Networked Aggregative Games with Linear Convergence

Rongping Zhu, Jiaqi Zhang and Keyou You

Abstract—This paper considers a networked aggregative game (NAG) where the players are distributed over a communication network. By only communicating with a subset of players, the goal of each player in the NAG is to minimize an individual cost function that depends on its own action and the aggregate of all the players’ actions. To this end, we design a novel distributed algorithm that jointly exploits the ideas of the consensus algorithm and the conditional projection descent. Under strongly monotone assumption on the pseudo-gradient mapping, the proposed algorithm with fixed step-sizes is proved to converge linearly to the unique Nash equilibrium of the NAG. Then the theoretical results are validated by numerical experiments.

I. INTRODUCTION

An aggregative game is a non-cooperative Nash game where each player’s cost function depends on its own action and the aggregate of all players’ actions. Such games are elaborately introduced in [1] and [2], and have been widely applied in resource allocation [3]–[5], demand response [6], [7], etc. This work considers Networked Aggregative Games (NAGs) where players are distributed over a peer-to-peer (P2P) communication network and no central coordinator is included. To reach the Nash equilibrium (NE) of an NAG, players only communicates with their neighbors that are defined over the network to mitigate the lack of global information about the aggregate action. Note that the NAG is still a non-cooperative game in the sense that each player selfishly minimizes its local cost function.

Distributed seeking of NE has extensively been studied in recent years. For games where players’ cost functions depend on others’ actions in a general manner, researchers have proposed fully distributed algorithms based on alternating direction method of multipliers [8], best-response dynamics [9], and projected-gradient method [10]–[12]. In these algorithms, players utilize local communications to estimate others’ actions, and the NE is found by solving equivalent variational inequality problems [13]. Linear convergence to the NE is proved for these algorithms when the games possess strongly monotone pseudo-gradient mappings, which coincides with the results of centralized algorithms [13].

However, the above mentioned works require each player to estimate all the other players’ actions and exchange its estimates with neighbors, which is not scalable with the network size and results in large communication overhead for NAGs. Clearly, it is not necessary here as we only need the aggregate action. Moreover, each player in each iteration updates from an inferior point rather than its own action. The so-called inferior point is obtained from the mixture of each player’s own action and neighbor’s estimates, which is essential for the consensus part. Thus, these algorithms may get stuck in the beginning phase with slow convergence rate, which is also confirmed in our numerical experiments in Section IV.

By exploiting the NAG’s structure, distributed algorithms with communication-efficiency have been designed. Researchers in [14] and [15] adopted a consensus protocol to dynamically track the aggregate action and used projected-gradient methods to update each player’s action. Since only aggregate estimates are exchanged, compared with the algorithms in [8]–[12], there is an $n$-fold reduction in the amount of data that is transmitted in each iteration, where $n$ is the number of players. However, the convergence was shown only for diminishing step-sizes with strictly monotone pseudo-gradient mappings. Authors in [16] proposed an asynchronous consensus-based algorithms and numerically showed the potential of linear convergence for fixed step-sizes. Nonetheless, theoretical guarantee of linear convergence of NAG still remains an open question [17].

In this work, we propose a novel linearly convergent algorithm to find an NE of the NAGs. The proposed algorithm consists of three key components: (a) a consensus term to dynamically track the aggregate action; (b) a projected-gradient step to compute a feasible direction; and (c) a step along the feasible direction to update the actions. Then, the linear convergence is proved via establishing a linear system inequality. To the best of our knowledge, our algorithm is the first one that is efficient for NAGs and converges linearily. Finally, numerical experiments confirm that our algorithm outperforms the existing pseudo-gradient based ones.

The rest of this paper is organized as follows. In Section II, we introduce the NAGs and some preliminaries. Section III devotes to describing our algorithm with the convergence analysis. In Section IV, we validate the proposed algorithm on a class of Nash-Cournot games. Concluding remarks are drawn in Section V.

Notations: Throughout the paper, we use $x^T$ and $X^T$ to denote the transposes of a vector $x$ and a matrix $X$, respectively. Let $\|x\| = \sqrt{x^T x}$ denote the Euclidean norm of $x$. The inner product of matrices $X$ and $Y$ is denoted by $\langle X, Y \rangle_F \triangleq \text{tr}(X^T Y)$, where $\text{tr}(\cdot)$ denotes the trace of a matrix. Then, the Frobenius norm of $X$ is defined as $\|X\|_F \triangleq \sqrt{\langle X, X \rangle_F}$. We use $[x_1, \ldots, x_n]$ and $[x_1; \ldots; x_n]$ to denote the horizontal and vertical stack of vectors $x_1, \ldots, x_n$.
respectively. For matrix $X$, we write $A_{ij}$ to denote its $(i, j)$-th element and $A_{j}$ to denote its $j$-th column vector. $I$ denotes the vector with all entries equal to 1. Finally, we use $\Pi_{X}[\cdot]$ to denote the Euclidean projection of a vector to a closed convex set $X$, i.e., $\Pi_{X}[x] \triangleq \arg\min_{z \in X} \|x - z\|$. 

II. PRELIMINARIES

In this section, we first introduce the NAGs over a multi-agent network. Then, we reformulate the NAG as a variational inequality which leads to our projection-based algorithm design. Finally, we introduce some common assumptions.

A. Networked Aggregative Games

Consider a set of $n$ players indexed by $\mathcal{N} = \{1, 2, \cdots, n\}$. In an aggregative game, player $i$’s cost function $f_i(x_i, \bar{x})$ depends on its own action $x_i \in \mathcal{X}_i \subset \mathbb{R}^p$ and the aggregate action $\bar{x} = \frac{1}{n} \sum_{i = 1}^{n} x_i$. The objective of player $i$ in the AG is to minimize its local cost:

$$
\begin{align*}
\text{minimize} & \quad f_i(x_i, \bar{x}), \\
\text{subject to} & \quad x_i \in \mathcal{X}_i
\end{align*}
$$

where $x_j, j \neq i, j \in \mathcal{N}$ are fixed, and to reach an NE among all players [18].

Definition 1 (Nash Equilibrium). An $n$-tuple of actions $x^* = [x_1^*; x_2^*; \cdots; x_n^*]$ is a Nash equilibrium (NE) of the AG \([1]\) if for all $i \in \mathcal{N}$ and $x_i \in \mathcal{X}_i$,

$$
f_i(x_i^*, x^*) \leq f_i(x_i, \frac{1}{n} x_i + \frac{1}{n} \sum_{j \neq i}^{n} x_j^*).
$$

This work considers the NAG where the players are distributed over a peer-to-peer (P2P) network and have no direct access to the aggregate action. Instead, every player can only communicate with neighboring players defined by the network. We model the P2P network by an undirected graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$ and a weight matrix $W$, where $\mathcal{N}$ is the set of players and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is the edge set of communications, i.e., $(i, j) \in \mathcal{E}$ if and only if player $i$ can exchange information directly with player $j$. The weight matrix $W = \{w_{ij}\}_{n \times n}$ satisfies that $w_{ij} > 0$ if $(i, j) \in \mathcal{E}$ or $i = j$, and $w_{ij} = 0$ otherwise. The neighbors of player $i$ is a set $\{j|(i, j) \in \mathcal{E}\}$.

B. Variational Inequality Approach

The following assumption on the action sets and cost functions is common in the literature [13].

Assumption 1. For each player $i \in \mathcal{N}$, the action set $\mathcal{X}_i$ is compact and convex. Each local cost function $f_i(x_i, z)$ is continuously differentiable in $(x_i, z)$ over some open set containing $\mathcal{X}_i \times \overline{\mathcal{X}}$, where

$$
\overline{\mathcal{X}} = \{ \frac{1}{n} \sum_{i = 1}^{n} x_i | x_i \in \mathcal{X}_i \}.
$$

Moreover, the function $x_i \rightarrow f_i(x_i, \frac{1}{n} x_i + \bar{x}_{-i})$ is convex in $x_i$ over $\mathcal{X}_i$ for any fixed $\bar{x}_{-i}$.

Under Assumption $[1]$ an NE of the AG $[1]$ can be reached by solving a variational inequality [13], which is to find an $x^* \in \mathcal{X}$ such that

$$
(x - x^*)^T \phi(x^*) \geq 0, \quad \forall x \in \mathcal{X},
$$

where $\mathcal{X} = \prod_{i = 1}^{n} \mathcal{X}_i$ and $\phi(x)$ is the pseudo-gradient mapping defined as

$$
\phi(x) = [\nabla_{x_1} f_1(x_1, \bar{x}); \nabla_{x_2} f_2(x_2, \bar{x}); \cdots; \nabla_{x_n} f_n(x_n, \bar{x})].
$$

To emphasize the coordinate of the pseudo-gradient, we define the mapping

$$
F(x, z) = [F_1(x_1, z); F_2(x_2, z); \cdots; F_n(x_n, z)],
$$

where

$$
F_i(x_i, z) = \nabla_{x_i} f_i(x_i, z) + \frac{1}{n} \nabla_{x_i} f_i(x_i, z).
$$

Clearly, we have $\phi(x) = F(x, \bar{x})$.

C. Some Assumptions and Lemmas

We first introduce the two assumptions on the pseudo-gradient mapping, which are necessary for the linear convergence of projection type algorithms even in the centralized setting [13], and are commonly used in the literature, see e.g., [8]–[12].

Assumption 2 (Strong Monotonicity). The pseudo-gradient $\phi$ is strongly monotone over $\mathcal{X}$, i.e., there exists a constant $\mu > 0$ such that

$$
(x - x')^T (\phi(x) - \phi(x')) \geq \mu \|x - x'\|^2, \forall x, x' \in \mathcal{X}.
$$

Assumption 3 (Lipschitz Continuity). Each mapping $F_i$ is uniformly Lipschitz continuous over $\mathcal{X}_i \times \overline{\mathcal{X}}$, i.e., there exists a constant $L > 0$ such that for any $x_i, x_i' \in \mathcal{X}_i$ and $z, z' \in \overline{\mathcal{X}}$, it holds

$$
\|F_i(x_i, z) - F_i(x_i', z')\| \leq L \| [x; z] - [x'; z'] \|.
$$

Lemma 1 (Thm. 2.3.3 in [13]). Under Assumptions $[2]$ and $[3]$ the AG in $[1]$ has a unique NE.

Next, we state an assumption on the communication network.

Assumption 4 (Communication Network).

1) The graph $\mathcal{G}$ is connected, i.e., there exists a sequence of consecutive edges to connect any pair of players.

2) The weight matrix $W$ is doubly stochastic, i.e., $W \mathbf{1} = W^T \mathbf{1} = \mathbf{1}$.

The first part of Assumption $[4]$ ensures that every player’s action can affect every other players eventually, which is necessary in the distributed problems. The second part induces the averaging property of the weight matrix (see Lemma $[2]$), and is common in solving distributed problems, see, e.g., [19], [20]. The selection of such matrices has been extensively studied in the literature [21]–[23].

Lemma 2 (Thm. 5.1 in [21]). Under Assumption $[4]$ there exists a constant $\sigma \in (0, 1)$ which is the spectral norm of
Algorithm 1: The distributed algorithm for the NAG

Initialization: for all $i \in \mathcal{N}$, set $x_i^0 \in X_i$ and $v_i^0 = x_i^0$.

Repeat until convergence: for all $i \in \mathcal{N}$,

\begin{align}
\hat{x}_i^{k+1} &= \sum_{j=1}^n w_{ij} v_j^k, \quad (4a) \\
x_i^{k+1} &= \Pi_{X_i}[x_i^k - \alpha F_i(x_i^k, \hat{x}_i^{k+1})], \quad (4b) \\
v_i^{k+1} &= \hat{x}_i^{k+1} + x_i^{k+1} - x_i^k. \quad (4d)
\end{align}

$W - \frac{1}{n}11^T$, or the second largest singular value of $W$, such that for any $w \in \mathbb{R}^n$,

$$||Ww - 1\hat{w}|| = ||(W - \frac{1}{n}11^T)(w - 1\hat{w})|| \leq \sigma ||w - 1\hat{w}||,$$

where $\hat{w} = \frac{1}{n}11^T w$.

We also use the following property of projection operators.

Lemma 3 (Prop. 1.1.4 in [24]). Let $\mathcal{X} \subset \mathbb{R}^p$ be a nonempty closed convex set. Then for any $x, y \in \mathbb{R}^p$, we have that $||\Pi_{\mathcal{X}}[x] - \Pi_{\mathcal{Y}}[y]|| \leq ||x - y||$.

III. MAIN RESULTS

In this section, we propose a novel distributed algorithm for the NAG in [1] and prove its linear convergence.

A. Algorithm Design

It follows from Section II-B that the NAG corresponds to the variational inequality in (2). We recall a standard centralized algorithm obtained by a fixed-point iteration:

$$x_i^{k+1} = \Pi_{X_i}[x_i^k - \alpha F_i(x_i^k, \hat{x}_i^{k+1})], \quad \forall i \in \mathcal{N}. \quad (3)$$

If the step-size $\alpha$ is chosen small enough, the sequences $\{x_i^k\}_{k \in \mathbb{N}}$ converges to a fixed point of the mapping $x \mapsto \Pi_{X_i}[x - \alpha \phi(x)]$, which is exactly $x^*$ [13].

However, as mentioned in the previous section, the players in the NAG have no direct access to the aggregate action $\hat{x}$. Thus, for all $i \in \mathcal{N}$, let player $i$ maintain a local vector $v_i$ and exchange it with its neighbors to estimate the aggregate action. Then, they use the estimated aggregate action to locally update their actions. The details are summarized in Algorithm 1 where $\alpha > 0$ and $\beta \in (0, 1)$ are two fixed step-sizes.

The main idea of Algorithm 1 is very natural. Similar to other consensus-based algorithms, e.g., [19], [20], [23], the distributed averaging in (4a) and (4d) drives both $\hat{x}_i$ and $v_i$ to the aggregate action $\hat{x}$, while $\hat{v}_i$ is a more accurate estimate due to one more mixing step in (4c). Then, a projected-gradient step is conducted in (4b) to compute a feasible direction $\hat{x}_i^{k+1} - x_i^k$, and the action is updated along the direction with a step-size $\beta$ in (4c). Note that $x_i^{k+1}$ is a convex combination of $\hat{x}_i$ and $\hat{x}_i$, thus also belongs to $X_i$.

Our key idea for designing Algorithm 1 lies in (4c) where $\beta \in (0, 1)$. Note that if $\beta = 1$ and $\alpha$ is replaced by a sequence of diminishing step-sizes; then Algorithm 1 is reduced to the algorithm in [14]. When $\beta < 1$, each player updates its action by a smaller step along the feasible direction. As shown in Proposition 2 when the second largest singular value $\sigma$ of the consensus matrix $W$ is too large, we necessarily need $\beta < 1$ to obtain a linear convergence. Moreover, we can further adjust $\beta$ to reach a faster convergence rate in Section IV.

Remark 1. Consensus-based projected-gradient method is also exploited for solving distributed constrained optimization problems where multiple computational nodes cooperate to optimize a common cost function, see [20], [25], [26] for details. However, the algorithm cannot be directly applied to NAG since each player needs to optimize an individual cost function.

B. Convergence Analysis

Firstly, we provide two lemmas that are important for the later analysis. Then, we prove the convergence of Algorithm 1 in (4) via constructing a linear system inequality. Finally, we derive the feasible region of the step-sizes $\alpha$ and $\beta$, and explicitly show the convergence rate.

Lemma 4. $\sum_{i=1}^n v_i^k = \sum_{i=1}^n x_i^k$ holds for all $k$.

Proof. Taking the summation of (4d) over $i$, we obtain that

$$\sum_{i=1}^n v_i^{k+1} = \sum_{i=1}^n v_i^k - \sum_{i=1}^n x_i^k + \sum_{i=1}^n x_i^{k+1} = \sum_{i=1}^n x_i^{k+1} + \sum_{i=1}^n \hat{x}_i^{k+1} = \sum_{i=1}^n x_i^k.$$

Since each player assigns $v_i^0 = x_i^0$, the result is arrived at by induction. \hfill \Box

Lemma 5. Under Assumption 3 the pseudo-gradient $\phi$ is Lipschitz continuous over $\mathcal{X}$ with coefficient $\tilde{L} = \sqrt{2L}$.

Proof. From the definition of the pseudo-gradient in Section II-B we have that for any $x, x' \in \mathcal{X}$,

$$||\phi(x) - \phi(x')||^2 = \sum_{i=1}^n ||F_i(x_i, x) - F_i(x'_i, x')||^2 \\
\leq L^2 \sum_{i=1}^n (||x_i - x_i'||^2 + ||x - x'||^2) \\
\leq 2L^2 ||x - x'||^2.$$

Thus, $\phi$ is Lipschitz continuous with coefficient $\sqrt{2L}$. \hfill \Box

In the following analysis, we denote

$$X^k = [x_1^k, x_2^k, \ldots, x_n^k]^T, \quad \hat{X}^k = [\hat{x}_1^k, \hat{x}_2^k, \ldots, \hat{x}_n^k]^T, \quad X_i^k = [x_i^k, \hat{x}_i^k, \ldots, \hat{x}_n^k]^T, \quad V_i^k = [v_i^k, v_i^k, \ldots, v_i^k]^T, \quad \text{and} \quad \hat{X}_i^k = 1(\hat{x}_i^k)^T.$$

Then, (4c) and (4d) can be compactly written as

$$X^{k+1} = \beta \hat{X}^{k+1} + (1 - \beta) X^k, \quad V_i^{k+1} = W V_i^k + X_i^{k+1} - X_i^k.$$

Proposition 1. Under Assumptions 1, 2 when $\alpha$ and $\beta$ are small enough such that the matrix

$$M = \begin{bmatrix}
1 - \alpha \beta (\mu - 2a \hat{L}^2) & \alpha \beta L^2 (1/\mu + 2a) \\
8 \sigma^2 - 4a \beta^2 (\mu - 2a \hat{L}^2) & 4a \beta^2 L^2 (1/\mu + 2a) + 2 \sigma^2 + 2 \gamma \sigma
\end{bmatrix}_{\frac{1}{1-\sigma^2}}$$

then $M$ is irreducible and aperiodic, and $\lambda_{\max}(M) < 1$.
has spectral radius $\rho(M) < 1$. Then, $\|X^k - X^*\|_F^2$ and $\|V^k - \overline{X}_k\|_F^2$ are all decaying with rate $O(\rho(M)^k)$.

**Proof.** We first bound $\|X^{k+1} - X^*\|_F^2$ and $\|V^{k+1} - \overline{X}^{k+1}\|_F^2$ by linear combinations of their values in the last iteration. Then, a linear system inequality is derived which implies the convergence rate.

**Step 1: Bound** $\|X^{k+1} - X^*\|_F^2$. It follows from (4b) that

$$\|X^{k+1} - X^*\|_F^2 = \|\Pi_{X_*}[x^k - \alpha \hat{F}_k] - \Pi_{X_*}[x^* - \alpha F^*_k]\|^2 \leq \|x^k - x^*_k\|^2 + \alpha^2 \|F_k - F^*_k\|^2 - 2\alpha (x^k - x^*_k)^T(\hat{F}_k - F^*_k),$$

where we have used $\hat{F}_k$ and $F^*_k$ to respectively denote $F_k(x^k, \hat{x}^k)$ and $F_k(x^k, \hat{x}^k)$ for simplicity. The inequality follows from Lemma 3. Denote $F_k(x^k, \hat{x}^k)$ by $F_k^*$, then

$$\|\hat{F}_k - F^*_k\|^2 = \|\hat{F}_k - F_k\|^2 + \|F_k - F^*_k\|^2 + 2(\hat{F}_k - F_k)^T(F_k - F^*_k) \leq L^2\|x^k - x^*_k\|^2 + \|F_k - F^*_k\|^2 + 2(\hat{F}_k - F_k)^T(F_k - F^*_k) \leq \|\hat{F}_k - F^*_k\|^2$$

and

$$(x^k - x^*_k)^T(\hat{F}_k - F^*_k) = (x^k - x^*_k)^T(\hat{F}_k - F_k) + (x^k - x^*_k)^T(F_k - F^*_k).$$

Plugging (6) into (5) and taking the summation over $i$, we have

$$\|\hat{X}^{k+1} - X^*\|_F^2 \leq \|X^k - X^*\|_F^2 + \alpha^2 L^2 ||WV^k - \overline{X}_k||_2^2 + \alpha^2 \|F_k - F^*_k\|^2 + 2\alpha (X^k - X^*, F_k - F^*)_F + 2\alpha (X^k - X^*, F_k - F^*)_F - 2\alpha (X^k - X^*, \hat{F}_k - F^*_k) \leq \|X^k - X^*\|_F^2 + \alpha^2 L^2 ||WV^k - \overline{X}_k||_2^2 + \alpha^2 \|F_k - F^*_k\|^2 + 2\alpha L^2 ||X^k - X^*\|_F^2 + \alpha^2 \|X^k - X^*\|_F^2 + 2\alpha L^2 ||WV^k - \overline{X}_k||_2^2 + \alpha \mu ||X^k - X^*\|_F^2 + 2\alpha L^2 ||WV^k - \overline{X}_k||_2^2,$$

where the second inequality due to Cauchy-Schwarz inequality and mean inequality. Namely,

$$\|\hat{X}^{k+1} - X^*\|_F^2 \leq (1 - \alpha \mu + \alpha^2 L^2)\|X^k - X^*\|_F^2 + (\alpha \mu + \alpha^2 L^2)\|WV^k - \overline{X}_k\|_F^2.$$

It follows from (4c) that

$$\|X^{k+1} - X^*\|_F^2 = \beta^2 \|\hat{X}^{k+1} - X^*\|_F^2 + (1 - \beta)^2 \|X^k - X^*\|_F^2 + 2\beta(1 - \beta)(\hat{X}^{k+1} - X^*)^T(X^k - X^*) \leq \beta \|\hat{X}^{k+1} - X^*\|_F^2 + (1 - \beta)^2 \|X^k - X^*\|_F^2.$$
where $a$, $b$, $c$ and $d$ are defined in (11) and (14). Since $M$ and $\zeta^k$ have non-negative entries, we can expand (16) recursively and obtain

$$\zeta^k \leq M^k \zeta^0.$$  

Thus, if $\alpha$ and $\beta$ is such that $\rho(M) < 1$; then $\|X^k - X^*\|_F^2$ and $\|V^k - X^k\|_F^2$ are all decaying with rate $O(\rho(M)^k)$. $\square$

Next we explicitly show the convergence rate of Algorithm 1

**Proposition 2.** When

$$0 < \beta \leq \min \left\{ 1, \frac{\mu(1 - \sigma^2)}{2\sigma L \sqrt{1 + 11\sigma^2}} \right\},$$

$$0 < \alpha \leq \frac{\mu^2 (1 - \sigma^2)^2 - 4(7 + 11\sigma^2)(\sigma L \beta)^2}{2\mu L^2 (1 - \sigma^2)^2 + 8\mu(7 + 11\sigma^2)(\sigma L \beta)^2},$$

we have

$$\rho(M) < 1 - \frac{\alpha \beta}{2} \left[ \frac{\mu - 2\alpha L^2 - 4(\sigma L \beta)^2}{1 - \sigma^2} \left( \frac{1}{\mu} + 2\alpha \right) \right] < 1.$$  

**Proof.** Since $0 < \alpha < \frac{\mu}{2\sigma^2}$, $0 < \alpha < 1$, the characteristic polynomial $p(\lambda)$ of $M$ is

$$p(\lambda) = \lambda^2 + (a - bc - d - 1)\lambda + [(1 - a)d - bc],$$

where $a$, $b$, $c$ and $d$ are defined in (11) and (14). The two roots of $p(\lambda)$ are $\frac{-B \pm \sqrt{B^2 - 4\Delta}}{2}$, where

$$B = a - bc - d - 1 < 0,$$

$$\Delta = B^2 - 4[(1 - a)d - bc] > 0.$$  

Thus, the spectral radius of $M$ is $\rho(M) = \frac{-B + \sqrt{B^2 - 4\Delta}}{2}$. Since

$$\beta \leq \frac{1}{\sqrt{\sigma(1 + \sigma^2)}};$$

we have $\alpha < 1$. Besides, in view of the upper bound of $\alpha$ and $\beta$ in (17), we have $a(1 - d) > (7 + 2d)bc$ and $a > bc$. Then, we can obtain that $

\Delta < (1 - d)^2. Thus,

$$\rho(M) < 1 - \frac{a - bc}{2} < 1,$$

which is the desired result. $\square$

**Remark 2.** When $\sigma$ of the consensus matrix $W$ is smaller, we can obtain a faster convergence rate. Besides, if $\sigma$ is small enough such that we can choose $\beta = 1$, then Algorithm 1 can be simplified. Otherwise, we must set $\beta < 1$ to theoretically guarantee the linear convergence.

## IV. NUMERICAL EXPERIMENTS

We consider the Nash-Cournot games in [14]. Consider $n = 20$ firms competing over $L = 10$ locations, each firm $i$ needs to decide its production $g_{il}$ and sales $s_{il}$ at location $l$. The cost of production of firm $i$ at location $l$ is $c_{il}(g_{il})$ and is defined as

$$c_{il}(g_{il}) = a_{il} g_{il} + b_{il} g_{il}^2,$$

where $a_{il}$ and $b_{il}$ are parameters for firm $i$. Firm $i$’s revenue at location $l$ is $s_{il} p_l(s_{il})$, where $s_{il} = \sum_{j=1}^n s_{il}$ is the total sales at location $l$ and $p_l$ is the corresponding price function.

The price function $p_l$ captures the reverse demand function and is defined as

$$p_l(s_l) = d_l - s_l,$$

where $d_l$ is a parameter for location $l$. The production capacity of firm $i$ at location $l$ is denoted by $\text{cap}_{il}$. Assuming that the transportation fees between any two locations are zero; then, firm $i$’s optimization problem is:

$$\text{minimize} \quad \sum_{l=1}^L [c_{il}(g_{il}) - s_{il} p_l(s_{il})],$$

subject to \(g_{il}, s_{il} \geq 0, g_{il} \leq \text{cap}_{il}, \sum_{l=1}^L g_{il} = \sum_{l=1}^L s_{il} \).

It is easy to verify that the problem and the corresponding pseudo-gradient mapping satisfy Assumptions [13]. In the simulations, we set $a_{il} \sim U(2, 12)$, $b_{il} \sim U(2, 3)$, $d_l \sim U(90, 100)$ and $\text{cap}_{il} = 500$ for $i = 1, \cdots, n$ and $l = 1, \cdots, L$, where $U(u_1, u_2)$ is the uniform distribution over the interval $[u_1, u_2]$.

We first validate Algorithm 1 over three network topologies:

- **Linear:** player $i$ communicates with players mod$(i + 11j, n) + 1, 0 \leq j < n/11$.
- **Log:** player $i$ communicates with players mod$(i + 2^j, n) + 1, 0 \leq j < \log_2 n$.
- **Complete:** every player communicates with every other players.

The weight matrix $W = \{w_{ij}\}$ is set as

$$w_{ij} = \begin{cases} 0, & \text{if } (i, j) \notin \mathcal{E}, \\ \delta, & \text{if } (i, j) \in \mathcal{E} \text{ and } i \neq j, \\ 1 - \delta \cdot d(i), & \text{if } i = j, \end{cases}$$

where $d(i)$ is the number of players communicating with player $i$, and $\delta = 0.5 / \max_i \{d(i)\}$.

Fig. 1 depicts the trajectories of suboptimality gap $\|x^k - x^*\|$ versus iterations over the three network topologies. All the trajectories start with the same initial point, and the step-sizes are tuned to be optimal. It can be seen that Algorithm 1 converges linearly over different topologies. Moreover, when the network has stronger connectivity, i.e., a smaller $\sigma$ that defined in Lemma 2, Algorithm 1 achieves a faster convergence rate.

Then, over the **linear** topology, we fix different values of $\beta$ and search the corresponding optimal value of $\alpha$. 

![Fig. 1. The trajectories of suboptimality gap vs. iterations over different network topologies.](image-url)
TABLE I

| $\beta$  | 0.8  | 0.85 | 0.9  | 0.95 | 1.0  |
|----------|------|------|------|------|------|
| $\alpha$ | 0.108| 0.092| 0.096| 0.090| 0.087|
| $\epsilon_{200}$ ($\times 10^{-6}$) | 1.615| 1.622| 1.418| 1.512| 1.591|

Fig. 2. The trajectories of suboptimality gap vs. iterations over different network topologies.

Then, we run Algorithm [12] for 200 iterations and obtain the suboptimality gap $\epsilon_{200} = |x^{200} - x^\ast|$. As shown in Table I, a smaller $\beta$ usually leads to a bigger $\alpha$, which coincides with Proposition II. It cannot be theoretically proved whether Algorithm [12] converges linearly for $\beta = 1$. However, our experiment shows that when $\beta$ is slightly less than 1, we can reach a faster convergence rate, i.e., obtain a smaller $\epsilon$.

Finally, still over the linear topology, we compare Algorithm [12] with algorithms in [12] and [14], of which the former is designed for games where cost functions depend on actions in a general manner, and the latter is designed for NAGs but only with diminishing step-sizes. Fig. 2 shows that Algorithm [12] outperforms the others. Algorithm in [12] gets stuck in the beginning phase, which is due to each players’ mixture of neighbors faulty estimates of their own action. Although it converges linearly in the long run, the convergence rate is slower than Algorithm [12]. As for algorithm in [14], since it uses diminishing step-sizes, the convergence rate is sublinear and also inferior to Algorithm [12].

V. CONCLUSIONS AND FUTURE WORKS

In this paper, we have proposed a fully distributed algorithm for the NAGs over an undirected communication network. When the NAG possesses a strongly monotone and Lipschitz continuous pseudo-gradient mapping, the proposed algorithm with fixed step-sizes is proved to converge linearly to the unique NE. Theoretical results have been validated by numerical experiments.

Future work will consider linearly convergent algorithms for NAGs over directed communication networks or time-varying networks. Fully asynchronous extension of the proposed algorithm is also an open and difficult problem.

REFERENCES

[1] M. Jensen, “Aggregative games,” Discussion Papers, Department of Economics, University of Birmingham, 2006.
[2] M. Jensen, “Aggregative games and best-reply potentials,” Economic theory, vol. 43, no. 1, pp. 45–66, 2010.
[3] T. Alpcan and T. Başar, “A game-theoretic framework for congestion control in general topology networks,” in Proceedings of the 41st IEEE Conference on Decision and Control, 2002., vol. 2, pp. 1218–1224, IEEE, 2002.
[4] J. Barrera and A. García, “Dynamic incentives for congestion control,” IEEE Transactions on Automatic Control, vol. 60, no. 2, pp. 299–310, 2014.
[5] B. Gharesifard, T. Başar, and A. D. Domínguez-García, “Price-based coordinated aggregation of networked distributed energy resources,” IEEE Transactions on Automatic Control, vol. 61, no. 10, pp. 2936–2946, 2016.
[6] Z. Ma, D. S. Callaway, and I. A. Hiskens, “Decentralized charging control of large populations of plug-in electric vehicles,” IEEE Transactions on Control Systems Technology, vol. 21, no. 1, pp. 67–78, 2013.
[7] N. Li, L. Chen, and M. A. Dahleh, “Demand response using linear supply function bidding,” IEEE Transactions on Smart Grid, vol. 6, no. 4, pp. 1827–1838, 2015.
[8] F. Salehisadaghiani, W. Shi, and L. Pavel, “Distributed Nash equilibrium seeking under partial-decision information via the alternating direction method of multipliers,” Automatica, vol. 103, pp. 27–35, 2019.
[9] M. Bianchi, G. Belgioioso, and S. Grammatico, “A fully-distributed proximal-point algorithm for Nash equilibrium seeking with linear convergence rate,” in 2020 59th IEEE Conference on Decision and Control (CDC), pp. 2303–2308, 2020.
[10] T. Tatarenko and A. Nedić, “Geometric convergence of distributed gradient play in games with unconstrained action sets,” arXiv preprint arXiv:1907.07144, 2019.
[11] T. Tatarenko, W. Shi, and A. Nedić, “Geometric convergence of gradient play algorithms for distributed Nash equilibrium seeking,” IEEE Transactions on Automatic Control, pp. 1–1, 2020.
[12] M. Bianchi and S. Grammatico, “Fully distributed Nash equilibrium seeking over time-varying communication networks with linear convergence rate,” IEEE Control Systems Letters, vol. 5, no. 2, pp. 499–504, 2021.
[13] F. Facchinei and J.-S. Pang, Finite-Dimensional Variational Inequalities and Complementarity Problems. Springer Science & Business Media, 2007.
[14] J. Koshal, A. Nedić, and U. V. Shanbhag, “Distributed algorithms for aggregative games on graphs,” Operations Research, vol. 64, no. 3, pp. 680–704, 2016.
[15] G. Belgioioso, A. Nedić, and S. Grammatico, “Distributed generalized Nash equilibrium seeking in aggregative games,” IEEE Transactions on Patients, 1998.
[16] T. Basar and G. J. Olsder, Dynamic Noncooperative Game Theory. Cambridge University Press, 1999.
[17] T. Basar and G. J. Olsder, Dynamic Noncooperative Game Theory. Cambridge University Press, 1999.
[18] T. Basar and G. J. Olsder, Dynamic Noncooperative Game Theory. Cambridge University Press, 1999.
[19] T. Basar and G. J. Olsder, Dynamic Noncooperative Game Theory. Cambridge University Press, 1999.
[20] T. Basar and G. J. Olsder, Dynamic Noncooperative Game Theory. Cambridge University Press, 1999.