A distributed adaptive steplength stochastic approximation method for monotone stochastic Nash Games

Farzad Yousefian, Angelia Nedić, and Uday V. Shanbhag

Abstract—We consider a distributed stochastic approximation (SA) scheme for computing an equilibrium of a stochastic Nash game. Standard SA schemes employ diminishing steplength sequences that are square summable but not summable. Such requirements provide a little or no guidance for how to leverage Lipschitzian and monotonicity properties of the problem and naive choices (such as $\gamma_k = 1/k$) generally do not perform uniformly well on a breadth of problems. While a centralized adaptive stepsize SA scheme is proposed in [1] for the optimization framework, such a scheme provides no freedom for the agents in choosing their own stepsizes. Thus, a direct application of centralized stepsize schemes is impractical in solving Nash games. Furthermore, extensions to game-theoretic regimes where players may independently choose steplength sequences are limited to recent work by Koshal et al. [2]. Motivated by these shortcomings, we present a distributed algorithm in which each player updates his steplength based on the previous steplength and some problem parameters. The steplength rules are derived from minimizing an upper bound of the errors associated with players’ decisions. It is shown that these rules generate sequences that converge almost surely to an equilibrium of the stochastic Nash game. Importantly, variants of this rule are suggested where players independently select steplength sequences while abiding by an overall coordination requirement. Preliminary numerical results are seen to be promising.

I. INTRODUCTION

We consider a class of stochastic Nash games in which every player solves a stochastic convex program parametrized by adversarial strategies. Consider an $N$-person stochastic Nash game in which the $i$th player solves the parametrized convex problem

$$\min_{x \in X_i} E[f_i(x_i, x_{-i}, \xi_i)],$$

where $x_{-i}$ denotes the collection $\{x_j, j \neq i\}$ of decisions of all players other than player $i$. For each $i$, the vector $\xi_i : \Omega_i \rightarrow \mathbb{R}^{n_i}$ is a random vector with a probability distribution on some set, while the function $E[f_i(x_i, x_{-i}, \xi_i)]$ is strongly convex in $x_i$ for all $x_{-i} \in \prod_{j \neq i} X_j$. For every $i$, the set $X_i \subseteq \mathbb{R}^{n_i}$ is closed and convex. We focus on the resulting stochastic variational inequality (VI) and consider the development of distributed stochastic approximation schemes that rely on adaptive steplength sequences. Stochastic approximation techniques have a long tradition. First proposed by Robbins and Monro [3] for differentiable functions and Ermoliev [4]–[6], significant effort has been applied towards theoretical and algorithmic examination of such schemes (cf. [7], [8]). Yet, there has been markedly little on the application of such techniques to solution of stochastic variational inequalities. Exceptions include the work by Jiang and Xu [9], and more recently by Koshal et al. [2]. The latter, in particular, develops a single timescale stochastic approximation scheme for precisely the class of problems being studied here viz., monotone stochastic Nash games.

Standard stochastic approximation schemes provide little guidance regarding the choice of a steplength sequence, apart from requiring that the sequence, denoted by $\{\gamma_k\}$, satisfies $\sum_{k=1}^{\infty} \gamma_k = \infty$ and $\sum_{k=1}^{\infty} \gamma_k^2 < \infty$. This paper is motivated by the need to develop adaptive steplength sequences that can be independently chosen by players under a limited coordination, while guaranteeing the overall convergence of the scheme. Adaptive stepsizes have been effectively used in gradient and subgradient algorithms. Vrahatis et al. [10] presented a class of gradient algorithms with adaptive stepsizes for unconstrained minimization. Spall [11] developed a general adaptive SA algorithm based on using a simultaneous perturbation approach for estimating the Hessian matrix. Cicic et al. [12] considered the Kiefer-Wolfowitz (KW) SA algorithm and derived general upper bounds on its mean-squared error, together with an adaptive version of the KW algorithm. Ram et al. [13] considered distributed stochastic subgradient algorithms for convex optimization problems and studied the effects of stochastic errors on the convergence of the proposed algorithm. Lizarraga et al. [14] considered a family of two person Multi-Plant game and developed Stackelberg-Nash equilibrium conditions based on the Robust Maximum Principle. More recently, Yousefian et al. [1], [15] developed centralized adaptive stepsize SA schemes for solving stochastic optimization problems and variational inequalities. The main contribution of the current paper lies in developing a class of distributed adaptive stepsize rules for SA scheme in which each agent chooses its own stepsize without any specific information regarding the policy employed by agents. This degree of freedom in choosing the stepsizes has not been addressed in the centralized schemes.

Before proceeding, we briefly motivate the question of distributed computation of Nash equilibria from two different standpoints: (i) First, the Nash game can be viewed as a competitive analog of a stochastic multi-user convex optimization problem of the form $\min_{x \in X} \sum_{i=1}^{N} E[f_i(x_i, x_{-i}, \xi_i)]$. Furthermore, under the assumption that equilibria of the associated stochastic Nash game are efficient, our scheme...
provides a distributed framework for computing solutions to this problem. In such a setting, we may prescribe that players employ stochastic approximation schemes since the Nash game represents an engineered construct employed for computing solutions; (ii) A second perspective is one drawn from a bounded rationality approach towards distributed computation of Nash equilibria. A fully rational avenue for computing equilibria suggests that each player employs a best response mapping in updating strategies, based on what the competing players are doing. Yet, when faced by computational or time constraints, players may instead take a gradient step. We work in precisely this regime but allow for flexibility in terms of the steplengths chosen by the players.

In this paper, we consider the solution of a stochastic Nash game whose equilibria are completely captured by a stochastic variational inequality with a strongly monotone mapping. Motivated by the need for efficient distributed simulation methods for computing solutions to such problems, we present a distributed scheme in which each player employs an adaptive rule for prescribing steplengths. Importantly, these rules can be implemented with relatively little coordination by any given player and collectively lead to iterates that are shown to converge to the unique equilibrium in an almost-sure sense.

This paper is organized as follows. In Section II, we introduce the formulation of a stochastic Nash games in which every player solves a stochastic convex problem. In Section III, we show the almost-sure convergence of the SA algorithm under specified assumptions. In Section IV, motivated by minimizing a suitably defined error bound, we develop an adaptive steplength stochastic approximation framework in which every player updates its steplength by leveraging a rule that adapts to problem parameters. It is shown that the choice of adaptive steplength rules can be obtained independently by each player under a limited coordination. Finally, in Section V, we provide some numerical results from a stochastic flow management game drawn from a communication network setting. Note that the proofs of the results in this article are omitted and may be found in [16].

Notation: Throughout this paper, a vector \( x \) is assumed to be a column vector. We write \( x^T \) to denote the transpose of a vector \( x \). \( \| x \| \) denotes the Euclidean vector norm, i.e., \( \| x \| = \sqrt{x^T x} \). We use \( \Pi_X(x) \) to denote the Euclidean projection of a vector \( x \) on a set \( X \), i.e., \( \| x - \Pi_X(x) \| = \min_{y \in X} \| x - y \| \). Vector \( g \) is a subgradient of a convex function \( f \) with domain \( \text{dom} f \) at \( \bar{x} \in \text{dom} f \) when \( f(\bar{x}) + g^T (x - \bar{x}) \leq f(x) \) for all \( x \in \text{dom} f \). The set of all subgradients of \( f \) at \( \bar{x} \) is denoted by \( \partial f(\bar{x}) \). We write a.s. as the abbreviation for “almost surely”, and use \( \mathbb{E}[z] \) to denote the expectation of a random variable \( z \).

II. PROBLEM FORMULATION

In this section, we present (sufficient) conditions associated with equilibrium points of the stochastic Nash game defined by (1). The equilibrium conditions of this game can be characterized by a stochastic variational inequality problem denoted by VI(\( X, F \)), where

\[
F(x) \triangleq \left( \begin{array}{c}
\nabla x_1 \mathbb{E}[f_1(x, \xi_1)] \\
\vdots \\
\nabla x_N \mathbb{E}[f_N(x, \xi_N)]
\end{array} \right), \quad X = \prod_{i=1}^N X_i, \quad (2)
\]

with \( x \triangleq (x_1, \ldots, x_N)^T \) and \( x_i \in X_i \subseteq \mathbb{R}^{n_i} \) for \( i = 1, \ldots, N \). Given a set \( X \subseteq \mathbb{R}^n \) and a single-valued mapping \( F : X \rightarrow \mathbb{R}^n \), then a vector \( x^* \in X \) solves a variational inequality \( VI(X, F) \), if

\[
(x - x^*)^T F(x^*) \geq 0 \quad \text{for all} \quad x \in X.
\]

Let \( n = \sum_{i=1}^N n_i \), and note that when the sets \( X_i \) are convex and closed for all \( i \), the set \( X \in \mathbb{R}^n \) is closed and convex.

In the context of solving the stochastic variational inequality \( VI(X, F) \) in (2)-(3), suppose each player employs a stochastic approximation scheme for given by

\[
x_{k+1,i} = \Pi_{X_i}(x_{k,i} - \gamma_{k,i}(F_i(x_k) + w_{k,i})),
\]

\[
w_{k,i} \overset{i.i.d.}{=} F_i(x_k, \xi_k) - F_i(x_k),
\]

for all \( k \geq 0 \) and \( i = 1, \ldots, N \), where \( \gamma_{k,i} > 0 \) is the stepsize of the \( i \)-th player at iteration \( k \), \( x_k = (x_{k,1}, x_{k,2}, \ldots, x_{k,N})^T \), \( \xi_k = (\xi_{k,1}, \xi_{k,2}, \ldots, \xi_{k,N})^T \), \( F_i = \mathbb{E}[\nabla x_i f_i(x, \xi_i)] \), and

\[
\hat{F}(x, \xi) \triangleq \left( \begin{array}{c}
\nabla x_1 f_1(x, \xi_1) \\
\vdots \\
\nabla x_N f_N(x, \xi_N)
\end{array} \right), \quad \xi \triangleq \left( \begin{array}{c}
\xi_1 \\
\vdots \\
\xi_N
\end{array} \right).
\]

Note that in terms of the definition of \( w_{k,i}, F_i, \) and \( \hat{F}_i, \mathbb{E}[w_{k,i} \mid F_k] = 0 \). In addition, \( x_0 \in X \) is a random initial vector independent of the random variable \( \xi \) and such that \( \mathbb{E} [\| x_0 \|^2] < \infty \). Note that each player uses its individual stepsize to update its decision.

III. A DISTRIBUTED SA SCHEME

In this section, we present conditions under which algorithm (4) converges almost surely to the solution of game (1) under suitable assumptions on the mapping. Also, we develop a distributed variant of a standard stochastic approximation scheme and provide conditions on the steplength sequences that lead to almost-sure convergence of the iterates to the unique solution. Our assumptions include requirements on the set \( X \) and the mapping \( F \).

Assumption 1: Assume the following:

(a) The sets \( X_i \subseteq \mathbb{R}^{n_i} \) are closed and convex.

(b) \( F(x) \) is strongly monotone with constant \( \eta > 0 \) and Lipschitz continuous with constant \( L \) over the set \( X \).

Remark: The strong monotonicity is assumed to hold throughout the paper. Although the convergence results may still hold with a weaker assumption, such as strict monotonicity, but the stepsize policy in this paper leverages the strong monotonicity parameter which prescribes a parametrized stepsize rule. This is the main reason that we assumed the stronger version of monotonicity. In Section V, we present an example where such an assumption is satisfied.

Next we present an assumption of stepsizes employed by each player.

4773
Assumption 2: Assume that:

(a) The stepsize sequences are such that $\gamma_{k,i} > 0$ for all $k$ and $i$, with $\sum_{k=0}^{\infty} \gamma_{k,i} = \infty$ and $\sum_{k=0}^{\infty} \gamma_{k,i}^2 < \infty$.

(b) There exists a scalar $\beta$ such that $0 \leq \beta < \frac{2}{L}$ and $\frac{\gamma_{i_k} - \delta_k}{\gamma_{i_k}} \leq \beta$ for all $k \geq 0$, where $\delta_k$ and $\Gamma_k$ are (fixed) positive sequences satisfying $\delta_k \leq \min_{i=1,\ldots,N} \gamma_{k,i}$ and $\Gamma_k \geq \max_{i=1,\ldots,N} \gamma_{k,i}$ for all $k \geq 0$.

We let $F_k$ denote the history of the method up to time $k$, i.e., $F_k = \{x_0, \delta_0, \xi_1, \ldots, \xi_{k-1}\}$ for $k \geq 1$ and $F_0 = \{x_0\}$.

Consider the following assumption on the stochastic errors, $w_k$, of the algorithm.

Assumption 3: The errors $w_k$ are such that for some constant $\nu > 0$,

$$E[||w_k||^2 \mid F_k] \leq \nu^2 \quad \text{a.s. for all } k \geq 0.$$ 

We use the Robbins-Siegmund lemma in establishing the convergence of method (4), which can be found in [17] (cf. Lemma 10, page 49).

Lemma 1: Let $\{v_k\}$ be a sequence of nonnegative random variables, where $E[v_k] < \infty$, and let $\{\alpha_k\}$ and $\{\mu_k\}$ be deterministic scalar sequences such that:

$$E[v_{k+1} \mid v_0, \ldots, v_k] \leq (1 - \alpha_k)v_k + \mu_k \quad \text{a.s. for all } k \geq 0,$$

$$0 \leq \alpha_k \leq 1, \quad \mu_k \geq 0,$$

$$\sum_{k=0}^{\infty} \alpha_k = \infty, \quad \sum_{k=0}^{\infty} \mu_k < \infty, \quad \lim_{k \to \infty} \frac{\mu_k}{\alpha_k} = 0.$$

Then, $v_k \to 0$ almost surely.

The following lemma provides an error bound for algorithm (4) under Assumption 1.

Lemma 2: Consider algorithm (4). Let Assumption 1 hold. Then, the following relation holds a.s. for all $k \geq 0$:

$$E[||x_{k+1} - x^*||^2 \mid F_k] \leq \Gamma_k^2 E[||w_k||^2 \mid F_k]$$

$$+ (1 - 2\langle \eta + L \rangle \delta_k + 2L\Gamma_k + L^2 \Gamma_k^2)||x_k - x^*||^2. \quad (5)$$

We next prove that algorithm (4) generates a sequence of iterates that converges a.s. to the unique solution of VI$(X,F)$, as seen in the following proposition. Our proof of this result makes use of Lemma 2.

Proposition 1 (Almost-sure convergence): Consider the algorithm 4. Let Assumptions 1, 2 and 3 hold. Then,

(a) The following relation holds a.s. for all $k \geq 0$:

$$E[||x_{k+1} - x^*||^2] \leq (1 + \beta)^2 \delta_k^2 \nu^2$$

$$+ (1 - 2\langle \eta - \beta L \rangle \delta_k + (1 + \beta)^2 L^2 \delta_k^2)E[||x_k - x^*||^2]. \quad (6)$$

(b) The sequence $\{x_k\}$ generated by algorithm (4), converges a.s. to the unique solution of VI$(X,F)$.

Consider now a special form of algorithm (4) corresponding to the case where all players employ the same stepsize, i.e., $\gamma_{k,i} = \gamma_k$ for all $k$. Then, the algorithm (4) reduces to the following:

$$x_{k+1} = \Pi_X (x_k - \gamma_k(F(x_k) + w_k)), \quad w_k = \hat{F}(x_k, \xi_k) - F(x_k), \quad (6)$$

for all $k \geq 0$. Observe that when $\gamma_{k,i} = \gamma_k$ for all $k$, Assumption 2a is satisfied when $\sum_{k=0}^{\infty} \gamma_k = \infty$ and $\sum_{k=0}^{\infty} \gamma_k^2 < \infty$. Assumption 2b is automatically satisfied with $\Gamma_k = \delta_k = \gamma_k$ and $\beta = 0$. Hence, as a direct consequence of Proposition 1, we have the following corollary.

Corollary 1 (Identical stepsizes): Consider algorithm (6).

Let Assumption 1 and 3 hold. Also, let $\sum_{k=0}^{\infty} \gamma_k = \infty$ and $\sum_{k=0}^{\infty} \gamma_k^2 < \infty$. Then,

(a) The following relation holds almost surely:

$$E[||x_{k+1} - x^*||^2] \leq (1 - 2\langle \eta - \beta L \rangle \delta_k + (1 + \beta)^2 L^2 \delta_k^2)E[||x_k - x^*||^2] + \gamma_k^2 \nu^2.$$

(b) The sequence $\{x_k\}$ generated by algorithm (6), converges a.s. to the unique solution of VI$(X,F)$.

IV. A DISTRIBUTED ADAPTIVE STEPLength SA SCHEME

Stochastic approximation algorithms require stepsize sequences to be square summable but not summable. These algorithms provide little advice regarding the choice of such sequences. One of the most common choices has been the harmonic steplength rule which takes the form of $\gamma_k = \frac{\theta}{k}$ where $\theta > 0$ is a constant. Although, this choice guarantees almost-sure convergence, it does not leverage problem parameters. Numerically, it has been observed that such choices can perform quite poorly in practice. Motivated by this shortcoming, we present a distributed adaptive steplength scheme for algorithm (4) which guarantees almost-sure convergence of $x_k$ to the unique solution of VI$(X,F)$. It is derived from the minimizer of a suitably defined error bound and leads to a recursive relation; more specifically, at each step, the new stepsize is calculated using the stepsize from the preceding iteration and problem parameters. To begin our analysis, we consider the result of Proposition 1a for all $k \geq 0$:

$$E[||x_{k+1} - x^*||^2] \leq (1 + \beta)^2 \delta_k^2 \nu^2$$

$$+ (1 - 2\langle \eta - \beta L \rangle \delta_k + (1 + \beta)^2 L^2 \delta_k^2)E[||x_k - x^*||^2]. \quad (7)$$

When the stepsizes are further restricted so that

$$0 < \delta_k \leq \frac{\theta - \beta L}{(1 + \beta)^2 L^2},$$

we have

$$1 - 2\langle \eta - \beta L \rangle \delta_k + (1 + \beta)^2 L^2 \delta_k^2 \leq 1 - (\eta - \beta L)\delta_k.$$ Thus, for $0 < \delta_k \leq \frac{\theta - \beta L}{(1 + \beta)^2 L^2}$, from inequality (7) we obtain

$$E[||x_{k+1} - x^*||^2] \leq (1 - (\eta - \beta L)\delta_k)E[||x_k - x^*||^2]$$

$$+ (1 + \beta)^2 \delta_k^2 \nu^2 \quad \text{for all } k \geq 0. \quad (8)$$

Let us view the quantity $E[||x_{k+1} - x^*||^2]$ as an error $e_{k+1}$ of the method arising from the use of the stepsize values $\delta_0, \delta_1, \ldots, \delta_k$. Relation (8) gives us an estimate of the error of algorithm (4). We use this estimate to develop an adaptive steplength procedure. Consider the worst case which is the case when (8) holds with equality. In this worst case, the error satisfies the following recursive relation:

$$e_{k+1} = (1 - (\eta - \beta L)\delta_k)e_k + (1 + \beta)^2 \delta_k^2 \nu^2.$$ Let us assume that we want to run the algorithm (4) for a fixed number of iterations, say $K$. The preceding relation shows that $e_K$ depends on the stepsize values up
to the $K$th iteration. This motivates us to see the stepsize parameters as decision variables that can minimize a suitably defined error bound of the algorithm. Thus, the variables are $\delta_0, \delta_1, \ldots, \delta_{K-1}$ and the objective function is the error function $e_K(\delta_0, \delta_1, \ldots, \delta_{K-1})$. We proceed to derive a stepsize rule by minimizing the error $e_{K+1}$. Importantly, $\delta_{K+1}$ can be shown to be a function of only the most recent stepsize $\delta_K$. We define the real-valued error function $e_k(\delta_0, \delta_1, \ldots, \delta_{k-1})$ by the upper bound in (8):

$$e_{k+1}(\delta_0, \ldots, \delta_k) \triangleq \left(1 - \frac{\eta - \beta L}{2} \right) e_k(\delta_0, \ldots, \delta_{k-1}) + (1 + \beta^2) \nu^2$$

for all $k \geq 0$. (9)

where $e_0$ is a positive scalar, $\eta$ is the strong monotonicity parameter and $\nu^2$ is the upper bound for the conditional second moments of the error norms $|w_k|$. Now, let us consider the stepsize sequence $\{\delta_k^*\}$ given by

$$\delta_k^* = \frac{\eta - \beta L}{2(1 + \beta)^2} e_0$$

$$\delta_k^* = \delta_{k-1} \left(1 - \frac{\eta - \beta L}{2} \delta_{k-1}^* \right)$$

for all $k \geq 1$. (11)

In what follows, we often abbreviate $e_k(\delta_0, \ldots, \delta_{k-1})$ by $e_k$ whenever this is unambiguous. The next proposition shows that the lower bound sequence of $\gamma_{k,i}^*$ is defined as in (9), where $e_0 > 0$ is such that $e_0 < \frac{2\nu^2}{L^2}$, and $L$ is the Lipschitz constant of mapping $F$. Let the sequence $\{\delta_k^*\}$ be given by (10)–(11). Then, the following hold:

(a) $e_k(\delta_0^*, \ldots, \delta_k^*) = \frac{(1 + \beta)^2}{\eta - \beta L} \delta_k^* \nu^2$ for all $k \geq 0$.

(b) For any $k \geq 1$, the vector $\{\delta_0^*, \delta_1^*, \ldots, \delta_{k-1}^*\}$ is the minimizer of the function $e_k(\delta_0, \ldots, \delta_{k-1})$ over the set

$$G_k \triangleq \left\{ \alpha \in \mathbb{R}^k : 0 < \alpha_j \leq \frac{\eta - \beta L}{(1 + \beta)^2 \nu^2}, j = 1, \ldots, k \right\},$$

i.e., for any $k \geq 1$ and $(\delta_0, \ldots, \delta_{k-1}) \in G_k$:

$$e_k(\delta_0, \ldots, \delta_{k-1}) - e_k(\delta_0^*, \ldots, \delta_{k-1}^*) \geq (1 + \beta)^2 \nu^2(\delta_{k-1} - \delta_{k-1}^*)^2.$$

We have just provided an analysis in terms of the lower bound sequence $\{\delta_k^*\}$. We can conduct a similar analysis for $\{\Gamma_k\}$ and obtain the corresponding adaptive stepsize scheme using the following relation:

$$E[|x_{k+1} - x^*|^2] \leq \Gamma_k^2 e_0^2$$

$$+ (1 - \frac{(\eta + \beta L)}{1 + \beta} \Gamma_k + 2L \Gamma_k + L^2 \Gamma_k^2) E[|x_k - x^*|^2].$$

When $0 < \Gamma_k \leq \frac{\eta - \beta L}{(1 + \beta)^2 L^2}$, we have

$$E[|x_{k+1} - x^*|^2] \leq (1 - \frac{(\eta - \beta L)}{1 + \beta} \Gamma_k) E[|x_k - x^*|^2]$$

$$+ \Gamma_k^2 e_0^2$$

for all $k \geq 0$. (12)

Using relation (12) and following similar approach in Proposition 2, we obtain the sequence $\{\Gamma_k^*\}$ given by

$$\Gamma_0^* = \frac{\eta - \beta L}{2(1 + \beta)^2} e_0$$

$$\Gamma_k^* = \Gamma_{k-1}^* \left(1 - \frac{\eta - \beta L}{2(1 + \beta)} \Gamma_{k-1}^* \right)$$

for all $k \geq 1$. (14)

Note that the adaptive stepsize scheme given by (13)–(14) converges to zero and moreover, it is notsummable but squared summable (cf. [1], Proposition 3). In the following lemma, we derive a relation between two recursive sequences, which we use later to obtain our main recursive stepsize scheme.

**Lemma 3:** Suppose that sequences $\{\lambda_k\}$ and $\{\gamma_k\}$ are based on the following recursive equations for all $k \geq 0$,

$$\lambda_{k+1} = \lambda_k(1 - \delta_k), \quad \text{and} \quad \gamma_{k+1} = \gamma_k(1 - c\gamma_k),$$

where $\lambda_0 = c\gamma_0, 0 < \gamma_0 < \frac{1}{2},$ and $c > 0$. Then for all $k \geq 0$,

$$\lambda_k = c\gamma_k.$$

Next, we show a relation for the sequences $\{\delta_k^*\}$ and $\{\Gamma_k^*\}$.

**Lemma 4:** Suppose that sequences $\{\delta_k^*\}$ and $\{\Gamma_k^*\}$ are given by relations (10)–(11) and (13)–(14) and $e_0 < \frac{\nu^2}{L^2}$. Then for all $k \geq 0$, $\Gamma_k^* = (1 + \beta)\delta_k^*$. The earlier set of results are essentially adaptive rules for determining the upper and lower bound of stepsize sequences, i.e. $\{\delta_k^*\}$ and $\{\Gamma_k^*\}$. The next proposition proposes recursive stepsize schemes for each player of game (1).

**Proposition 3:** [Distributed adaptive steplength SA rules] Suppose that Assumptions 1 and 3 hold. Assume that the set $X$ is bounded, i.e. there exists a positive constant $D \triangleq \max_{x,y \in X} \|x - y\|$. Suppose that the stepizes for any player $i = 1, \ldots, N$ are given by the following recursive equations

$$\gamma_{0,i} = r_i \frac{c - \gamma_{k-1,i}}{(1 + c_0\gamma_{k-1,i})^2} \nu^2 D^2$$

$$\gamma_{k+1} = \gamma_{k-1,i} \left(1 - \frac{c_0\gamma_{k-1,i}}{r_i} \right)$$

for all $k \geq 1$. (16)

where $r_i$ is an arbitrary parameter associated with ith player such that $r_i \in [1, 1 + \frac{\nu^2}{L^2}]$, $c$ is an arbitrary fixed constant $0 < c < \frac{1}{2}$, $L$ denotes the Lipschitz constant of mapping $F$, and $\nu$ is the upper bound given by Assumption 3 such that $D < \sqrt{\frac{\nu^2}{L}}$. Then, the following hold:

(a) $\frac{2\gamma_{k+1}}{r_i}$ for any $i, j = 1, \ldots, N$ and $k \geq 0$.

(b) Assumption 2b holds with $\beta = \frac{2c_0\gamma_{k-1,i}}{r_i}$, $\delta_k^* = \Gamma_k^* = \frac{e_0}{D^2}$, and $e_0 = D^2$, where $\delta_k^*$ and $\Gamma_k^*$ are given by (10)–(11) and (13)–(14) respectively.

(c) The sequence $\{x_k\}$ generated by algorithm (4) converges a.s. to the unique solution of stochastic VI($X, F$).

(d) The results of Proposition 2 hold for $\delta_k^*$ when $e_0 = D^2$.

V. Numerical results

In this section, we report the results of our numerical experiments on a stochastic bandwidth-sharing problem in communication networks (Sec. V-A). We compare the performance of the distributed adaptive stepsize SA scheme

4775
(DASA) given by (15)-(16) with that of SA schemes with harmonic stepsize sequences (HSA), where agents use the stepsize \(\frac{\theta}{k}\) at iteration \(k\). More precisely, we consider three different values of the parameter \(\theta\), i.e., \(\theta = 0.1, 1,\) and \(10\). This diversity of choices allows us to observe the sensitivity of the HSA scheme to different settings of the parameters.

**A. A bandwidth-sharing problem in computer networks**

We consider a communication network where users compete for the bandwidth. Such a problem can be captured by an optimization framework (cf. [18]). Motivated by this model, we consider a network with 16 nodes, 20 links and 5 users. Figure 1 shows the configuration of this network. Users have access to different routes as shown in Figure 1.

For example, user 1 can access routes 1, 2, and 3. Each user is characterized by a cost function. Additionally, there is a congestion cost function that depends on the aggregate flow. More specifically, the cost function user \(i\) with flow rate (bandwidth) \(x_i\) is defined by

\[
f_i(x_i, \xi_i) \triangleq - \sum_{r \in \mathcal{R}(i)} \xi_i(r) \log(1 + x_i(r)),
\]

for \(i = 1, \ldots, 5\), where \(x \triangleq (x_1; \ldots; x_5)\) is the flow decision vector of the users, \(\xi \triangleq (\xi_1; \ldots; \xi_5)\) is a random parameter corresponding to the different users, \(\mathcal{R}(i) = \{1, 2, \ldots, n_i\}\) is the set of routes assigned to the \(i\)-th user, \(x_i(r)\) and \(\xi_i(r)\) are the \(r\)-th element of the decision vector \(x_i\) and the random vector \(\xi_i\), respectively. We assume that \(\xi_i(r)\) is drawn from a uniform distribution for each \(i\) and \(r\) and the links have limited capacities given by \(b\).

We may define the routing matrix \(A\) that describes the relation between set of routes \(\mathcal{R} = \{1, 2, \ldots, 9\}\) and set of links \(\mathcal{L} = \{1, 2, \ldots, 20\}\). Assume that \(A_{ir} = 1\) if route \(r \in \mathcal{R}\) goes through link \(l \in \mathcal{L}\) and \(A_{ir} = 0\) otherwise. Using this matrix, the capacity constraints of the links can be described by \(Ax \leq b\).

We formulate this model as a stochastic optimization problem given by

\[
\begin{aligned}
\text{minimize} & \quad \sum_{i=1}^{N} E[f_i(x_i, \xi_i)] + c(x) \\
\text{subject to} & \quad Ax \leq b, \quad x \geq 0,
\end{aligned}
\]

where \(c(x)\) is the network congestion cost. We consider this cost of the form \(c(x) = \|Ax\|^2\). Problem (17) is a convex optimization problem and the optimality conditions can be stated as a variational inequality given by \(\nabla f(x^*)^T(x - x^*) \geq 0\), where \(f(x) \triangleq \sum_{i=1}^{N} E[f_i(x_i, \xi_i)] + c(x)\). Using our notation in Sec. II, we have

\[
F(x) = - \left( \frac{\xi_i(1)}{1 + x_i(1)}, \ldots; \frac{\xi_5(2)}{1 + x_5(2)} \right) + 2A^T Ax,
\]

where \(\xi_i(r_i) \triangleq E[\xi_i(r_i)]\) for any \(i = 1, \ldots, 5\) and \(r_i = 1, \ldots, n_i\). It can be shown that the mapping \(F\) is strongly monotone and Lipschitz with specified parameters (cf. [16]).

We solve the bandwidth-sharing problem for 12 different settings of parameters shown in Table 1. We consider 4 parameters in our model that scale the problem. Here, \(m_b\) denotes the multiplier of the capacity vector \(b\), \(m_c\) denotes the multiplier of the congestion cost function \(c(x)\), and \(m_\xi\) and \(d_\xi\) are two multipliers that parameterize the random variable \(\xi\). \(S(i)\) denotes the \(i\)-th setting of parameters. For each of these 4 parameters, we consider 3 settings where one parameter changes and other parameters are fixed. This allows us to observe the sensitivity of the algorithms with respect to each of these parameters. The SA algorithms are terminated after 4000 iterates. To measure the error of the schemes, we run each scheme 25 times and then compute the mean squared error (MSE) using the metric \(\frac{1}{25} \sum_{i=1}^{25} \|x_k^i - x^*\|^2\) for any \(k = 1, \ldots, 4000\), where \(i\) denotes the \(i\)-th sample. Table II and III show the 90% confidence intervals (CIs) of the error for the DASA and HSA schemes.

<table>
<thead>
<tr>
<th>(i)</th>
<th>(S(i))</th>
<th>DASA 90% CI</th>
<th>HSA with # = 0.1 90% CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[2.97e-5, 4.95e-5]</td>
<td>[1.52e-5, 2.37e-5]</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>[2.97e-5, 4.95e-5]</td>
<td>[1.52e-5, 2.37e-5]</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>[1.52e-5, 2.95e-5]</td>
<td>[2.10e-5, 3.90e-5]</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>[3.90e-5, 7.55e-5]</td>
<td>[1.35e-5, 3.10e-5]</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>[1.20e-5, 1.87e-5]</td>
<td>[9.00e-6, 1.37e-5]</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>[3.40e-5, 6.35e-5]</td>
<td>[2.26e-5, 4.25e-5]</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>[1.80e-5, 4.83e-5]</td>
<td>[7.92e-6, 1.49e-5]</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>[1.80e-5, 4.83e-5]</td>
<td>[3.40e-6, 6.35e-5]</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>[1.80e-5, 4.83e-5]</td>
<td>[1.20e-6, 3.90e-5]</td>
<td></td>
</tr>
<tr>
<td>d_\xi</td>
<td>[0.50e-5, 0.91e-4]</td>
<td>[0.84e-6, 1.29e-5]</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>[0.84e-6, 1.29e-5]</td>
<td>[2.84e-6, 3.65e-5]</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>[0.84e-6, 1.29e-5]</td>
<td>[2.84e-6, 3.65e-5]</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>[0.84e-6, 1.29e-5]</td>
<td>[2.84e-6, 3.65e-5]</td>
<td></td>
</tr>
</tbody>
</table>

**TABLE I: Parameter settings**

**TABLE II: 90% CIs for DASA and HSA schemes – Part I**

**Insights:** We observe that DASA scheme performs well and is far more robust in comparison with the HSA schemes.
by different choices of $\theta$. Importantly, in most settings, DASA stands close to the HSA scheme in terms of the MSE. Note that when $\theta = 1$ or $\theta = 10$, the stepsize $\beta_k$ is not within the interval $(\frac{1}{(1+\beta_k)^2}, \frac{1}{\beta_k})$ for small $k$ and is not feasible in the sense of Prop. 2. Comparing the performance of each HSA scheme in different settings, we observe that HSA schemes are fairly sensitive to the choice of parameters. For example, HSA with $\theta = 0.1$ performs very well in settings $S(1)$, $S(2)$, and $S(3)$, while its performance deteriorates in settings $S(10)$, $S(11)$, and $S(12)$. A similar discussion holds for other two HSA schemes. A good instance of this argument is shown in Figure 2 and 3.

<table>
<thead>
<tr>
<th>$S(1)$</th>
<th>HSA w/ $\theta = 1$</th>
<th>90% CI</th>
<th>HSA w/ $\theta = 10$</th>
<th>90% CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$[1.70 \pm 0.2, 6.97 \pm 0.6]$</td>
<td>$[1.33 \pm 0.1, 8.14 \pm 0.5]$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>$[1.70 \pm 0.2, 6.97 \pm 0.6]$</td>
<td>$[1.33 \pm 0.1, 8.14 \pm 0.5]$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$[4.66 \pm 0.1, 1.76 \pm 0.1]$</td>
<td>$[8.97 \pm 0.1, 4.35 \pm 0.1]$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TABLE III: 90% CIs for DASA and HSA schemes – Part II

Fig. 2: DASA vs. HSA schemes – Setting $S(4)$

Fig. 3: DASA vs. HSA schemes – Setting $S(11)$

VI. CONCLUDING REMARKS

We considered distributed monotone stochastic Nash games where each player minimizes a convex function on a closed convex set. We first formulated the problem as a stochastic VI and then showed that under suitable conditions, for a strongly monotone and Lipschitz mapping, the SA scheme guarantees almost-sure convergence to the solution.

Next, motivated by the naive stepsize choices of standard SA schemes, we proposed a class of distributed adaptive steplengh rules where each player can choose his own stepsize independent of the other players from a specified range. We showed that this scheme provides almost-sure convergence and also minimizes a suitably defined error bound of the SA algorithm. Numerical experiments reported in Section V suggest that the scheme performs well on a broad set of problem settings.

REFERENCES