)). It is remarked that such a trade-off also applies to the classical subgradient method [16], and the result in Theorem 1 can be viewed as its stochastic counterpart. It is remarked that the
results in [16] can be readily obtained by taking expectation on both sides of (16) since we have that \( \mathbb{E} [C_t(n, \epsilon)] = 0 \). Observe further that unlike the results in [13]–[15], [23], [45] that hold on an average, the almost sure results in (16) cannot be specified in terms of problem parameters alone. Indeed, while it holds that \( C_t(n, \epsilon) \leq 2G^2(\Lambda_{\text{max}} + 1) \), such a bound is not very useful in the present case, as compared to the stronger convergence rate result in (16).

Finally, it is remarked that it may be possible to minimize the bound in (16) to the extent possible, by fixing \( T \) and choosing a corresponding step size. In the present case, given \( T \), the bound is the smallest when \( \epsilon = 1/\sqrt{T} \) which yields the following result

\[
\min_{0 \leq \tau \leq T-1} g(\lambda_\tau) \leq \frac{1}{T} \sum_{\tau=0}^{T-1} g(\lambda_\tau) \leq D + \mathcal{O}\left(\frac{1}{\sqrt{T}}\right) + C_T
\]

where the random variable \( C_T = \mathcal{O}(T^{-1/4}) \) almost surely. The result in Theorem 1 may therefore be seen as the generalization of the results in [15], [24] that have also reported an \( \mathcal{O}(T^{-1/2}) \) bound on average but have not analyzed the almost sure behavior. It is emphasized however that in practice, minimizing the bound may not necessarily translate to an improved convergence rate. Moreover, the number of iterations \( T \) for which the algorithm runs may not necessarily be known in advance, e.g., in target tracking applications. Instead, it may be simpler to specify a fixed value of \( \epsilon \), and continue to run the algorithm till the contribution of the \( \mathcal{O}(\frac{1}{n}) \) term becomes tolerably small.

Before proceeding with the proof of Theorem 1, an intermediate lemma establishing rate results on various time-averages is provided. The proof of Theorem 1 will subsequently utilize these results by expressing the optimality gap in (16) in terms of these time-averages.

**Lemma 1.** Let \( \mathcal{T} := \{t_1, t_2, \ldots, t_{|\mathcal{T}|}\} \) be a set of natural numbers such that \( t_i \neq t_j \). Then for any \( T \geq |\mathcal{T}| \) and \( \lambda, \lambda' \in \Lambda \), it holds under (A3')-(A4) that

\[
\left\| \frac{1}{T} \sum_{t \in \mathcal{T}} e_t(\lambda) \right\| \leq L_1^{\mathcal{T}}(T) \tag{19a}
\]

\[
\left\| \frac{1}{T} \sum_{t \in \mathcal{T}} e_t(\lambda) - e_t(\lambda') \right\| \leq L_2^{\mathcal{T}}(T) \|\lambda - \lambda'\| \tag{19b}
\]

where, for a given \( \zeta > 0 \), the random variables \( \{L_i^{\mathcal{T}}(T)\}_{i=1,2} \) satisfy

\[
T^{1/2 - \zeta} L_i^{\mathcal{T}}(T) \to 0 \quad \text{a.s. as } T \to \infty \tag{20a}
\]

\[
\mathbb{P}(L_i^{\mathcal{T}}(T) > T^{\zeta - 1/2}) < A_i \exp(-T^{2\zeta}) \tag{20b}
\]

where the constant \( A_i < \infty \) does not depend on \( T \).
Proof: Observe that the i.i.d. process \( \{ e_t(\lambda) \}_{t \in T} \) is zero-mean and satisfies
\[
\| e_t(\lambda) \| \leq \| f_t(\lambda) \| + \| f(\lambda) \| \leq 2G
\] (21)
where the last inequality holds from (A4). Therefore, it follows from the strong law of large numbers that for any \( T \geq |T| \), \( \frac{1}{T} \sum_{t \in T} e_t(\lambda) \rightarrow 0 \) almost surely as \( T \rightarrow \infty \). It can also be seen that the same holds for \( L_T(\lambda) := \left\| \frac{1}{T} \sum_{t \in T} e_t(\lambda) \right\| \). The rate results in (20) hold as consequences of the strong law of large numbers for i.i.d. sequences with bounded moments; see [46, Chap. 7] for (20a). Finally, (20b), follows from the Bernstein inequality applied to i.i.d. zero-mean and bounded random vectors \( \{ e_t(\lambda) \} \) [47].

Denote the \( j \)-th entry of \( e_t(\lambda) \) by \( e_t^j(\lambda) \) for \( 1 \leq j \leq K \). From (A3)-(A4), we have that \( e_t^j(\lambda) \) is bounded and continuously differentiable on \( \Lambda \). Consider arbitrary \( \lambda \neq \lambda' \in \Lambda \), and observe that since \( \Lambda \) is convex, it holds for any \( \beta \in [0,1] \) that \( \lambda_\beta := \beta \lambda + (1-\beta)\lambda' \in \Lambda \). It is now possible to use the mean-value theorem, which guarantees that there exists some \( \beta_j \in [0,1] \), such that
\[
e_t^j(\lambda) - e_t^j(\lambda') = \langle \nabla e_t^j(\lambda_\beta_j), \lambda - \lambda' \rangle.
\] (22)
Here, \( \nabla e_t^j(\lambda_\beta_j) \) is an i.i.d. random variable that is also zero-mean, since for continuously differentiable and bounded functions (cf. (A4)), we have that \( \mathbb{E} [ \nabla e_t^j(\lambda_\beta_j) ] = \nabla \mathbb{E} [ e_t^j(\lambda_\beta_j) ] = 0 \).

Taking summation in (22) and stacking the \( K \) components, it follows for any \( T \geq |T| \), that
\[
\frac{1}{T} \sum_{t \in T} e_t(\lambda) - e_t(\lambda') = E_T(\lambda, \lambda')(\lambda - \lambda')
\] (23)
where the \( K \times K \) matrix \( E_T(\lambda, \lambda') \) is defined as \( [ E_T(\lambda, \lambda') ]_{jk} := \frac{1}{T} \sum_{t \in T} [ \nabla e_t^j(\lambda_\beta_j) ]_k \), where the subscript is used to denote the \( k \)-th element of vector \( \nabla e_t^j(\lambda_\beta_j) \). Applying the Cauchy-Schwarz inequality to (23), we obtain \( \left\| \frac{1}{T} \sum_{t \in T} e_t(\lambda) - e_t(\lambda') \right\| \leq \| E_T(\lambda, \lambda') \| \| \lambda - \lambda' \| \). From the strong law of large numbers, we have that \( [ E_T(\lambda, \lambda') ]_{jk} \rightarrow 0 \) almost surely as \( T \rightarrow \infty \) for all \( 1 \leq j, k \leq K \). It can be seen that the same also holds for \( L_T^2(\lambda) := \left\| E_T(\lambda, \lambda') \right\| \). Finally, the rate results in (20) follow from [46, Chap. 7] and the Bernstein inequality applied to i.i.d. zero-mean and bounded random variables \( \{ [ \nabla e_t^j(\lambda_\beta_j) ]_k \} \) [47].

The proof of Theorem 1 follows in two steps: the derivation of the overall form required in (16), presented next; and the analysis of the random term \( C_t(n, \epsilon) \) deferred to Appendix B.

Proof of Theorem 1: In order to derive the bound in (16), recall that since \( g(\lambda) \) is convex, we have that,
\[
g(\lambda_t) \leq g(\lambda^*) + \langle f(\lambda_t), \lambda_t - \lambda^* \rangle, \quad t \in \mathbb{N}_0.
\] (24)
Letting \( g_t := g(\lambda_t) \), it follows from the non-expansive property of \( \mathcal{P}_\Lambda (\cdot) \) that
\[
\delta_{t+1} := \| \lambda_{t+1} - \lambda^* \|^2 = \| \mathcal{P}_\Lambda (\lambda_t - \epsilon f_t(\lambda_t)) - \lambda^* \|^2
\] (25)
where (27) follows from (A3') and (28) follows from (24). Rearranging (28) yields
\[
2\epsilon (g_t - D) \leq (\delta_t - \delta_{t+1}) - 2\epsilon (f_t(\lambda_t) - f(\lambda_t), \lambda_t - \lambda^*) + \epsilon G^2
\]  
Taking sum over \( \tau = 0, 1, \ldots, t - 1 \) and noting that \( B_0 = \delta_0 \) and that \( et \geq n \), yields
\[
\frac{1}{t} \sum_{\tau=0}^{t-1} g_{\tau} \leq D + \frac{\delta_0 - \delta_t}{2et} - \frac{1}{t} \sum_{\tau=0}^{t-1} (f_{\tau}(\lambda_{\tau}) - f(\lambda_{\tau}), \lambda_{\tau} - \lambda^*) + \frac{\epsilon G^2}{2} \leq D + \frac{B_0}{2n} + \frac{\epsilon G^2}{2} + C_t(n, \epsilon)
\]  
where the last inequality follows since \( \delta_t \geq 0 \) and the stochastic term in (30) is defined as
\[
C_t(n, \epsilon) := \left| \frac{1}{t} \sum_{\tau=0}^{t-1} (f_{\tau}(\lambda_{\tau}) - f(\lambda_{\tau}), \lambda_{\tau} - \lambda^*) \right|.
\]
Since (30) is of the same form as required in (16), it remains to show that \( C_t(n, \epsilon) \) converges in the sense of (17)-(17c). The convergence analysis for \( C_t(n, \epsilon) \) makes use of the bounds developed in Lemma 1 and is deferred to Appendix B.

It is remarked that the results in Theorem 1 can likely be generalized to the case when \( \Lambda \) is not necessarily compact. Such a generalization is likely possible because the strong law of large numbers, as well as the rate results in [46, Chap. 7] and [48, Theorem 1] only require the random process to have bounded moments. Nevertheless, the requirement that \( \|\lambda_t - \lambda^*\| \leq \Lambda_{\text{max}} < \infty \) is not too restrictive, and greatly simplifies the analysis.

B. Convergence rate for the SDSD algorithm

In order to apply the results developed in Sec. IV-A to the dual problem (8), observe that the stochastic subgradient of \( g(\lambda) \) for any \( \lambda \in \mathbb{R}^K_+ \) is given by
\[
f_t(\lambda) = s_t(x_t(\lambda), p_t(\lambda))
\]  
\[
x_t(\lambda) := \arg \max_{x \in \mathcal{X}} f_0(x) + \langle \lambda, u(x) \rangle
\]  
\[
p_t(\lambda) := \arg \max_{p \in \Pi_t} \langle \lambda, v(\varphi_t, \hat{p}) \rangle.
\]
With \( f_t(\lambda) \) as defined in (32), the projected SSD updates take the same form as (15), with \( \Lambda_{\text{max}} = 2\sqrt{K} \lambda_{\text{max}} \). Further the bound required in (A3') follows from (A3). Therefore, Theorem 1 applies as is to the dual objective function under (A3) and (A4).
For the resource allocation problem however, the behavior of the primal objective function is more important. The subsequent theorem characterizes the primal near-optimality when the running average of \( \{x_t(\lambda_t)\} \) is used for allocating resources. For the purpose of rate analysis, time is divided into epochs of duration \( \frac{1}{\epsilon} \) each, and the result is expressed in terms of \( \epsilon \) and \( n \).

**Theorem 2.** Under (A1)-(A4), and for \( n/\epsilon \leq t < (n + 1)/\epsilon \), the average primal objective function is near optimal in the following sense:

\[
    f_0(\bar{x}_t) \geq \frac{1}{t} \sum_{\tau=0}^{t-1} f_0(x_\tau) \geq P - \frac{R_0}{2n} - \frac{\epsilon G^2}{2} - C'_{\epsilon}(n, \epsilon) 
\]

where \( R_0 := \|\lambda_0\|^2 \), \( \bar{x}_t = \frac{1}{t} \sum_{\tau=0}^{t-1} x_\tau \), and the random variable \( C'_{\epsilon}(n, \epsilon) \) is such that for \( \zeta > 0 \),

\[
    \epsilon^{-1/2} C'_{\epsilon}(n, \epsilon) \to 0 \quad \text{a.s. as } \epsilon \to 0 \text{ for fixed } n < \infty \quad (36a)
\]

\[
    n^{1/2-\zeta} C'_{\epsilon}(n, \epsilon) \to 0 \quad \text{a.s. as } n \to \infty \text{ for fixed } \epsilon > 0 \quad (36b)
\]

\[
    \mathbb{P}(C_{\epsilon}(n, \epsilon) > \nu^{\zeta-1/2}) < A \exp(-\nu^2\zeta) \quad (36c)
\]

where \( \nu := \max\{\frac{1}{\epsilon}, n\} \) and \( A < \infty \) is a constant that does not depend on \( n \) or \( \epsilon \).

The term \( C'_{\epsilon}(n, \epsilon) \) in Theorem 2 is very similar to \( C^*_{\epsilon}(n, \epsilon) \) in Theorem 1, and therefore decays at the same rate. It follows from Theorem 2 that the resource allocation yielded by the projected SDSD algorithm is near optimal since the average primal objective value is close to \( P \). Similar to (16), the bound in (35) also holds for \( \max_{0 \leq \tau \leq t-1} f_0(x_\tau) \), as well as for \( f_0(\bar{x}_t) \). Further, the optimality gap in (35) is also similar to the one in (16), and therefore decays at the same rate. For details, see the discussion after the statement of Theorem 1.

In order to prove Theorem 2, the specific form of the bound in (35) is first established. The rest of the proof is much the same as before, and results from Lemma 1 are again used to derive the bounds on \( C'_{\epsilon}(n, \epsilon) \) as in the proof of Theorem 1.

**Proof of Theorem 2:** Recall that the subgradient of \( g(\lambda) \) is given by \( \mathbf{f}(\lambda) = \mathbb{E} [\mathbf{s}_t(x_t(\lambda), p_t(\lambda))] \), so that \( g(\lambda) = f_0(x(\lambda)) + \langle \lambda, \mathbf{f}(\lambda) \rangle \). Since \( f_0 \) is concave, the following inequalities hold:

\[
    f_0(\bar{x}_t) \geq \frac{1}{t} \sum_{\tau=0}^{t-1} f_0(x_\tau) = \frac{1}{t} \sum_{\tau=0}^{t-1} (f_0(x_\tau) + \langle \lambda_\tau, f(x_\tau) \rangle) - \frac{1}{t} \sum_{\tau=0}^{t-1} \langle \lambda_\tau, f(x_\tau) \rangle 
\]

\[
    = \frac{1}{t} \sum_{\tau=0}^{t-1} g(\lambda_\tau) - \frac{1}{t} \sum_{\tau=0}^{t-1} \langle \lambda_\tau, f(x_\tau) \rangle \geq g(\lambda^*) - \frac{1}{t} \sum_{\tau=0}^{t-1} \langle \lambda_\tau, f(x_\tau) \rangle. 
\]

Next, the second term in (38) can be bounded as

\[
    \|\lambda_{t+1}\|^2 = \|P_{\Lambda}(\lambda_t - \epsilon f_t(\lambda_t))\|^2 \leq \|\lambda_t - \epsilon f_t(\lambda_t)\|^2 
\]

\[
    \leq \|\lambda_t\|^2 - 2\epsilon(\lambda_t, f(\lambda_t)) + \epsilon^2 \|f_t(\lambda_t)\|^2 - 2\epsilon(\lambda_t - f(\lambda_t), \lambda_t) 
\]

October 15, 2018 DRAFT
\[ 2\varepsilon(\lambda_t, f(\lambda_t)) \leq \|\lambda_t\|^2 - \|\lambda_{t+1}\|^2 + \varepsilon^2 G^2 - 2\varepsilon(f_i(\lambda_t) - f(\lambda_t), \lambda_t) \] (41)

where, (39) follows from the non-expansiveness property of the projection operator and from the fact that \(0 \in \Lambda (\|\lambda_{t+1}\| = \|\lambda_{t+1} - 0\|)\), and (41) follows from (A3). Taking sum over \(\tau = 0, 1, \ldots, t - 1\) and dividing by \(2\varepsilon t\) yields

\[
\frac{1}{t} \sum_{\tau=0}^{t-1} \langle \lambda_\tau, f(\lambda_\tau) \rangle \leq \frac{\|\lambda_0\|^2}{2\varepsilon t} - \frac{\|\lambda_{t+1}\|^2}{2\varepsilon t} + \frac{\varepsilon G^2}{2} - \frac{1}{t} \sum_{\tau=0}^{t-1} \langle f(\lambda_\tau) - f(\lambda_\tau), \lambda_\tau \rangle \leq \frac{\|\lambda_0\|}{2n} + \frac{\varepsilon G^2}{2} + C'(n, \varepsilon) \tag{42}
\]

where, 

\[
C'(n, \varepsilon) := \left| \frac{1}{t} \sum_{\tau=0}^{t-1} (f(\lambda_\tau) - f(\lambda_\tau), \lambda_\tau) \right| .
\]

The bound in (35) follows by plugging back (42) into (38). The analysis for \(C'(n, \varepsilon)\) is much the same as in the proof of Theorem 1. The only difference for this case is that the iterate bound becomes \(\|\lambda_t\| \leq \sqrt{K}\lambda_{\text{max}}\) from (10). Consequently, after rearranging various terms in \(C'(n, \varepsilon)\) and using the triangle inequality in (69), (82), and (83), all occurrences of \(\Lambda_{\text{max}}\) get replaced with \(\sqrt{K}\lambda_{\text{max}}\). Since this is equivalent to redefining the constant \(\Lambda_{\text{max}}\) appropriately, the required rate results continue to hold. ■

V. APPLICATION TO D2D COMMUNICATIONS

This section details some implementation aspects of the SDS algorithm in the context of D2D communication problem considered in this paper under slow and fast fading scenarios. The Assumptions (A1)-(A4) are also verified for the problems at hand so as to ensure that Theorems 1 and 2 hold.

Before proceeding, the SDS algorithm for the general form of the D2D problem (1) is detailed. Specifically, the Lagrangian is given by

\[ L(r, \{p^i\}_{i \in M}, \lambda) = U(r) - E \left[ \sum_{i \in M} c^i p^i \right] + \lambda R \left[ \sum_{i \in M} R_i (p^i, \gamma^i) - r \right] \tag{43} \]

which yields the following stochastic algorithm. Since the Lagrangian is separable in \(r\) and \(p^i\), starting with arbitrary \(\lambda_0\), the primal iterates at time slot \(t\) become:

\[ r_t(\lambda_t) \in \arg \max_{r \in \mathbb{R}} U(r) - \lambda_r r \tag{44} \]

\[ \{p_t^i(\lambda_t)\}_{i \in M} \in \arg \max_{\{\tilde{p}^i\}_{i \in M}} \sum_{i \in M} \left[ \lambda_t R_i (\tilde{p}^i, \gamma^i) - c^i \tilde{p}^i \right] . \tag{45} \]

At the end of each time slot, the dual variable is updated as

\[ \lambda_{t+1} = \mathcal{P}_\Lambda \left( \lambda_t - \varepsilon \left[ \sum_{i \in M} R_i (p_t^i(\lambda_t), \gamma^i) - r_t(\lambda_t) \right] \right) \tag{46} \]

Recall that the set of functions \(\mathcal{P}\) is such that only one user, denoted by

\[ i_t := \arg \max_{i \in M} p_t^i(\lambda_t) , \tag{47} \]
is allocated non-zero power at time slot $t$. Therefore, the dual variable is updated as

$$
\lambda_{t+1} = P\Lambda (\lambda_t - \epsilon \left[ R_{it}(p_{it}^{\Lambda}(\lambda_t), \gamma_{it}^{\Lambda}) - r_t(\lambda_t) \right])
$$

(48)

The full algorithm is summarized in Algorithm 1. The rate analysis developed in Sec. IV applies to the present problem under the following assumptions.

**B1. Continuous random variables:** The random variables $\varphi_t = (M_t, \{c_i^t\}_{i \in M_t}, \{\gamma_i^t\}_{i \in M_t})$ are i.i.d., have continuous cdfs, and finite supports, i.e., $M_t \subset M$, $\gamma_i^t \in [\gamma_{\min}, \gamma_{\max}]$, and $c_i^t \in [c_{\min}, c_{\max}]$ for each $i \in M_t$.

**B2. Power constraints:** The set $\mathcal{P} := \{p : \mathbb{R}^{3M} \to \mathbb{R}^M \mid p_{\varphi} \in \Pi_{\varphi}\}$, where for any $\varphi_t$, we have that $\Pi_{\varphi_t} := \Pi_t = \{p_t \in \mathbb{R}^M \mid p_{it} = 0 \; j \notin M_t, \|p_t\|_0 = 1, p_{it}^0c_i^t \in [C_{\min}, C_{\max}]\}$, where $i_t := \arg\max_{i \in M_t} p_{it}^0$ and $p_{it}^0 = [p_{it}]$. 

**B3. High SNR:** It is assumed that $\gamma_i^t \gg 1$ for all $i \in M_t$.

The finite support of the random quantities is again motivated from practical considerations. The set $\mathcal{P}$ also includes limits on the maximum affordable cost $C_{\max}$ and the minimum operational cost or minimum allowable transaction amount $C_{\min}$. A maximum power constraint of the form $p_{it}^0 \leq P_{\max}$ may also be included within $\mathcal{P}$. However, for the present application, it is assumed that the caches are not energy constrained, so that $P_{\max} \gg C_{\max}/c_{it}^0$ for all $i \in M_t$. In other words, the user’s cost constraint is much more stringent than the cache’s energy constraint.

Finally, the high SNR assumption is justified if there are always enough mobile caches available at all slots. In a typical setting, the MoUE may “see” hundreds of advertisements from potential mobile cache servers, but may choose to consider only tens of users with which control messages may be exchanged easily. Next, the discussion for slow and fast fading cases will be carried out.

---

**Algorithm 1** Edge Caching via D2D Communications

1: **Initialize** $\lambda_0$
2: **Repeat** for $t = 0, 1, 2, \ldots$,
3: Collect $c_i^t$ and $\gamma_i^t$ from each active UE $i \in M_t$
4: **Primal update** Determine the “winning” cache $i_t$ and the allocated power from (45)
5: **Download at rate** $R_{it}(p_{it}^0(\lambda_t), \gamma_{it}^0)$ from user $i_t$
6: **Dual update** Update $\lambda_{t+1}$ using (48)
A. Slow Fading

Recall that under slow fading, since power allocation occurs every coherence interval, we have for high SNR (cf. (B3)), that \( R_i(p_t^i(\lambda_t), \gamma_t^i) \approx W \log_2(p_t^i(\lambda_t)\gamma_t^i/\alpha) \). The primal iterate in (45) can be found in two steps. First the optimum transmit power for all potential users is determined, i.e., for each \( i \in \mathcal{M}_t \),

\[
p_t^i(\lambda_t) = \arg \max_{\hat{p}^i} \lambda_t \log_2(\hat{p}^i\gamma_t^i/\alpha) - c_t^i\hat{p}^i
\]

\[
\text{s. t. } C_{\min} \leq c_t^i\hat{p}^i \leq C_{\max}
\]

\[
= \left[ \frac{W\lambda_t}{c_t^i} \right] \frac{C_{\max}/c_t^i}{C_{\min}/c_t^i}
\]

(50)

The winning user is the one that maximizes the objective function, i.e.,

\[
i_t = \arg \max_{i \in \mathcal{M}_t} \lambda_t R_i(p_t^i, \gamma_t^i) - c_t^i\hat{p}_t^i = \arg \max_{i \in \mathcal{M}_t} \frac{\gamma_t^i}{c_t^i}
\]

(51)

where the expression in (51) derived in Appendix C. An implication of (51) is that the random variable \( i_t \) is i.i.d. Finally, it holds that \( p_t^j = \hat{p}_t^i(\lambda_t) \) for \( j = i_t \) and zero otherwise. Similarly, \( r_t \) is calculated as \( r_t(\lambda_t) = \left[ \frac{1}{\lambda_t} \right]_{r_{\min}}^{r_{\max}} \) resulting in the dual update

\[
\lambda_{t+1} = P_\lambda (\lambda_t - \epsilon \left[ W \log_2(p_t^i(\lambda_t)\gamma_t^i/\alpha) - r_t(\lambda_t) \right])
\]

(52)

An additional assumption regarding the parameter values is made in the slow fading case:

B4. **Strict feasibility:** The problem parameters satisfy \( r_{\min} < \mathbb{E} [\max_i \log_2(C_{\max}\gamma_t^i/c_t^i)] \).

The strict feasibility condition is required for ensuring the existence of a Slater point. Since it holds that \( \gamma_t^i/c_t^i \geq \gamma_{\min}/c_{\max} \), it is possible to satisfy (B4) by keeping \( r_{\min} \) sufficiently small and/or if \( \gamma_{\min} \) is sufficiently large.

Having stated the algorithm and all required assumptions, the following Lemma summarizes the main result of this subsection.

**Corollary 1.** Under (B1)-(B4), the iterates obtained from (50)-(52) adhere to the rate bounds stated in Theorems 1 and 2.

**Proof:** For the results in Theorem 1 and 2 to apply, it suffices to verify that assumptions (A1)-(A4) are satisfied under the slow fading case. The random variable \( \varphi_t \) has a non-atomic pdf since the channel gains \( \gamma_t^i \) have a continuous cdf, thus confirming (A1); see also [40]. The Slater’s condition is met by choosing \( \bar{r} = r_{\min} \) and \( \hat{p}_t^i = C_{\max}/c_t^i \) where \( i_t \) is given in (51) and

October 15, 2018 DRAFT
zero for all \( j \neq i_t \). For such a choice, it holds from (B4) that \( \tilde{r} < \mathbb{E} \left[ \log_2(p_i^t \gamma_i^t) \right] \), which is the required condition for strict feasibility. For a given \( \lambda \), the subgradient function is given by

\[
f_t(\lambda) = W \log_2(p_i^t(\lambda)\gamma_i^t/\alpha) - r_t(\lambda)
\]

where \( i_t \) and \( p_i^t \) are evaluated as in (51) and (50). A bound on the subgradient (cf. (A3)) may therefore be found as \( |f_t(\lambda)| \leq W \log_2 \left( \frac{C_{\max} \gamma_i^t}{\alpha c_{\min}} \right) + r_{\max} =: G \). Next, in order to verify (A4), the expression for the stochastic subgradient error \( e_t(\lambda) := f_t(\lambda) - \mathbb{E}\left[ f_t(\lambda) \right] \) is first derived. Recalling that \( i_t = \arg \max_i \gamma_i^t/c_i^t \), consider the following three cases,

1) When \( \lambda < C_{\min}/W \), it holds that \( p_i^t = C_{\min}/c_i^t \), implying that

\[
e_t(\lambda) = W \log_2 \left( \frac{C_{\min}\gamma_i^t}{\alpha c_i^t} \right) - \left[ \frac{1}{\lambda} \right]_{r_{\min}}^{r_{\max}} - \mathbb{E} \left[ W \log_2 \left( \frac{C_{\min}\gamma_i^t}{\alpha c_i^t} \right) - \left[ \frac{1}{\lambda} \right]_{r_{\min}}^{r_{\max}} \right] = W \log_2 \left( \frac{\gamma_i^t}{c_i^t} \right) - \mathbb{E} \left[ W \log_2 \left( \frac{\gamma_i^t}{c_i^t} \right) \right].
\]

where the expectations are with respect to \( \varphi_t \).

2) When \( C_{\min}/W \leq \lambda \leq C_{\max}/W \), it holds that \( p_i^t = W \lambda \gamma_i^t/c_i^t \), implying that

\[
e_t(\lambda) = W \log_2 \left( \frac{W \lambda \gamma_i^t}{c_i^t} \right) - \left[ \frac{1}{\lambda} \right]_{r_{\min}}^{r_{\max}} - \mathbb{E} \left[ W \log_2 \left( \frac{W \lambda \gamma_i^t}{c_i^t} \right) - \left[ \frac{1}{\lambda} \right]_{r_{\min}}^{r_{\max}} \right] = W \log_2 \left( \frac{\gamma_i^t}{c_i^t} \right) - \mathbb{E} \left[ W \log_2 \left( \frac{\gamma_i^t}{c_i^t} \right) \right].
\]

3) Similarly, when \( \lambda > C_{\max}/W \), it holds that \( p_i^t = C_{\max}/c_i^t \), implying that

\[
e_t(\lambda) = W \log_2 \left( \frac{\gamma_i^t}{c_i^t} \right) - \mathbb{E} \left[ W \log_2 \left( \frac{\gamma_i^t}{c_i^t} \right) \right].
\]

Therefore, the subgradient error is a zero-mean random variable that does not depend on \( \lambda \), and is therefore trivially continuously differentiable in \( \lambda \).

\[\square\]

**B. Fast Fading**

In the more realistic fast fading case, the power allocation and downloads occur over several coherence intervals. Under the high SNR assumption, the rate becomes \( R_t(p_i^t, \gamma_i^t) \approx W \log_2(p_i^t \gamma_i^t/\alpha) + W \psi_i \) where \( \psi_i = \mathbb{E}_h [\log_2(h_i)] \) for a given user \( i \) [38]. As in the slow fading case, the primal iterates are again found in two steps. First, the power allocation for a potential user \( i \) is found,

\[
\hat{p}_i^t(\lambda_t) = \left[ \frac{W \lambda_t}{c_i^t} \right]^{C_{\max}/c_i^t} \left[ \frac{C_{\min}/c_i^t}{C_{\max}/c_i^t} \right].
\]
It is shown in Appendix C that the winning user for the fast fading case can be written as

$$i_t = \text{arg max}_{i \in M_t} \log_2 \left( \frac{\gamma_i^t}{c_i^t} \right) + \psi_i,$$  \hspace{1cm} (60)

Finally, since \( r_t(\lambda_t) = \max\{\min\{1/\lambda_t, r_{\text{max}}\}, r_{\text{min}}\} \) as before, the dual update is given by

$$\lambda_{t+1} = \mathcal{P}_A (\lambda_t - \epsilon \left[ W \log_2 (p_i^t(\lambda_t) \gamma_i^t / \alpha) + W \psi_i - r_t(\lambda_t) \right]) .$$  \hspace{1cm} (61)

In order to apply the rate bounds in Theorems 1 and 2, we again assume (B1)-(B3), and make the following assumption analogous to (B4).

**B5. Strict feasibility:** The problem parameters satisfy \( r_{\text{min}} < \mathbb{E} \{\max_i \{\log_2(C_{\text{max}} \gamma_i^t / c_i^t) + \psi_i\}\} \).

As in the slow fading case, (B5) allows us to obtain a Slater point, as required by (A2). The following Lemma summarizes the result for the fast fading case.

**Corollary 2.** Under (B1)-(B3) and (B5), the iterates obtained from (59)-(61) adhere to the rate bounds stated in Theorems 1 and 2.

**Proof:** The As in Lemma 1, it suffices to verify assumptions (A1)-(A4). The random variable \( \varphi_t \) has a non-atomic pdf as remarked earlier. Similarly, it can be verified that the Slater point is given by \( \bar{r} = r_{\text{min}} \) and \( \bar{p}^i_t = C_{\text{max}} / c_i^t \) where \( i_t \) is as given in (60), and zero for all \( j \neq i_t \). The subgradient bound required in (A3) now becomes, \( |f_t(\lambda)| \leq W \log_2 \left( \frac{C_{\text{max}} \gamma_{\text{max}}}{\alpha c_{\text{min}}} \right) + W \psi_{\text{max}} + r_{\text{max}} =: G \) where \( \psi_{\text{max}} := \max_i \psi_i \). Finally, in order to verify (A4), we proceed as in the proof of Lemma 1 and derive an expression for the subgradient error \( e_t(\lambda) := f_t(\lambda) - \mathbb{E} [f_t(\lambda)] \). Since expression for the allocated power is the same for the two cases, it can be seen that for the fast fading case as well \( e_t(\lambda) = W \log_2 \left( \frac{\gamma_i^t}{c_i^t} \right) - \mathbb{E} \left[ W \log_2 \left( \frac{\gamma_i^t}{c_i^t} \right) \right] \) where \( i_t \) is found as in (60). Since \( e_t(\lambda) \) does not depend on \( \lambda \), (A4) also holds trivially in the fast fading scenario.

**Q.E.D.**
VI. Numerical Tests

This section describes the numerical tests on the D2D example discussed in Sec. V. The convergence rate of the SDSD algorithm is studied for the fast fading scenario depicted in Fig. 2. For the simulations, we consider $M = 25$ operational UEs. At each time slot, the MoUE receives advertisements from a random subset $\mathcal{M}_t$ of 5 to 25 UEs. Without loss of generality, downloading from the $i$-th UE incurs a cost of $c_i = i$ per unit of transmit power. The lower and upper limits for each transaction are set as $C_{\min} = 1$ and $C_{\max} = 25$, respectively. The average channel gains $\gamma^t_i$ are assumed to be Rayleigh distributed with $\gamma_{\min} = 0.1$ and $\gamma_{\max} = 65$, and for simplicity, the parameters $\alpha$, and $\psi_i$ are all set to unity. In order to keep the numbers realistic, we set $W = 1$ MHz. Finally, in order to ensure Slater’s condition, we set $r_{\min} = 0.2$ and $r_{\max} = 10$. In realistic scenarios, since the optimal rate is expected to be greater than $r_{\min}$, it follows from the definition of $r_t(\lambda^t)$ in Sec.V-A that $\lambda^* > 1/r_{\min}$. Therefore it is safe to take $\lambda_{\max} \gg 1/r_{\min}$.

Fig. 2a shows the evolution of the utility function calculated using running averages $\bar{r}_t := 1/t \sum_{\tau=0}^{t-1} r_\tau$ of the allocated rate with iterations. As expected from Theorem 2, the utility function converges to a value that is closer to the optimal when $\epsilon$ is small. Similarly, Fig. 2b shows the evolution of the dual objective function, which again converges to a point closer to the optimal when $\epsilon$ is small. Observe from the results that for $\epsilon = 0.1$, the oscillations continue even as number of iterations go to infinity, as implied by Theorem 1. These oscillations are allowed due to the presence of an $O(\epsilon)$ term on the right-hand side of (17), and are well-documented for the constant step size stochastic subgradient type algorithms [8], [10], [13].

The convergence rate result of Theorem 1 is further illustrated in Fig. 3(a) and Fig. 3(b), where the deterministic terms in (16) are not included. The stochastic term $C_t(n, \epsilon)$ is calculated from (31) and plotted against both $\epsilon$ and $n$. It can be seen from both plots that $C_t(n, \epsilon) \to 0$ as either $n \to \infty$ or $\epsilon \to 0$, as claimed in Theorem 1.

| UE selection   | Downloaded data (Mb) | Cost incurred | Avg. Utility minus penalty |
|----------------|-----------------------|---------------|---------------------------|
| Proposed       | 16316                 | 4501          | 0.79                      |
| Random         | 12465                 | 19951         | -3.20                     |
| Opportunistic  | 3987                  | 177           | 0.67                      |

TABLE I: Performance comparison (1000 times slots)

Having studied the convergence properties of the SDSD algorithm, we now focus on some of the nuances of the edge-caching formulation in (1). To begin with, the performance of the proposed scheme is compared against that obtained from two naive algorithms: random and
opportunistic. The maximum transmit power for the three cases is scaled so as to ensure equal aggregate power consumption. As the name suggests, an MoUE following the random scheme selects an available UE randomly and without paying any attention to the channel or the cost of the UE. The data is transmitted at the maximum power so as to ensure the maximum rate. As evident from Table I, such a scheme is able to obtain a higher download rate but also at a significantly higher cost. In contrast, the opportunistic scheme advocates a parsimonious approach wherein the MoUE always selects an available UE with the lowest cost. Subsequently, the UE transmits with the maximum power but ultimately achieves a lower aggregate download rate, due to suboptimal channel conditions.

Fig. 4 provides results from the perspective of the UEs and is generated by running the same algorithm for 1000 independent identically distributed MoUEs. In particular, if an MoUE follows the optimal policy determined by (1), the UEs may be interested in knowing a reasonable price value. As expected, it is clear from Fig. 4, that the UEs that charge more are selected less often have lower data usage. Consequently, the aggregate revenue of the UEs with the lowest charges is also the highest. More interestingly however, such high-priced UEs have a very high revenue per Mb of data served. The intuition here is that UEs with high costs are only selected when their channel gains are proportionally higher than the others. Therefore, all transmissions to such UEs occur at higher rates and correspondingly lower power. In summary, by operating only under favorable channel conditions, the high-priced UEs extract a greater revenue for every bit that they serve. Note however that the revenue appears to saturate, and increases very slowly for very high prices.

VII. CONCLUSIONS

This paper considers a general stochastic resource allocation problem and solved using constant step-size stochastic subgradient descent algorithm in an online manner. A stochastic bound on the gap between the objective function and the optimum is developed and analyzed in an almost sure
sense, generalizing the existing results. The bounds characterize the precise manner in which the optimality gap behaves for fixed and arbitrarily small step-sizes. The convergence rate analysis is also extended to a class of stochastic resource allocation problems that utilize stochastic dual subgradient descent (SDSD) iterations. Existing results on near-optimality of the primal average objective function are again generalized for convergence rate analysis. As an example, a resource allocation problem is formulated in the context of mobile caching in device-to-device communications, and solved via SDSD. The regularity conditions required for the rate analysis are verified, and numerical tests are provided, further substantiating the convergence rate results.

### APPENDIX A

**A BOUND ON $\|\lambda^*\|_\infty$**

From (A2), there exists some $\bar{x} \in \mathcal{X}$ and $\bar{p} \in \mathcal{P}$, such that $\mathbb{E}[s_t(\bar{x}, \bar{p}_t)] > 0$, where recall that $\bar{p}_t := \bar{p}_{\varphi_t}$ for all $t \in \mathbb{N}_0$ and the expectation is with respect to $\varphi_t$. Given $\bar{\lambda} \in \mathbb{R}^K_+$, define the sublevel set $Q_{\bar{\lambda}} := \{\lambda \in \mathbb{R}^K_+ \mid g(\lambda) \leq g(\bar{\lambda})\}$, and observe that for any $\lambda \in Q_{\bar{\lambda}}$, it holds that

$$ g(\bar{\lambda}) \geq g(\lambda) = \max_{x \in \mathcal{X}, p \in \mathcal{P}} f_0(x) + \langle \lambda, \mathbb{E}[s_t(x, p_t)] \rangle \geq f_0(\bar{x}) + \langle \lambda, \mathbb{E}[s_t(\bar{x}, \bar{p}_t)] \rangle. \quad (62) $$

Rearranging the expression in (62), we obtain

$$ \sum_{k=1}^{K} [\lambda]_k \mathbb{E}[s_t(\bar{x}, \bar{p}_t)]_k \leq g(\bar{\lambda}) - f_0(\bar{x}) \Rightarrow \|\lambda\|_\infty \leq \sum_{k=1}^{K} [\lambda]_k \leq \frac{g(\bar{\lambda}) - f_0(\bar{x})}{\chi(\bar{x}, \bar{p})} \quad (63) $$

where $\chi(\bar{x}, \bar{p}) := \min_{1 \leq k \leq K} \mathbb{E}[s_t(\bar{x}, \bar{p}_t)]_k$. Observe that $Q_{\lambda^*} = \{\lambda \in \mathbb{R}^K_+ \mid g(\lambda) \leq D\}$, so that it follows from (63) that

$$ \|\lambda^*\|_\infty \leq \frac{D - f_0(\bar{x})}{\chi(\bar{x}, \bar{p})} \quad (64) $$

Finally, since $g(\bar{\lambda}) \geq D$ for all $\bar{\lambda} \in \mathbb{R}^K_+$, the bound in (64) can be relaxed to yield (12).
APPENDIX B

ASYMPTOTIC PROPERTIES OF $C_T$

Proof: In order to study the convergence rate of $C_T$, the time is divided into epochs of duration $1/\epsilon$, so that there exists some $n \geq 0$ that satisfies $n/\epsilon \leq T < (n+1)/\epsilon$. Since $n := \lfloor \epsilon t \rfloor$, where $\lfloor \cdot \rfloor$ denotes the floor operation, is an arbitrary number, such a split allows the value of $t$ to be increased by keeping either $n$ or $\epsilon$ fixed, and varying the other. It is therefore possible to separately study the effects of choosing larger $n$ or smaller $\epsilon$ values. For instance, if $\epsilon = 0.1$, the time is divided into epochs of duration 10 iterations each. Hence, the zeroth epoch consists of iterations $0 \leq t \leq 9$, the first epoch consists of iterations $10 \leq t < 19$, and so on. With such a split, the classical asymptotic analysis for $t \to \infty$ is equivalent to fixing $\epsilon$ and letting $n \to \infty$.

Additionally, the proposed split allows us to study the case when $n$ is fixed, but the algorithm is run with different values of $\epsilon$. This proof is devoted to the analysis of $C_t(n, \epsilon)$, and relies on rearranging the terms in (31) so that the results in developed in Lemma 1 can be applied. The proof is divided into two parts, one for each mode of convergence in (17).

Fixed $n < \infty$ and $\epsilon \to 0$: For this case, $C_t(n, \epsilon)$ is split into summands corresponding to each epoch till time $t$, that is,

$$
C_t(n, \epsilon) = \frac{1}{\epsilon t} \sum_{m=0}^{n} C^m(\epsilon) \quad \text{where,} \quad C^m(\epsilon) := \epsilon \sum_{\tau=\ell_m}^{u_m} \langle f_\tau(\lambda_\tau) - f(\lambda_\tau), \lambda_\tau - \lambda^* \rangle.
$$

(65)

The limits in the summation are defined as $\ell_m := m/\epsilon$ and $u_m := (m+1)/\epsilon - 1$ for $m < n$ while $u_n := t - 1$. Next, define for all $\lambda \in \Lambda$ and $\ell_m \leq \tau \leq u_m$

$$
z_\tau(\lambda) = \epsilon \sum_{i=\ell_m}^{\tau} \langle f_i(\lambda) - f(\lambda), \lambda - \lambda^* \rangle.
$$

(66)

Substituting (66) in (65), we obtain

$$
C^m(\epsilon) = z_{u_m}(\lambda_{u_m+1}) = \sum_{\tau=\ell_m}^{u_m} (z_\tau(\lambda_{\tau+1}) - z_{\tau}(\lambda_\tau)).
$$

(67)

Such a split allows us to use (19) in order to bound the magnitude of each term separately. Specifically, letting $T_m := \{\ell_m, \ell_m + 1, \ldots, u_m\}$,

$$
|z_{u_m}(\lambda_{u_m+1})| = \epsilon \left| \sum_{i=\ell_m}^{u_m} \langle f_i(\lambda_{u_m+1}) - f(\lambda_{u_m+1}), \lambda_{u_m+1} - \lambda^* \rangle \right|
$$

(68)

$$
\leq \epsilon \sum_{i=\ell_m}^{u_m} \|f_i(\lambda_{u_m+1}) - f(\lambda_{u_m+1})\| \|\lambda_{u_m+1} - \lambda^*\| \leq L_{1/\epsilon}^1(T_m) \Lambda_{\max}.
$$

(69)
Similarly, denoting $\mathcal{T}_\tau := \{\ell_m, \ell_m + 1, \ldots, \tau\}$ for all $\ell_m \leq \tau \leq u_m$, it holds from using triangle inequality and (19), that

\[
|z_\tau(\lambda_{\tau+1}) - z_\tau(\lambda_\tau)| = \epsilon \left| \sum_{i=\ell_m}^\tau (\langle f_i(\lambda_{\tau+1}) - f(\lambda_{\tau+1}), \lambda_{\tau+1} - \lambda^* \rangle - \langle f_i(\lambda_\tau) - f(\lambda_\tau), \lambda_\tau - \lambda^* \rangle) \right|
\]

(70)

\[
\leq \epsilon \left| \sum_{i=\ell_m}^\tau (f_i(\lambda_{\tau+1}) - f(\lambda_{\tau+1}) - f(\lambda_\tau) + f(\lambda_\tau), (\lambda_{\tau+1} - \lambda^*) - (\lambda_\tau - \lambda^*)) \right| + \epsilon \left| \sum_{i=\ell_m}^\tau (f_i(\lambda_\tau) - f(\lambda_\tau), \lambda_{\tau+1} - \lambda_\tau) \right|
\]

(71)

where the (71) uses the non-expansive property of the projection operator $P_{\Lambda}(\cdot)$ and the boundedness of the stochastic subgradients (cf. (A3)). Substituting (69) and (71) into the expression for $C^m(\epsilon)$ yields the following bound

\[
|C^m(\epsilon)| \leq L_{1/\epsilon}(T_m)\Lambda_{\text{max}} + \epsilon G \sum_{\tau = \ell_m}^{u_m} L_{1/\epsilon}^2(T_\tau')\Lambda_{\text{max}} + L_{1/\epsilon}^1(T_\tau')
\]

\[
\leq L_{1/\epsilon}(T_m)\Lambda_{\text{max}} + G \sup_{0 \leq \tau < t} (L_{1/\epsilon}^2(T_\tau')\Lambda_{\text{max}} + L_{1/\epsilon}^1(T_\tau'))
\]

(73)

Finally, the bound for $C_t(n, \epsilon)$ becomes

\[
C_t(n, \epsilon) \leq \frac{1}{n} \sum_{m=0}^n |C^m(\epsilon)| \leq \sup_{0 \leq m \leq n} |C^m(\epsilon)|
\]

(72)

\[
\leq \sup_{0 \leq m \leq n} L_{1/\epsilon}(T_m)\Lambda_{\text{max}} + G \sup_{0 \leq \tau < t} (L_{1/\epsilon}^2(T_\tau')\Lambda_{\text{max}} + L_{1/\epsilon}^1(T_\tau'))
\]

(73)

Therefore, the rate result from Lemma 1 implies that

\[
\epsilon^{1/2}C_t(n, \epsilon) \leq \Lambda_{\text{max}} \sup_{0 \leq m \leq n} \epsilon^{1/2}L_{1/\epsilon}^1(T_m) + G \sup_{0 \leq \tau < t} (\epsilon^{1/2}L_{1/\epsilon}^2(T_\tau')\Lambda_{\text{max}} + \epsilon^{1/2}L_{1/\epsilon}^1(T_\tau'))
\]

(74)

which goes to zero almost surely as $\epsilon \to 0$, yielding the bound in (17a). Likewise, let $A_{1m} < \infty$ be the constant associated with the bounds for $\Lambda_{\text{max}}L_{1/\epsilon}^1(T_m)$, as necessitated by Lemma 1. Then, using the union bound, it follows that

\[
P \left( \sup_{0 \leq m \leq n} L_{1/\epsilon}(T_m) > \epsilon^{1/2-\zeta} \right) \leq \sum_{m=0}^n A_{1m} \exp \left( -\epsilon^{-2\zeta} \right) \leq A_1 \exp \left( -\epsilon^{-2\zeta} \right)
\]

(75)

where $A_1 := \sum_m A_{1m}$. Along the same lines, the result in Lemma 1 and the subsequent use of the union bound imply that there exist a constant $A_2 < \infty$ such that the probability of the second term in (73) exceeds $\epsilon^{1/2-\zeta}$ is bounded by $A_2 \exp \left( -\epsilon^{-2\zeta} \right)$. Combining the two bounds, and again using union bound, we have that

\[
P \left( C_t(n, \epsilon) > \epsilon^{1/2-\zeta} \right) \leq A_3 \exp \left( -\epsilon^{-2\zeta} \right)
\]

(76)
where $A_3 = A_1 + A_2$.

*Fixed $\epsilon > 0$ and $n \to \infty$: In this case, $C'_t(n, \epsilon)$ must now be split into two terms as follows,*

\[ C'_t(n, \epsilon) \leq |C(n)| + \frac{|D(n)|}{n} \]  

(77)

where,

\[ C(n) := \sum_{\tau=\ell_m}^{n/\epsilon - 1} (f_\tau(\lambda_\tau) - f(\lambda_\tau), \lambda_\tau - \lambda^*) \]  

(78)

\[ D(n) := \epsilon \sum_{\tau=n/\epsilon}^{T-1} (f_\tau(\lambda_\tau) - f(\lambda_\tau), \lambda_\tau - \lambda^*) \]  

(79)

For this analysis, it is assumed without loss of generality that $1/\epsilon$ is an integer. That way, the subscripts $m/\epsilon + \tau$ are also integers and the floor operation is not required. Given $\epsilon$, note that $D(n)$ is a sum of a fixed number of bounded random variables, so that $D(n)/n \to 0$ surely as $n \to \infty$. In order to bound $C(n)$, define for all $\lambda \in \Lambda$ and $0 \leq \tau \leq 1/\epsilon - 1$,

\[ z_\tau(\lambda) = \frac{1}{n} \sum_{\ell=0}^{\tau} \sum_{m=0}^{n-1} (f_{m/\epsilon + \tau}(\lambda) - f(\lambda), \lambda - \lambda^*). \]  

(80)

Then, it follows that

\[ C(n) = z_{n/\epsilon-1}(\lambda_{n/\epsilon}) - \sum_{\tau=0}^{1/\epsilon-1} (z_\tau(\lambda_{\tau+1}) - z_\tau(\lambda_\tau)). \]  

(81)

It is now possible to bound each term in (81) separately. Defining $T^\tau := \{ m/\epsilon + \tau \}_{m=0}^{n-1}$, and using (19), it follows that

\[ |z_{n/\epsilon-1}(\lambda_{n/\epsilon})| \leq \sum_{\tau=0}^{1/\epsilon-1} \left\| \frac{1}{n} \sum_{m=0}^{n-1} (f_{m/\epsilon + \tau}(\lambda_{n/\epsilon}) - f(\lambda_{n/\epsilon}), \lambda_{n/\epsilon} - \lambda^*) \right\| \]  

\[ \leq \sum_{\tau=0}^{1/\epsilon-1} \left\| \frac{1}{n} \sum_{m=0}^{n-1} (f_{m/\epsilon + \tau}(\lambda_{n/\epsilon}) - f(\lambda_{n/\epsilon})) \right\| \| \lambda_{n/\epsilon} - \lambda^* \| \]  

\[ \leq \lambda_{\text{max}} \sum_{\tau=0}^{1/\epsilon-1} L^1_n(T^\tau). \]  

(82)

Proceeding similarly,

\[ |z_\tau(\lambda_{\tau+1}) - z_\tau(\lambda_\tau)| \leq \sum_{\ell=0}^{\tau} \left\| \frac{1}{n} \sum_{m=0}^{n-1} (f_{m/\epsilon + \tau}(\lambda_{\tau+1}) - f(\lambda_{\tau+1}), \lambda_{\tau+1} - \lambda^*) - (f_{m/\epsilon + \tau}(\lambda_\tau) - f(\lambda_\tau), \lambda_\tau - \lambda^*) \right\| \]  

\[ \leq \sum_{\ell=0}^{\tau} \left\| \frac{1}{n} \sum_{m=0}^{n-1} (f_{m/\epsilon + \tau}(\lambda_{\tau+1}) - f(\lambda_{\tau+1}) - f_{m/\epsilon + \tau}(\lambda_\tau) + f(\lambda_\tau)) \right\| \| \lambda_{\tau+1} - \lambda^* \| \]
\[
+ \sum_{\tau=0}^{\tau} \left| \frac{1}{n} \sum_{m=0}^{n-1} (f_{m/\epsilon^+}(\lambda_\tau) - f(\lambda_\tau)) \right| |\lambda_{\tau+1} - \lambda_\tau|
\leq \sum_{\tau=0}^{\tau} \left( L_n^2(T) \Lambda_{\max} + L_n^1(T) \right) |\lambda_{\tau+1} - \lambda_\tau|
\leq \varepsilon G \sum_{\tau=0}^{\tau} \left( L_n^2(T) \Lambda_{\max} + L_n^1(T) \right).
\]  
\text{(83)}

Finally, substituting (82) and (84) into the expression for \( C(n) \), and noting that \( 1/\epsilon \) is a fixed number, the following bound is obtained
\[
\varepsilon |C(n)| \leq \Lambda_{\max} \varepsilon \sum_{\tau=0}^{1/\epsilon-1} L_n^1(T) + \varepsilon^2 \sum_{\tau=0}^{1/\epsilon-1} G \sum_{\tau=0}^{\tau} \left( L_n^2(T) \Lambda_{\max} + L_n^1(T) \right)
\leq \Lambda_{\max} \sup_{0 \leq \tau < 1/\epsilon} L_n^2(T) + G \sup_{0 \leq \tau < 1/\epsilon} \sup_{0 \leq \tau < \tau} \left( L_n^2(T) \Lambda_{\max} + L_n^1(T) \right)
\text{(85)}
\]

which goes to zero almost surely as \( n \to \infty \), implying that \( C_t(n, \epsilon) \to 0 \) almost surely as \( n \to \infty \). Both the rate results can again be inferred as in the previous case. Indeed, similar to (76), given \( \zeta > 0 \), there exist \( A_4 < \infty \) such that \( \mathbb{P} \left( C_t(n, \epsilon) > n^{\epsilon-1/2} \right) \leq A_4 \exp(-n^{2\zeta}) \).

Combining with (76), the probability bounds can be written as
\[
\mathbb{P} \left( C_t(n, \epsilon) > \epsilon^{1/2-\zeta} \right) \leq A \exp(-\epsilon^{-2\zeta}) \quad \text{and} \quad \mathbb{P} \left( C_t(n, \epsilon) > n^{\epsilon-1/2} \right) \leq A \exp(-n^{2\zeta})
\text{(86)}
\]

where \( A = \max\{A_3, A_4\} \). The required result follows by choosing \( \nu = \max\{\frac{1}{\epsilon}, n\} \) in (86).

\textbf{APPENDIX C}

\textbf{DERIVATION OF (51) AND (60)}

Consider first the slow fading case, where the winning user is given by
\[
i_t = \arg \max_{i \in M_t} \lambda_i \left( W \log_2 \left( \frac{\hat{p}^2_i}{P_i} \right) \right) - c_i^2 \hat{p}^2_i
\text{(87)}
\]
where \( \hat{p}^2_i \) is given by (50). Thus, the objective function in (87) for a given \( \lambda \) can be written as
\[
T_i^l(\lambda) = \begin{cases} 
\lambda W \log_2 \left( C_{\min \lambda \frac{\gamma_i}{c_i^2}} \right) - C_{\min}, & \lambda \leq \frac{C_{\min}}{W} \\
\lambda W \log_2 \left( C_{\frac{\gamma_i}{c_i^2}} \right) - C_{\max}, & \lambda \geq \frac{C_{\min}}{W} \\
\lambda W \log_2 \left( \frac{\lambda W \gamma_i}{c_i^2} \right) - \lambda W, & \text{otherwise.}
\end{cases}
\text{(88)}
\]

Since \( \log_2 \) is monotonic function, observe in (88) that in all three cases, \( T_i^l(\lambda) \) depends monotonically on \( \gamma_i/c_i^2 \) for all \( \lambda > 0 \). This allows us to conclude that
\[
i_t = \arg \max_{i \in M_t} T_i^l(\lambda) = \arg \max_{i \in M_t} \frac{\gamma_i}{c_i^2}
\]
which is the required identity in (51). Similarly for the fast fading case, the objective function for the winning user in (60) is given by
\[
T_i^f(\lambda) = \begin{cases} 
\lambda W \log_2 \left( C_{\min \lambda \frac{\gamma_i}{c_i^2}} \right) + \lambda W \psi_i - C_{\min}, & \lambda \leq \frac{C_{\min}}{W} \\
\lambda W \log_2 \left( C_{\frac{\gamma_i}{c_i^2}} \right) + \lambda W \psi_i - C_{\max}, & \lambda \geq \frac{C_{\min}}{W} \\
\lambda W \log_2 \left( \frac{\lambda W \gamma_i}{c_i^2} \right) + \lambda W \psi_i - \lambda W, & \text{otherwise}
\end{cases}
\text{(89)}
\]
which, for $\lambda > 0$, again depends monotonically on $\log_2(\gamma_i^t/c_i^t) + \psi_i$. The expression in (60) therefore follows.

REFERENCES

[1] L. Georgiadis, M. J. Neely, and L. Tassiulas, “Resource allocation and cross-layer control in wireless networks,” Found. Trends. Network., vol. 1, no. 1, pp. 1–149, 2006.
[2] Z. Fan, P. Kulkarni, S. Gormus, C. Efthymiou, G. Kalogridis, M. Sooriyabandara, Z. Zhu, S. Lambotharan, and W. H. Chin, “Smart grid communications: overview of research challenges, solutions, and standardization activities,” IEEE Commun. Surveys Tuts., vol. 15, no. 1, pp. 21–38, 2013.
[3] R. Afolabi, A. Dadlani, and K. Kim, “Multicast Scheduling and Resource Allocation Algorithms for OFDMA-Based Systems: A Survey,” IEEE Commun. Surveys Tuts., vol. 15, no. 1, pp. 240–254, First 2013.
[4] X. Wang and N. Gao, “Stochastic resource allocation over fading multiple access and broadcast channels,” IEEE Trans. Inf. Theory, vol. 56, no. 5, pp. 2382–2391, 2010.
[5] A. Ribeiro and G. Giannakis, “Separation Principles in Wireless Networking,” IEEE Trans. Inf. Theory, vol. 56, no. 9, pp. 4488–4505, Sept. 2010.
[6] A. G. Marques, L. M. Lopez-Ramos, G. B. Giannakis, and J. Ramos, “Resource allocation for interweave and underlay CRs under probability-of-interference constraints,” IEEE J. Sel. Areas Commun., vol. 30, no. 10, pp. 1922–1933, 2012.
[7] K. Rajawat, N. Gatsis, and G. Giannakis, “Cross-Layer Designs in Coded Wireless Fading Networks With Multicast,” IEEE/ACM Trans. Netw., vol. 19, no. 5, pp. 1276–1289, Oct 2011.
[8] A. Ribeiro, “Ergodic Stochastic Optimization Algorithms for Wireless Communication and Networking,” IEEE Trans. Signal Process., vol. 58, no. 12, pp. 6369–6386, Dec. 2010.
[9] X. Wang and G. Giannakis, “Resource Allocation for Wireless Multiuser OFDM Networks,” IEEE Trans. Inf. Theory, vol. 57, no. 7, pp. 4359–4372, July 2011.
[10] A. Nedić and D. Bertsekas, “Convergence rate of Incremental Subgradient Algorithms,” in Stochastic Optimization: Algorithms and Applications. Springer, 2001, pp. 223–264.
[11] D. P. Bertsekas, “Incremental Gradient, Subgradient, and Proximal methods for Convex Optimization: A survey,” LIDS-P-2848, 2011.
[12] Z.-Q. Luo, “On the Convergence of the LMS Algorithm with Adaptive Learning Rate for Linear Feedforward Networks,” Neural Comput., vol. 3, no. 2, pp. 226–245, Jun. 1991.
[13] L. Bottou, “Online learning and stochastic approximations,” On-line learning in neural networks, vol. 17, no. 9.
[14] ——, “Stochastic gradient descent tricks,” in Neural Networks: Tricks of the Trade. Springer, 2012, pp. 421–436.
[15] E. Moulines and F. R. Bach, “Non-asymptotic analysis of stochastic approximation algorithms for machine learning,” in Adv. Neural Inf. Process. Syst., 2011, pp. 451–459.
[16] A. Nedic and A. Ozdaglar, “Approximate primal solutions and rate analysis for dual subgradient methods,” SIAM J. Optim., vol. 19, no. 4, pp. 1757–1780, 2009.
[17] M. Ji, G. Caire, and A. F. Molisch, “Wireless device-to-device caching networks: Basic principles and system performance,” IEEE J. Sel. Areas Commun., vol. 34, no. 1, pp. 176–189, 2016.
[18] N. Golrezaei, P. Mansourifard, A. F. Molisch, and A. G. Dimakis, “Base-station assisted device-to-device communications for high-throughput wireless video networks,” IEEE Trans. Wireless Commun., vol. 13, no. 7, pp. 3665–3676, 2014.
[19] H. Robbins and S. Monro, “A Stochastic Approximation Method,” Ann. Math. Statistics, vol. 22, no. 3, pp. 400–407, 09 1951.
[20] B. Widrow and S. D. Stearns, Adaptive Signal Processing. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1985.
[21] A. H. Sayed, Adaptive filters. John Wiley & Sons, 2011.
[22] L. Bottou, “Stochastic Gradient Learning in Neural Networks,” in *Proceedings of Neuro-Nîmes’91*. Nimes, France: EC2, 1991.

[23] H. J. Kushner and J. Yang, “Analysis of adaptive step size SA algorithms for parameter tracking,” in *Proc. of the IEEE Conf. on Decision and Control*, vol. 1, 1994, pp. 730–737.

[24] L. Bottou, “Large-scale machine learning with stochastic gradient descent,” in *Proc. of COMPSTAT*. Springer, 2010, pp. 177–186.

[25] S. Suvrit, S. Nowozin, and S. J. Wright, Eds., *Optimization for Machine Learning*. Cambridge, Mass.: MIT Press, 2012.

[26] F. Kelly, “Charging and rate control for elastic traffic,” *Euro. Trans. Telecommun.*, vol. 8, no. 1, pp. 33–37, 1997.

[27] T. Larsson, M. Patriksson, and A.-B. Strömberg, “Ergodic, primal convergence in dual subgradient schemes for convex programming,” *Math. Program.*, vol. 86, no. 2, pp. 283–312, 1999.

[28] N. Gatsis, A. Ribeiro, and G. Giannakis, “A class of convergent algorithms for resource allocation in wireless fading networks,” *IEEE Trans. Wireless Commun.*, vol. 9, no. 5, pp. 1808–1823, May 2010.

[29] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms: Stability and Performance*. Upper Saddle River, NJ, USA: Prentice-Hall, 1995.

[30] M. J. Neely, “Stochastic network optimization with application to communication and queueing systems,” *Synthesis Lectures on Commun. Netw.*, vol. 3, no. 1, pp. 1–211, 2010.

[31] H. Yu and M. J. Neely, “On the convergence time of the drift-plus-penalty algorithm for strongly convex programs,” in *Proc. of the IEEE Conf. on Decision and Control*, 2015, pp. 2673–2679.

[32] N. D. Sidropoulos, T. N. Davidson, and Z.-Q. T. Luo, “Transmit beamforming for physical-layer multicasting,” *IEEE Trans. Signal Process.*, vol. 54, no. 6, pp. 2239–2251, 2006.

[33] D. Palomar and M. Chiang, “A tutorial on decomposition methods for network utility maximization,” *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1439–1451, Aug 2006.

[34] N. Gatsis and A. G. Marques, “A stochastic approximation approach to load shedding in power networks,” in *Proc. of the IEEE ICASSP*, 2014, pp. 6464–6468.

[35] X. Wang, Y. Zhang, T. Chen, and G. B. Giannakis, “Dynamic energy management for smart-grid-powered coordinated multipoint systems,” *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1348–1359, 2016.

[36] X. Wang, T. Chen, X. Chen, X. Zhou, and G. B. Giannakis, “Dynamic resource allocation for smart-grid powered mimo downlink transmissions,” *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3354–3365, 2016.

[37] X. Wang, T. Ma, R. Zhang, and X. Zhou, “Stochastic online control for energy-harvesting wireless networks with battery imperfections,” *IEEE Trans. Wireless Commun.*, vol. 15, no. 12, pp. 8437–8448, 2016.

[38] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge University Press, 2005.

[39] N. Gatsis and G. Giannakis, “Residential Load Control: Distributed Scheduling and Convergence With Lost AMI Messages,” *IEEE Trans. Smart Grid*, vol. 3, no. 2, pp. 770–786, June 2012.

[40] K. B. Rao, M. B. Rao et al., “A remark on nonatomic measures,” *Ann. Math. Stat.*, vol. 43, no. 1, pp. 369–370, 1972.

[41] H. Hermes and J. P. Lasalle, *Functional analysis and time optimal control*. Academic Press, 1969.

[42] B. Shitovitz, “Oligopoly in markets with a continuum of traders,” *Econometrica*, pp. 467–501, 1973.

[43] Z.-Q. Luo and S. Zhang, “Dynamic spectrum management: Complexity and duality,” *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 57–73, 2008.

[44] K.-C. Toh and S. Yun, “An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems,” *Pacific J. Opt.*, vol. 6, no. 615-640, p. 15, 2010.

[45] B. Recht, C. Re, S. Wright, and F. Niu, “Hogwild: A lock-free approach to parallelizing stochastic gradient descent,” in *Advances in Neural Information Processing Systems*, 2011, pp. 693–701.
[46] V. V. Petrov, *Limit Theorems of Probability Theory*. Oxford Studies in Probability, 1995.

[47] D. K. Fuk and S. V. Nagaev, “Probability inequalities for sums of independent random variables,” *Theory of Probability & Its Applications*, vol. 16, no. 4, pp. 643–660, 1971.

[48] L. E. Baum, M. Katz, and R. R. Read, “Exponential convergence rates for the law of large numbers,” *Trans. Amer. Math. Soc.*, vol. 102, no. 2, pp. 187–199, 1962.