Fully distributed Nash equilibrium seeking over time-varying communication networks with linear convergence rate

Bianchi, M.; Grammatico, S.

DOI
10.1109/LCSYS.2020.3002734

Publication date
2021

Document Version
Final published version

Published in
IEEE Control Systems Letters

Fully Distributed Nash Equilibrium Seeking Over Time-Varying Communication Networks With Linear Convergence Rate

Mattia Bianchi and Sergio Grammatico, *Senior Member, IEEE*

***Abstract*—We design a distributed algorithm for learning Nash equilibria over time-varying communication networks in a partial-decision information scenario, where each agent can access its own cost function and local feasible set, but can only observe the actions of some neighbors. Our algorithm is based on projected pseudo-gradient dynamics, augmented with consensual terms. Under strong monotonicity and Lipschitz continuity of the game mapping, we provide a simple proof of linear convergence, based on a contractivity property of the iterates. Compared to similar solutions proposed in literature, we also allow for time-varying communication and derive tighter bounds on the step sizes that ensure convergence. In fact, in our numerical simulations, our algorithm outperforms the existing gradient-based methods, when the step sizes are set to their theoretical upper bounds. Finally, to relax the assumptions on the network structure, we propose a different pseudo-gradient algorithm, which is guaranteed to converge on time-varying balanced directed graphs.**

***Index Terms*—Game theory, optimization algorithms, networked control systems.**

I. Introduction

NASH equilibrium (NE) problems arise in several network systems, where multiple selfish decision-makers, or agents, aim at optimizing their individual, yet inter-dependent, objective functions. Engineering applications include communication networks [1], demand-side management in the smart grid [2], charging of electric vehicles [3] and demand response in competitive markets [4]. From a game-theoretic perspective, the challenge is to assign the agents behavioral rules that eventually ensure the attainment of a NE, a joint action from which no agent has an incentive to unilaterally deviate.

Literature review: Typically, NE seeking algorithms are designed under the assumption that each agent can access the decisions of all the competitors [5], [6], [7]. This *full-decision information* hypothesis requires the presence of a coordinator, that broadcast the data to the network, and it is impractical for some applications [8], [9]. One example is the Nash-Cournot competition model described in [10], where the profit of each of a group of firms depends not only on its own production, but also on the whole amount of sales, a quantity not directly accessible by any of the firms. Therefore, in recent years, there has been an increased attention for fully distributed algorithms that allow to compute NEs relying on local information only. In this letter, we consider the so-called *partial-decision information* scenario, where the agents engage in nonstrategic information exchange with some neighbors on a network; based on the data received, they can estimate and eventually reconstruct the actions of all the competitors. This setup has only been introduced very recently. In particular, most of the results available resort to (projected) gradient and consensus dynamics, both in continuous time [11], [12], and discrete time. For the discrete time case, fixed-step algorithms were proposed in [13], [14], [15] (the latter for generalized games), all exploiting a certain restricted monotonicity property. Alternatively, the authors of [16] developed a gradient-play scheme by leveraging contractivity properties of doubly stochastic matrices. Nevertheless, in all these approaches theoretical guarantees are provided only for step sizes that are typically very small, affecting the speed of convergence. Furthermore, all the methods cited are designed for a time-invariant, undirected network. To the best of our knowledge, switching communication topologies have only been addressed with diminishing step sizes. For instance, the early work [10] considered aggregative games over time-varying jointly connected undirected graphs. This result was extended by the authors of [17] to games with coupling constraints. In [18], an asynchronous gossip algorithm was presented to seek a NE over directed graphs. The drawback is that vanishing steps typically result in slow convergence.

Contribution: Motivated by the above, in this letter we present the first fixed-step NE seeking algorithms for strongly monotone games over time-varying communication networks. Our novel contributions are summarized as follows:

- We propose a fully distributed projected gradient-play method, that is guaranteed to converge with linear rate when the network adjacency matrix is doubly stochastic.

Manuscript received March 15, 2020; revised May 14, 2020; accepted June 1, 2020. Date of publication June 16, 2020; date of current version July 3, 2020. This work was supported in part by Netherlands Organization for Scientific Research through Project OMEGA under Grant 613.001.702, and in part by the European Research Council through Project COSMOS under Grant 802348. Recommended by Senior Editor M. Guay. *(Corresponding author: Mattia Bianchi.)*

The authors are with the Delft Center for Systems and Control, TU Delft, 2613 GA Delft, The Netherlands (e-mail: m.bianchi@tudelft.nl; s.grammatico@tudelft.nl).

Digital Object Identifier 10.1109/LCSYS.2020.3002734

With respect to [16], we consider a time-varying communication network and we allow for constrained action sets. Moreover, differently from the state of the art, we provide an upper bound on the step size that does not vanish as the number of agents increases (Section III);
- We show via numerical simulations that, even in the case of fixed networks, our algorithm outperforms the existing pseudo-gradient based dynamics, when the step sizes are set to their theoretical upper bounds (Section V);
- We prove that linear convergence to a NE on time-varying weight-balanced directed graphs can be achieved via a forward-backward algorithm [19, Sec. 12.7.2], which has been studied in [14], [15], but only for the special case of fixed undirected networks (Section IV).

Basic notation: $\mathbb{N}$ is the set of natural numbers, including 0. $\mathbb{R}$ is the set of real numbers. $\mathbf{0}_n$ ($\mathbf{1}_n$) denotes the vector of dimension n with all elements equal to 0 (1); I_n the identity matrix of dimension n; the subscripts might be omitted when there is no ambiguity. For a matrix $A \in \mathbb{R}^{m\times n}$, $A^\top$ denotes its transpose; $[A]_{i,j}$ is the element on row i and column j, and the second subscript is omitted if $n = 1$; $\sigma_{\min}(A) = \sigma_1(A) \leq \cdots \leq \sigma_n(A) =: \sigma_{\max}(A) = \|A\|$ denote its singular values. If $A \in \mathbb{R}^{n\times n}$, $\det(A)$ is its determinant; $A \succ 0$ stands for symmetric positive definite matrix; if A is symmetric, $\lambda_{\min}(A) = \lambda_1(A) \leq \cdots \leq \lambda_n(A) =: \lambda_{\max}(A)$ denote its eigenvalues. $\otimes$ denotes the Kronecker product. $\mathrm{diag}(A_1, \ldots, A_N)$ denotes the block diagonal matrix with $A_1, \ldots, A_N$ on its diagonal. Given N vectors $x_1, \ldots, x_N$, $x := \mathrm{col}(x_1, \ldots, x_N) = [x_1^\top \ldots x_N^\top]^\top$ and $x_{-i} = \mathrm{col}(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_N)$. $\|\cdot\|$ denotes the Euclidean vector norm. For a differentiable function $g : \mathbb{R}^n \to \mathbb{R}$, $\nabla_x g(x)$ denotes its gradient. A mapping $\mathcal{A} : \mathbb{R}^m \to \mathbb{R}^n$ is ℓ-Lipschitz continuous if, for any $x, y \in \mathbb{R}^m$, $\|\mathcal{A}(x) - \mathcal{A}(y)\| \leq \ell\|x - y\|$. $\mathrm{proj}_S : \mathbb{R}^n \to S$ denotes the Euclidean projection onto a closed convex set S. An operator $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ is (μ-strongly) monotone if, for any $x, y \in \mathbb{R}^n$, $(\mathcal{F}(x) - \mathcal{F}(y))^\top (x - y) \geq 0 \,(\geq \mu\|x - y\|^2)$. The variational inequality $\mathrm{VI}(\mathcal{F}, S)$ is the problem of finding a vector $x^* \in S$ such that $\mathcal{F}(x^*)^\top (x - x^*) \geq 0$, for all $x \in S$.

II. Mathematical Setup

We consider a set of agents $\mathcal{I} := \{1, \ldots, N\}$, where each agent $i \in \mathcal{I}$ shall choose its action (i.e., decision variable) x_i from its local decision set $\Omega_i \subseteq \mathbb{R}^{n_i}$. Let $x = \mathrm{col}((x_i)_{i\in\mathcal{I}}) \in \Omega$ denote the stacked vector of all the agents' decisions, $\Omega = \Omega_1 \times \cdots \times \Omega_N \subseteq \mathbb{R}^n$ the overall action space and $n := \sum_{i=1}^N n_i$. The goal of each agent $i \in \mathcal{I}$ is to minimize its objective function $J_i(x_i, x_{-i})$, which depends on both the local variable x_i and the decision variables of the other agents $x_{-i} = \mathrm{col}((x_j)_{j\in\mathcal{I}\setminus\{i\}})$. The game is then represented by the inter-dependent optimization problems:

$$\forall i \in \mathcal{I}: \quad \operatorname*{argmin}_{y_i \in \Omega_i} J_i(y_i, x_{-i}). \tag{1}$$

The technical problem we consider in this letter is the computation of a NE, as defined next.

Definition 1: A Nash equilibrium is a set of strategies $x^* = \mathrm{col}((x_i^*)_{i\in\mathcal{I}}) \in \Omega$ such that, for all $i \in \mathcal{I}$:

$$J_i\big(x_i^*, x_{-i}^*\big) \leq \inf\{J_i\big(y_i, x_{-i}^*\big) | y_i \in \Omega_i\}.$$

The following regularity assumptions are common for NE problems, see, e.g., [15, Ass. 1], [14, Ass. 1].

Standing Assumption 1 (Regularity and Convexity): For each $i \in \mathcal{I}$, the set Ω_i is non-empty, closed and convex; J_i is continuous and the function $J_i(\cdot, x_{-i})$ is convex and continuously differentiable for every x_{-i}.

Under Standing Assumption 1, a joint action x^* is a NE of the game in (1) if and only if it solves the variational inequality $\mathrm{VI}(F, \Omega)$ [19, Prop. 1.4.2], or, equivalently, if and only if, for any $\alpha > 0$ [19, Prop. 1.5.8],

$$x^* = \mathrm{proj}_\Omega(x^* - \alpha F(x^*)), \tag{2}$$

where F is the *pseudo-gradient* mapping of the game:

$$F(x) := \mathrm{col}\big((\nabla_{x_i} J_i(x_i, x_{-i}))_{i\in\mathcal{I}}\big). \tag{3}$$

Next, we postulate a sufficient condition for the existence of a unique NE, namely the strong monotonicity of the pseudo-gradient [19, Th. 2.3.3]. This assumption is always used for (G)NE seeking under partial-decision information with fixed step sizes, e.g., in [14, Ass. 2], [15, Ass. 3]. It implies strong convexity of the functions $J_i(\cdot, x_{-i})$ for every x_{-i}, but not necessarily (strong) convexity of J_i in the full argument.

Standing Assumption 2: The pseudo-gradient mapping in (3) is μ-strongly monotone and ℓ_0-Lipschitz continuous, for some μ, $\ell_0 > 0$.

In our setup, each agent i can only access its own cost function J_i and feasible set Ω_i. Moreover, agent i does not have full knowledge of x_{-i}, and only relies on the information exchanged locally with neighbors over a time-varying directed communication network $\mathcal{G}_k(\mathcal{I}, \mathcal{E}_k)$. The ordered pair (i, j) belongs to the set of edges, $\mathcal{E}_k$, if and only if agent i can receive information from agent j at time k. Let $W_k \in \mathbb{R}^{N\times N}$ denote the weighted adjacency matrix of $\mathcal{G}_k$, and $w_{i,j}^k := [W_k]_{i,j}$, with $w_{i,j}^k > 0$ if $(i, j) \in \mathcal{E}_k$, $w_{i,j}^k = 0$ otherwise; $D_k = \mathrm{diag}((d_i^k)_{i\in\mathcal{I}})$ and $L_k = D_k - W_k$ the in-degree and Laplacian matrices of $\mathcal{G}^k$, with $d_i^k = \sum_{j=1}^N w_{i,j}^k$; $\mathcal{N}_i^k = \{j | (i, j) \in \mathcal{E}_k\}$ the set of in-neighbors of agent i.

Standing Assumption 3: For each $k \in \mathbb{N}$, the graph $\mathcal{G}_k$ is strongly connected.

Assumption 1: For all $k \in \mathbb{N}$, the following hold:
(i) *Self-loops:* $w_{i,i}^k > 0$ for all $i \in \mathcal{I}$;
(ii) *Double stochasticity:* $W_k \mathbf{1}_N = \mathbf{1}_N$, $\mathbf{1}_N^\top W_k = \mathbf{1}_N^\top$.

Remark 1: Assumption 1(i) is intended just to ease the notation. Instead, Assumption 1(ii) is stronger. It is typically used for networked problems on undirected symmetric graphs, e.g., in [10, Ass. 6], [17, Ass. 3], [16, Ass. 3], justified by the fact that it can be satisfied by assigning the following Metropolis weights to the communication:

$$\tilde{w}_{i,j}^k = \begin{cases} w_{i,j}^k/(\max\{d_i^k, d_j^k\} + 1) & \text{if } j \in \mathcal{N}_i\setminus\{i\}; \\ 0 & \text{if } j \notin \mathcal{N}_i; \\ 1 - \sum_{j\in\mathcal{N}_i\setminus\{i\}} \tilde{w}_{i,j}^k & \text{if } i = j. \end{cases}$$

In practice, to satisfy Assumption 1(ii) in case of symmetric communication, even under time-varying topology, it suffices for the agents to exchange their in-degree with their neighbors at every time step. Therefore, Standing Assumption 3 and Assumption 1 are easily fulfilled for undirected graphs connected at each step. For directed graphs, given any strongly connected topology, weights can be assigned such that the

resulting adjacency matrix (with self-loops) is doubly stochastic, via an iterative distributed process [20]. However, this can be impractical if the network is time-varying.

Under Assumption 1, it holds that $\sigma_{N-1}(W_k) < 1$, for all k, where $\sigma_{N-1}(W_k)$ denotes the second largest singular value of W_k. Moreover, for any $y \in \mathbb{R}^N$,

$$\|W_k(y - \mathbf{1}_N\bar{y})\| \leq \sigma_{N-1}(W_k)\|y - \mathbf{1}_N\bar{y}\|, \tag{4}$$

where $\bar{y} = \frac{1}{N}\mathbf{1}_N^\top y$ is the average of y. We will further assume that $\sigma_{N-1}(W_k)$ is bounded away from 1; this automatically holds if the networks $\mathcal{G}_k$ are chosen among a finite family.

Assumption 2: There exists $\bar{\sigma} \in (0, 1)$ such that $\sigma_{N-1}(W_k) \leq \bar{\sigma}$, for all $k \in \mathbb{N}$.

III. Distributed Nash Equilibrium Seeking

In this section, we present a pseudo-gradient algorithm to seek a NE of the game (1) in a fully distributed way. To cope with partial-decision information, each agent keeps an estimate of all other agents' actions. Let $\boldsymbol{x}_i = \mathrm{col}((\boldsymbol{x}_{i,j})_{j\in\mathcal{I}}) \in \mathbb{R}^{Nn}$, where $\boldsymbol{x}_{i,i} := x_i$ and $\boldsymbol{x}_{i,j}$ is agent i's estimate of agent j's action, for all $j \neq i$; also, $\boldsymbol{x}_{j,-i} = \mathrm{col}((\boldsymbol{x}_{j,l})_{l\in\mathcal{I}\setminus\{i\}})$. The agents aim at asymptotically reconstructing the true value of the opponents' actions, based on the data received from their neighbors. The procedure is summarized in Algorithm 1. Each agent updates its estimates according to consensus dynamics, then its action via a gradient step. We remark that each agent computes the partial gradient of its cost in its local estimates $\boldsymbol{x}_i$, not on the actual joint action x.

To write the algorithm in compact form, let $\boldsymbol{x} = \mathrm{col}((\boldsymbol{x}_i)_{i\in\mathcal{I}})$; as in [15, eqs. 13-14], let, for all $i \in \mathcal{I}$,

$$\mathcal{R}_i := \begin{bmatrix} \mathbf{0}_{n_i\times n_{<i}} & I_{n_i} & \mathbf{0}_{n_i\times n_{>i}} \end{bmatrix} \in \mathbb{R}^{n_i\times n}, \tag{5}$$

where $n_{<i} := \sum_{j=1}^{i-1} n_j$, $n_{>i} := \sum_{j=i+1}^{N} n_j$; let also $\mathcal{R} := \mathrm{diag}((\mathcal{R}_i)_{i\in\mathcal{I}}) \in \mathbb{R}^{n\times Nn}$. In simple terms, $\mathcal{R}_i$ selects the i-th n_i dimensional component from an n-dimensional vector. Thus, $\mathcal{R}_i\boldsymbol{x}_i = \boldsymbol{x}_{i,i} = x_i$, and $x = \mathcal{R}\boldsymbol{x}$. We define the *extended pseudo-gradient* mapping $\boldsymbol{F}$ as

$$\boldsymbol{F}(\boldsymbol{x}) := \mathrm{col}\big((\nabla_{x_i} J_i(x_i, \boldsymbol{x}_{i,-i}))_{i\in\mathcal{I}}\big). \tag{6}$$

Therefore, Algorithm 1 reads in compact form as:

$$\boldsymbol{x}^{k+1} = \mathrm{proj}_{\boldsymbol{\Omega}}(\boldsymbol{W}_k\boldsymbol{x}^k - \alpha\mathcal{R}^\top\boldsymbol{F}(\boldsymbol{W}_k\boldsymbol{x}^k)), \tag{7}$$

where $\boldsymbol{\Omega} := \{\boldsymbol{x} \in \mathbb{R}^{Nn} | \mathcal{R}\boldsymbol{x} \in \Omega\}$ and $\boldsymbol{W}_k := W_k \otimes I_n$.

Lemma 1 [21, Lemma 3]: The mapping $\boldsymbol{F}$ in (6) is ℓ-Lipschitz continuous, for some $\mu \leq \ell \leq \ell_0$.

Theorem 1: Let Assumptions 1-2 hold and let

$$M_\alpha = \begin{bmatrix} 1 - \frac{2\alpha\mu}{N} + \frac{\alpha^2\ell_0^2}{N} & \left(\frac{\alpha(\ell+\ell_0)+\alpha^2\ell_0\ell_0}{\sqrt{N}}\right)\bar{\sigma} \\ \left(\frac{\alpha(\ell+\ell_0)+\alpha^2\ell_0\ell}{\sqrt{N}}\right)\bar{\sigma} & \left(1 + 2\alpha\ell + \alpha^2\ell^2\right)\bar{\sigma}^2 \end{bmatrix}. \tag{8}$$

If the step size $\alpha > 0$ is chosen such that

$$\rho_\alpha := \lambda_{\max}(M_\alpha) = \|M_\alpha\| < 1, \tag{9}$$

then, for any initial condition, the sequence $(\boldsymbol{x}^k)_{k\in\mathbb{N}}$ generated by Algorithm 1 converges to $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, where x^* is the NE of the game in (1), with linear rate: for all $k \in \mathbb{N}$,

$$\|\boldsymbol{x}^k - \boldsymbol{x}^*\| \leq \left(\sqrt{\rho_\alpha}\right)^k \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|.$$

Algorithm 1 Fully Distributed NE Seeking

Initialization: for all $i \in \mathcal{I}$, set $x_i^0 \in \Omega_i$, $\boldsymbol{x}_{i,-i}^0 \in \mathbb{R}^{n-n_i}$.
Iterate until convergence: for all $i \in \mathcal{I}$,

Distributed averaging: $\hat{\boldsymbol{x}}_i^k = \sum_{j=1}^N w_{i,j}^k \boldsymbol{x}_j^k$

Local variables update: $x_i^{k+1} = \mathrm{proj}_{\Omega_i}(\hat{\boldsymbol{x}}_{i,i}^k - \alpha\nabla_{x_i} J_i(\hat{\boldsymbol{x}}_i^k))$

$\boldsymbol{x}_{i,-i}^{k+1} = \hat{\boldsymbol{x}}_{i,-i}^k$.

Lemma 2: The condition in (9) holds if $\alpha > 0$ and

$$\alpha < \tfrac{\bar{\sigma}}{3\ell_0} \tag{10a}$$

$$\alpha < \tfrac{2\mu}{\ell_0^2} \tag{10b}$$

$$0 < 2\mu(1-\bar{\sigma}^2) - \alpha(\bar{\sigma}^2(2\ell_0\ell + \ell^2 + 4\mu\ell + 2\ell_0^2) - \ell_0^2) - \alpha^2(\ell_0\ell^2 + \mu\ell^2 + 2\ell_0^2\ell)2\bar{\sigma}^2 - \alpha^3 2\ell_0^2\ell^2\bar{\sigma}^2. \tag{10c}$$

Proof: The condition in (10a) implies that $M_\alpha \succ 0$ (by diagonal dominance and positivity of the diagonal elements, as can be checked by recalling that $\ell \leq \ell_0$, $\mu \leq \ell_0$, $N \geq 2$, $\bar{\sigma} < 1$). The inequalities in (10b)-(10c) are the Sylvester's criterion for the matrix $I_2 - M_\alpha$: they impose that $[I_2 - M_\alpha]_{1,1} > 0$ (10b) and $\det(I_2 - M_\alpha) > 0$ (10c), hence $I_2 - M_\alpha \succ 0$. Altogether, this implies $\|M_\alpha\| < 1$. ∎

Remark 2: The conditions in (10) always hold for α small enough, since, in the monomial inequality (10c), the constant term is $2\mu(1-\bar{\sigma}^2) > 0$. While explicit solutions are known for cubic equations, we prefer the compact representation in (10c). The bounds in (10) are not tight, and in practice better bounds on the step size α are obtained by simply checking the Euclidean norm of the 2×2 matrix M_α in (8). Instead, the key observation is that the conditions in (10) do not depend on the number of agents: given the parameters $\bar{\sigma}$, μ, ℓ_0 and ℓ, a constant α that ensures convergence can be chosen independently of N. On the contrary, the rate $\sqrt{\rho_\alpha}$ does depend on N and, in fact, it approaches 1 as N grows unbounded (analogously to the results in [13], [14], [16]).

Remark 3: Compared to [16, Algorithm 7] (or [14, Algorithm 1]), in Algorithm 1 the agents *first* exchange information with their neighbors, and *then* evaluate their gradient term, resulting in better bounds on the step size α. Moreover, differently from [16, Th. 1], Theorem 1 provides a contractivity property for the iterates in (7) that holds at *each* step. This has beneficial consequences in terms of robustness, see Remark 6.

A. Technical Discussion

In Algorithm 1, the partial gradients $\nabla_{x_i} J_i$ are evaluated on the local estimates $\boldsymbol{x}_{i,-i}$, not on the actual strategies x_{-i}. Only if the estimates of all the agents coincide with the actual value, i.e., $\boldsymbol{x} = \mathbf{1}_N \otimes x$, we have that $\boldsymbol{F}(\boldsymbol{x}) = F(x)$. As a consequence, the mapping $\mathcal{R}^\top\boldsymbol{F}$ is not necessarily monotone, not even under strong monotonicity of the game mapping. Indeed, the loss of monotonicity is the main technical difficulty arising from the partial-decision information setup. Some works [12], [13], [14], [15], [21] deal with this issue by leveraging a restricted strong monotonicity property, which can be ensured, by opportunely choosing the parameter γ, for the augmented mapping $\boldsymbol{F}_a(\boldsymbol{x}) := \gamma\mathcal{R}^\top\boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{L}\boldsymbol{x}$, where $\boldsymbol{L} = L \otimes I_n$ and L is the

Laplacian of a fixed undirected connected network. Since the unique solution of the VI($\boldsymbol{F}_{\rm a}, \boldsymbol{\Omega}$) is $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, with x^* the unique NE of the game in (1) [14, Prop. 1], one can design NE seeking algorithms via standard solution methods for variational inequalities (or the corresponding monotone inclusions, [15]). For instance, in [14], a forward-backward algorithm [19, 12.4.2] is proposed to solve VI($\boldsymbol{F}_{\rm a}, \boldsymbol{\Omega}$), resulting in the algorithm

$$\boldsymbol{x}^{k+1} = \operatorname{proj}_{\boldsymbol{\Omega}}\Big(\boldsymbol{x}^k - \tau(\boldsymbol{F}_{\rm a}(\boldsymbol{x}))\Big). \tag{11}$$

We also recover this iteration when considering [15, Algorithm 1] in the absence of coupling constraints. However, exploiting the monotonicity of $\boldsymbol{F}_{\rm a}$ results in conservative upper bounds on the parameters τ and γ, and hence in slow convergence (see Sections IV-V). More recently, the authors of [16] studied the convergence of (11) based on contractivity of the iterates, in the case of a fixed undirected network with doubly stochastic adjacency matrix W, unconstrained action sets (i.e., $\Omega = \mathbb{R}^n$), and by fixing $\tau = 1$, which results in the algorithm:

$$\boldsymbol{x}^{k+1} = (W \otimes I_N)\boldsymbol{x} - \alpha\mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}^k). \tag{12}$$

Nonetheless, the upper bound on α provided in [16, Th. 1] is decreasing to zero when the number of agents N grows unbounded (in contrast with that in Theorem 1, see Lemma 2).

IV. Balanced Directed Graphs

In this section, we relax Assumption 1 to the following.

Assumption 3: For all $k \in \mathbb{N}$, the communication graph $\mathcal{G}_k$ is weight balanced: $(\mathbf{1}_N^\top W_k)^\top = W_k \mathbf{1}_N$.

For weight-balanced digraphs, in-degree and out-degree of each node coincide. Therefore, the matrix $\tilde{L}_k := (L_k + L_k^\top)/2 = D_k - (W_k + W_k^\top)/2$ is itself the symmetric Laplacian of an undirected graph. Besides, such a graph is connected by Standing Assumption 3; hence $\tilde{L}_k$ has a simple eigenvalue in 0, and the others are positive, i.e., $\lambda_2(\tilde{L}_k) > 0$.

Assumption 4: There exist $\bar{\sigma}, \bar{\lambda} > 0$ such that $\sigma_{\max}(L_k) \le \bar{\sigma}$ and $\lambda_2(\tilde{L}_k) \ge \bar{\lambda}$, for all $k \in \mathbb{N}$.

Remark 4: Assumptions 2 and 4 always hold if the networks switch among a finite family. Yet, $\bar{\sigma}$, $\tilde{\sigma}$ and $\bar{\lambda}$ are global parameters, that could be difficult to compute in a distributed way; upper/lower bounds might be available for special classes of networks, e.g., unweighted graphs.

To seek a NE over switching balanced digraphs, we propose the iteration in Algorithm 2. In compact form, it reads as

$$\boldsymbol{x}^{k+1} = \operatorname{proj}_{\boldsymbol{\Omega}}\Big(\boldsymbol{x}^k - \tau(\gamma\mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}^k) + \boldsymbol{L}_k \boldsymbol{x}^k)\Big) \tag{13}$$

where $\boldsymbol{L}_k = L_k \otimes I_n$. Clearly, (13) is the same scheme of (11), just adapted to take the switching topology into account. In fact, the proof of convergence of Algorithm 2 is based on a restricted strong monotonicity property of the operator

$$\boldsymbol{F}_{\rm a}^k(\boldsymbol{x}) := \gamma\mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{L}_k \boldsymbol{x}, \tag{14}$$

that still holds for balanced directed graphs, as we show next.

Theorem 2: Let Assumptions 3-4 hold, and let

$$M := \gamma \begin{bmatrix} \frac{\mu}{N} & -\frac{\ell_0+\ell}{2\sqrt{N}} \\ -\frac{\ell_0+\ell}{2\sqrt{N}} & \frac{\bar{\lambda}}{\gamma} - \theta \end{bmatrix}, \quad \gamma_{\max} := \frac{4\mu\bar{\lambda}}{(\ell_0+\ell)^2 + 4\mu\theta},$$
$$\bar{\mu} := \lambda_{\min}(M), \qquad \bar{\ell} := \ell + \tilde{\sigma}$$
$$\tau_{\max} := 2\bar{\mu}/\bar{\ell}^2, \tag{15}$$

Algorithm 2 Fully Distributed NE Seeking

Initialization: for all $i \in \mathcal{I}$, set $x_i^0 \in \Omega_i$, $\boldsymbol{x}_{i,-i}^0 \in \mathbb{R}^{n-n_i}$.
Iterate until convergence: for all $i \in \mathcal{I}$,

$$\hat{\boldsymbol{x}}_i^k = \textstyle\sum_{j=1}^N w_{i,j}^k (\boldsymbol{x}_i^k - \boldsymbol{x}_j^k)$$
$$x_i^{k+1} = \operatorname{proj}_{\Omega_i}\big(x_i^k - \tau(\gamma \nabla_{x_i} J_i(\boldsymbol{x}_i^k) + \hat{\boldsymbol{x}}_{i,i}^k)\big)$$
$$\boldsymbol{x}_{i,-i}^{k+1} = \hat{\boldsymbol{x}}_{i,-i}^k.$$

$$\rho_{\gamma,\tau} := 1 - 2\tau\bar{\mu} + \tau^2\bar{\ell}^2.$$

If $\gamma \in (0, \gamma_{\max})$, then $M \succ 0$ and, for any $\tau \in (0, \tau_{\max})$, for any initial condition, the sequence $(\boldsymbol{x}^k)_{k\in\mathbb{N}}$ generated by Algorithm 2 converges to $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, where x^* is the unique NE of the game in (1), with linear rate: for all $k \in \mathbb{N}$,

$$\|\boldsymbol{x}^k - \boldsymbol{x}^*\| \le \big(\sqrt{\rho_{\gamma,\tau}}\big)^k \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|.$$

Remark 5: Differently from the bound $\alpha_{\max}$ in (8), $\tau_{\max}$ in (15) vanishes as N grows (fixed the other parameters), as $\bar{\mu}$ decreases to 0 (by continuity of the eigenvalues).

Remark 6: Based on Theorems 1, 2, it can be proven that the discrete-time systems (7), (13) are input-to-state-stable (ISS) with respect to additive disturbances, with ISS-Lyapunov function $\|\boldsymbol{x} - \boldsymbol{x}^*\|^2$. By Lipschitz continuity of the updates, this implies ISS for noise both on the communication and in the evaluation of the partial gradients.

V. Numerical Example: A Nash-Cournot Game

We consider the Nash-Cournot game in [15, Sec. 6]. N firms produce a commodity that is sold to m markets. Each firm $i \in \mathcal{I} = \{1, \dots, N\}$ can only participate in $n_i \le m$ of the markets; its action $x_i \in \mathbb{R}^{n_i}$ is the vector of quantities of product to be sent to these n_i markets, bounded by the local constraints $\mathbf{0}_{n_i} \le x_i \le X_i$. Let $A_i \in \mathbb{R}^{m\times n_i}$ be the matrix that specifies which markets firm i participates in. Specifically, $[A_i]_{k,j} = 1$ if $[x_i]_j$ is the amount of product sent to the k-th market by agent i, $[A_i]_{k,j} = 0$ otherwise, for all $k = 1, \dots, m$, $j = 1, \dots, n_i$. Let $A := [A_1 \dots A_N]$; then $Ax = \sum_{i=1}^N A_i x_i \in \mathbb{R}^m$ are the quantities of total product delivered to each market. Firm i aims at maximizing its profit, i.e., minimizing the cost function $J_i(x_i, x_{-i}) = c_i(x_i) - p(Ax)^\top A_i x_i$. Here, $c_i(x_i) = x_i^\top Q_i x_i + q_i^\top x_i$ is firm i's production cost, with $Q_i \in \mathbb{R}^{n_i\times n_i}$, $Q_i \succ 0$, $q_i \in \mathbb{R}^{n_i}$. Instead, $p{:}\mathbb{R}^m \to \mathbb{R}^m$ associates to each market a price that depends on the amount of product delivered to that market. Specifically, the price for the market k, for $k = 1, \dots, m$, is $[p(Ax)]_k = \bar{P}_k - \chi_k [Ax]_k$, where $\bar{P}_k$, $\chi_k > 0$. We set $N = 20$, $m = 7$. The market structure is as in [15, Fig. 1], that defines which firms are allowed to participate in which markets. Therefore, $x \in \mathbb{R}^n$, with $n = 32$. We select randomly with uniform distribution r_k in $[1, 2]$, Q_i diagonal with diagonal elements in $[14, 16]$, q_i in $[1, 2]$, $\bar{P}_k$ in $[10, 20]$, χ_k in $[1, 3]$, X_i in $[5, 10]$, for all $i \in \mathcal{I}$, $k = 1, \dots, m$. The resulting setup satisfies Standing Assumptions 1-2 [15, Sec. 6]. The firms cannot access the production of all the competitors, but can communicate with some neighbors on a network.

We first consider the case of a fixed, undirected graph, under Assumption 1. Algorithm 2 in this case reduces to [14, Algorithm 1]. We compare Algorithms 1-2 with the

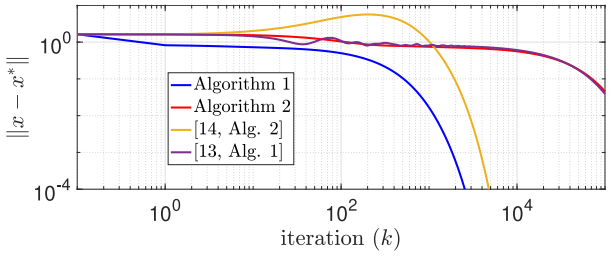

Fig. 1. Distance from the NE for different pseudo-gradient NE seeking methods, with step sizes that guarantee convergence.

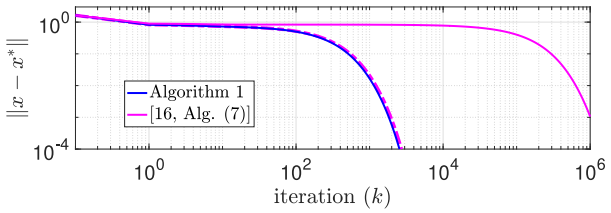

Fig. 2. Distance from the NE for Algorithm 1, with step size $\alpha = 2{*}10^{-3}$ (upper bound in Theorem 1), and the method in [16, Algorithm 1], with step size $\alpha = 4 * 10^{-6}$ (upper bound in [16, Th. 1]). Algorithm 1 converges much faster, thanks to the larger step size. The scheme in [16, Algorithm 1] still converges if we set $\alpha = 2 * 10^{-3}$ (dashed line, not supported theoretically).

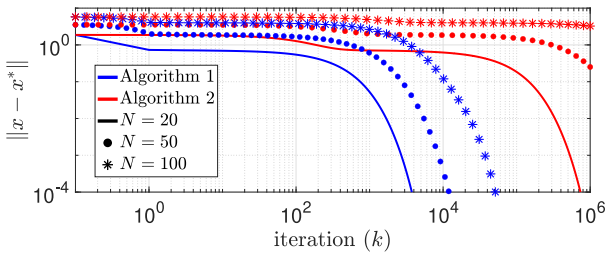

Fig. 3. Comparison of Algorithms 1 and 2, on a time-varying graph, for 20, 50 or 100 agents, with the step sizes set to their theoretical upper bounds.

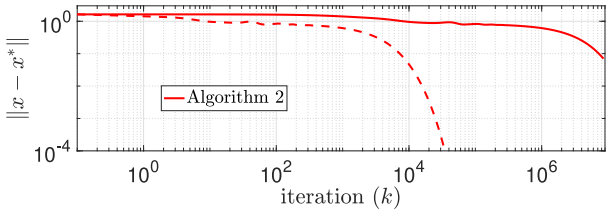

Fig. 4. Distance from the NE for Algorithm 2, on a time-varying digraph. Since the networks are sparse, Theorem 2 ensures convergence only for small step sizes ($\gamma = 5 * 10^{-4}$, $\tau = 3 * 10^{-4}$), and convergence is slow (solid line). However, the bounds are conservative: the iteration still converges with τ 1000 times larger than the theoretical value (dashed line).

inexact ADMM in [13] and the accelerated gradient method in [14], for the step sizes that ensure convergence. Specifically, we set α as in Theorem 1 for Algorithm 1. The convergence of all the other Algorithms is based on the monotonicity of $\boldsymbol{F}_a$ in (14); hence we set γ as in Theorem 2. Instead of using the conservative bounds in (15) for the parameters, $\bar{\mu}$ and $\bar{\ell}$, we obtain a better result by computing the values numerically. $\boldsymbol{F}_a$ is (non-restricted) strongly monotone for our parameters, hence also the convergence result for [14, Algorithm 2] holds. Figure 1 shows that Algorithm 1 outperforms all the other methods (we also note that the accelerated gradient in [14, Algorithm 2] requires two projections and two communications per iterations). As a numerical example, we also compare Algorithm 1 with the scheme in (12) by removing the local constraints, in Figure 2.

For the case of doubly stochastic time-varying networks, we randomly generate 5 connected graphs and for each iteration we pick one with uniform distribution. In Figure 3, we compare the performance of Algorithms 1-2, for step sizes set to their upper bounds as in Theorems 1-2. Since the theoretical convergence rate in Theorems 1-2 worsens as the number of agents grows, to show how the performance is affected in practice, we repeat the experiment for different values of N and random market structures (Figure 3).

Finally, in Figure 4, we test Algorithm 2 with communication topology chosen at each step with uniform distribution between two unweighted balanced directed graphs: the directed ring, where each agent i can send information to the agent $i+1$ (with the convention $N+1 \equiv 1$), and a graph where agent i is also allowed to transmit to agent $i+2$, for all $i \in \mathcal{I}$.

VI. Conclusion

Nash equilibrium problems on time-varying graphs can be solved with linear rate via fixed-step pseudo-gradient algorithms, if the network is connected at every iteration and the game mapping is Lipschitz continuous and strongly monotone. Our algorithm proved much faster than the existing gradient-based methods, when the step sizes satisfy their theoretical upper bounds. The extension to games with coupling constraints is left as future research. It would be also valuable to relax our uniform connectedness assumption, i.e., allowing for jointly strongly connected directed graphs.

Appendix

A. Proof of Theorem 1

We define the estimate consensus subspace $\boldsymbol{E} := \{\boldsymbol{y} \in \mathbb{R}^{Nn} | \boldsymbol{y} = \mathbf{1}_N \otimes y,\ y \in \mathbb{R}^n\}$ and its orthogonal complement $\boldsymbol{E}_\perp = \{\boldsymbol{y} \in \mathbb{R}^{Nn} | (\mathbf{1}_N \otimes I_n)^\top \boldsymbol{y} = \mathbf{0}_n\}$. Thus, any vector $\boldsymbol{x} \in \mathbb{R}^{Nn}$ can be written as $\boldsymbol{x} = \boldsymbol{x}_\parallel + \boldsymbol{x}_\perp$, where $\boldsymbol{x}_\parallel = \text{proj}_{\boldsymbol{E}}(\boldsymbol{x}) = \frac{1}{N}(\mathbf{1}_N \mathbf{1}_N^\top \otimes I_n)\boldsymbol{x}$, $\boldsymbol{x}_\perp = \text{proj}_{\boldsymbol{E}_\perp}(\boldsymbol{x})$, and $\boldsymbol{x}_\parallel^\top \boldsymbol{x}_\perp = 0$. Also, we use the shorthand notation $\boldsymbol{F}\boldsymbol{x}$ and Fx in place of $\boldsymbol{F}(\boldsymbol{x})$ and $F(x)$. We recast the iteration in (7) as

$$\boldsymbol{x}^{k+1} = \text{proj}_{\boldsymbol{\Omega}}(\hat{\boldsymbol{x}}^k - \alpha \mathcal{R}^\top \boldsymbol{F}\hat{\boldsymbol{x}}^k),\ \hat{\boldsymbol{x}}^k = \boldsymbol{W}_k \boldsymbol{x}^k. \tag{16}$$

Let x^* be the unique NE of the game in (1), and $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$. We recall that $x^* = \text{proj}_\Omega(x^* - \alpha F x^*)$ by (2), and then $\boldsymbol{x}^* = \text{proj}_{\boldsymbol{\Omega}}(\boldsymbol{x}^* - \alpha \mathcal{R}^\top \boldsymbol{F}\boldsymbol{x}^*)$. Moreover, $\boldsymbol{W}_k \boldsymbol{x}^* = (W_k \otimes I_n)(\mathbf{1}_N \otimes x^*) = \mathbf{1}_N \otimes x^* = \boldsymbol{x}^*$; hence $\boldsymbol{x}^*$ is a fixed point for (16). Let $\boldsymbol{x}^k = \boldsymbol{x} \in \mathbb{R}^{Nn}$ and $\hat{\boldsymbol{x}} = \boldsymbol{W}_k \boldsymbol{x} = \hat{\boldsymbol{x}}_\parallel + \hat{\boldsymbol{x}}_\perp = \mathbf{1}_N \otimes \hat{x}_\parallel + \hat{\boldsymbol{x}}_\perp \in \mathbb{R}^{Nn}$. Thus, it holds that

$$\begin{aligned}
&\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\|^2 \\
&= \|\text{proj}_{\boldsymbol{\Omega}}(\hat{\boldsymbol{x}} - \alpha \mathcal{R}^\top \boldsymbol{F}\hat{\boldsymbol{x}}) - \text{proj}_{\boldsymbol{\Omega}}(\boldsymbol{x}^* - \alpha \mathcal{R}^\top \boldsymbol{F}\boldsymbol{x}^*)\|^2 \\
&\le \|(\hat{\boldsymbol{x}} - \alpha \mathcal{R}^\top \boldsymbol{F}\hat{\boldsymbol{x}}) - (\boldsymbol{x}^* - \alpha \mathcal{R}^\top \boldsymbol{F}\boldsymbol{x}^*)\|^2 \\
&= \|\hat{\boldsymbol{x}}_\parallel + \hat{\boldsymbol{x}}_\perp - \boldsymbol{x}^* + \alpha \mathcal{R}^\top(-\boldsymbol{F}\hat{\boldsymbol{x}} + \boldsymbol{F}\boldsymbol{x}^* + \boldsymbol{F}\hat{\boldsymbol{x}}_\parallel - \boldsymbol{F}\hat{\boldsymbol{x}}_\parallel)\|^2 \\
&= \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|^2 + \|\hat{\boldsymbol{x}}_\perp\|^2 \\
&\quad + \alpha^2 \|\mathcal{R}^\top(\boldsymbol{F}\hat{\boldsymbol{x}} - \boldsymbol{F}\hat{\boldsymbol{x}}_\parallel + \boldsymbol{F}\hat{\boldsymbol{x}}_\parallel - \boldsymbol{F}\boldsymbol{x}^*)\|^2 \\
&\quad - 2\alpha(\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*)^\top \mathcal{R}^\top(\boldsymbol{F}\hat{\boldsymbol{x}} - \boldsymbol{F}\hat{\boldsymbol{x}}_\parallel)
\end{aligned}$$

$$
\begin{aligned}
&- 2\alpha(\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*)^\top \mathcal{R}^\top (\boldsymbol{F}\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{F}\boldsymbol{x}^*) \\
&- 2\alpha\hat{\boldsymbol{x}}_{\perp}^\top \mathcal{R}^\top (\boldsymbol{F}\hat{\boldsymbol{x}} - \boldsymbol{F}\hat{\boldsymbol{x}}_{\parallel}) \\
&- 2\alpha\hat{\boldsymbol{x}}_{\perp}^\top \mathcal{R}^\top (\boldsymbol{F}\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{F}\boldsymbol{x}^*) \qquad (17) \\
\leq\ & \|\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*\|^2 + \|\hat{\boldsymbol{x}}_{\perp}\|^2 + \alpha^2(\ell^2\|\hat{\boldsymbol{x}}_{\perp}\|^2 + \tfrac{\ell_0^2}{N}\|\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*\|^2 \\
&+ \tfrac{2\ell_0\ell}{\sqrt{N}}\|\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*\|\|\hat{\boldsymbol{x}}_{\perp}\|) + \tfrac{2\alpha\ell}{\sqrt{N}}\|\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*\|\|\hat{\boldsymbol{x}}_{\perp}\| \\
&- \tfrac{2\alpha\mu}{N}\|\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*\|^2 + 2\alpha\ell\|\hat{\boldsymbol{x}}_{\perp}\|^2 + \tfrac{2\alpha\ell_0}{\sqrt{N}}\|\hat{\boldsymbol{x}}_{\perp}\|\|\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*\|,
\end{aligned}
$$

where the first inequality follows by nonexpansiveness of the projection ([22, Prop. 4.16]), and to bound the addends in (17) we used, in the order:

- 3rd term: $\|\mathcal{R}\| = 1$, Lipschitz continuity of $\boldsymbol{F}$, and $\|\boldsymbol{F}\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{F}\boldsymbol{x}^*\| = \|F\hat{x}_{\parallel} - Fx^*\| \leq \ell_0\|\hat{x}_{\parallel} - x^*\| = \frac{\ell_0}{\sqrt{N}}\|\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*\|$;
- 4th term: $\|\mathcal{R}(\mathbf{1}\otimes(\hat{x}_{\parallel} - x^*))\| = \|\hat{x}_{\parallel} - x^*\| = \frac{1}{\sqrt{N}}\|\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*\|$;
- 5th term: $(\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*)^\top \mathcal{R}^\top (\boldsymbol{F}\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{F}\boldsymbol{x}^*) = (\hat{x}_{\parallel} - x^*)^\top (F\hat{x}_{\parallel} - Fx^*) \geq \mu\|\hat{x}_{\parallel} - x^*\|^2 = \frac{1}{N}\|\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*\|^2$;
- 6th term: Lipschitz continuity of $\boldsymbol{F}$;
- 7th term: $\|\boldsymbol{F}\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{F}\boldsymbol{x}^*\| \leq \frac{\ell_0}{\sqrt{N}}\|\hat{\boldsymbol{x}}_{\parallel} - \boldsymbol{x}^*\|$ as above.

Besides, for every $\boldsymbol{x} = \boldsymbol{x}_{\parallel} + \boldsymbol{x}_{\perp} \in \mathbb{R}^{Nn}$ and for all $k \in \mathbb{N}$, it holds that $\hat{\boldsymbol{x}} = \boldsymbol{W}_k\boldsymbol{x} = \boldsymbol{x}_{\parallel} + \boldsymbol{W}_k\boldsymbol{x}_{\perp}$, where $\boldsymbol{W}_k\boldsymbol{x}_{\perp} \in \boldsymbol{E}_{\perp}$, by doubly stochasticity of W_k, and $\|\hat{\boldsymbol{x}}_{\perp}\| = \|\boldsymbol{W}_k\boldsymbol{x}_{\perp}\| \leq \bar{\sigma}\|\boldsymbol{x}_{\perp}\|$ by (4) and properties of the Kronecker product. Therefore we can finally write, for all $k \in \mathbb{N}$, for all $\boldsymbol{x}^k \in \mathbb{R}^{Nn}$,

$$
\begin{aligned}
\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\|^2 &\leq \begin{bmatrix} \|\boldsymbol{x}_{\parallel}^k - \boldsymbol{x}^*\| \\ \|\boldsymbol{x}_{\perp}^k\| \end{bmatrix}^\top M_\alpha \begin{bmatrix} \|\boldsymbol{x}_{\parallel}^k - \boldsymbol{x}^*\| \\ \|\boldsymbol{x}_{\perp}^k\| \end{bmatrix} \\
&\leq \lambda_{\max}(M_\alpha)(\|\boldsymbol{x}_{\parallel}^k - \boldsymbol{x}^*\|^2 + \|\boldsymbol{x}_{\perp}^k\|^2) \\
&= \lambda_{\max}(M_\alpha)\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2.
\end{aligned}
$$

B. Proof of Theorem 2

Let x^* be the unique NE of the game in (1), and $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$. We recall that the null space $\text{null}(\boldsymbol{L}_k) = \boldsymbol{E} = \{\boldsymbol{y} \in \mathbb{R}^{Nn} | \boldsymbol{y} = \mathbf{1}_N \otimes y,\ y \in \mathbb{R}^n\}$ by Standing Assumption 3. Therefore, $\boldsymbol{L}_k\boldsymbol{x}^* = \mathbf{0}_N$ and $\boldsymbol{x}^*$ is a fixed point of the iteration in (13) by (2). With $\boldsymbol{F}_{\text{a}}^k$ as in (14), for all $k \in \mathbb{N}$, for any $\boldsymbol{x} \in \mathbb{R}^{Nn}$, it holds that $(\boldsymbol{x} - \boldsymbol{x}^*)^\top(\boldsymbol{F}_{\text{a}}^k\boldsymbol{x} - \boldsymbol{F}_{\text{a}}^k\boldsymbol{x}^*) = (\boldsymbol{x} - \boldsymbol{x}^*)^\top\gamma\mathcal{R}^\top(\boldsymbol{F}\boldsymbol{x} - \boldsymbol{F}\boldsymbol{x}^*) + (\boldsymbol{x} - \boldsymbol{x}^*)^\top\boldsymbol{L}_k(\boldsymbol{x} - \boldsymbol{x}^*) = (\boldsymbol{x} - \boldsymbol{x}^*)^\top\gamma\mathcal{R}^\top(\boldsymbol{F}\boldsymbol{x} - \boldsymbol{F}\boldsymbol{x}^*) + (\boldsymbol{x} - \boldsymbol{x}^*)^\top\tilde{\boldsymbol{L}}_k(\boldsymbol{x} - \boldsymbol{x}^*)$, where $\tilde{\boldsymbol{L}}_k = (\boldsymbol{L}_k + \boldsymbol{L}_k^\top)/2 = (L_k + L_k^\top) \otimes I_n/2 = \tilde{L}_k \otimes I_n$, and $\tilde{L}_k$ is the Laplacian of a connected graph (see Section IV) and $\lambda_2(\tilde{L}_k) > \bar{\lambda}$ by Assumption 4. Therefore we can apply [15, Lemma 3] to conclude that $(\boldsymbol{x} - \boldsymbol{x}^*)^\top(\boldsymbol{F}_{\text{a}}^k\boldsymbol{x} - \boldsymbol{F}_{\text{a}}^k\boldsymbol{x}^*) \geq \bar{\mu}\|\boldsymbol{x} - \boldsymbol{x}^*\|^2$, with $\bar{\mu} > 0$ as in (15). Also, $\boldsymbol{F}_{\text{a}}^k$ is Lipschitz continuous with constant $\bar{\ell} = \ell + \tilde{\sigma}$, $\tilde{\sigma}$ as in Assumption 4. Therefore we have

$$
\begin{aligned}
&\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\|^2 \\
&= \|\text{proj}_{\boldsymbol{\Omega}}(\boldsymbol{x}^k - \tau\boldsymbol{F}_{\text{a}}^k(\boldsymbol{x}^k)) - \text{proj}_{\boldsymbol{\Omega}}(\boldsymbol{x}^* - \tau\boldsymbol{F}_{\text{a}}^k\boldsymbol{x}^*)\|^2 \\
&\leq \|(\boldsymbol{x}^k - \tau\boldsymbol{F}_{\text{a}}^k\boldsymbol{x}^k) - (\boldsymbol{x}^* - \tau\boldsymbol{F}_{\text{a}}^k\boldsymbol{x}^*)\|^2 \\
&= \|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2 - 2\tau(\boldsymbol{x}^k - \boldsymbol{x}^*)^\top(\boldsymbol{F}_{\text{a}}^k\boldsymbol{x}^k - \boldsymbol{F}_{\text{a}}^k\boldsymbol{x}^*) \\
&\quad + \tau^2\|\boldsymbol{F}_{\text{a}}^k\boldsymbol{x}^k - \boldsymbol{F}_{\text{a}}^k\boldsymbol{x}^*\|^2 \\
&\leq (1 - 2\tau\bar{\mu} + \tau^2(\ell + \tilde{\sigma})^2)\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2 = \rho_{\gamma,\tau}\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2,
\end{aligned}
$$

where in the first inequality we used [22, Prop. 4.16], and $\rho_{\gamma,\tau} \in (0, 1)$ if τ is chosen as in Theorem 2.

REFERENCES

[1] F. Facchinei and J. Pang, "Nash equilibria: The variational approach," in *Convex Optimization in Signal Processing and Communications*, D. P. Palomar and Y. C. Eldar, Eds. Cambridge, U.K.: Cambridge Univ. Press, 2009, pp. 443–493.

[2] W. Saad, Z. Han, H. V. Poor, and T. Basar, "Game-theoretic methods for the smart grid: An overview of microgrid systems, demand-side management, and smart grid communications," *IEEE Signal Process. Mag.*, vol. 29, no. 5, pp. 86–105, Sep. 2012.

[3] S. Grammatico, "Dynamic control of agents playing aggregative games with coupling constraints," *IEEE Trans. Autom. Control*, vol. 62, no. 9, pp. 4537–4548, Sep. 2017.

[4] N. Li, L. Chen, and M. A. Dahleh, "Demand response using linear supply function bidding," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1827–1838, Jul. 2015.

[5] C. Yu, M. van der Schaar, and A. H. Sayed, "Distributed learning for stochastic generalized Nash equilibrium problems," *IEEE Trans. Signal Process.*, vol. 65, no. 15, pp. 3893–3908, Aug. 2017.

[6] G. Belgioioso and S. Grammatico, "Projected-gradient algorithms for generalized equilibrium seeking in aggregative games are preconditioned forward-backward methods," in *Proc. Eur. Control Conf.*, 2018, pp. 2188–2193.

[7] J. S. Shamma and G. Arslan, "Dynamic fictitious play, dynamic gradient play, and distributed convergence to Nash equilibria," *IEEE Trans. Autom. Control*, vol. 50, no. 3, pp. 312–327, Mar. 2005.

[8] J. Ghaderi and R. Srikant, "Opinion dynamics in social networks with stubborn agents: Equilibrium and convergence rate," *Automatica*, vol. 50, no. 12, pp. 3209–3215, 2014.

[9] K. Bimpikis, S. Ehsani, and R. Ilkiliç, "Cournot competition in networked markets," in *Proc. 15th ACM Conf. Econ. Comput. (EC)*, 2014, p. 733.

[10] J. Koshal, A. Nedić, and U. V. Shanbhag, "Distributed algorithms for aggregative games on graphs," *Oper. Res.*, vol. 64, no. 3, pp. 680–704, 2016.

[11] M. Ye and G. Hu, "Distributed Nash equilibrium seeking by a consensus based approach," *IEEE Trans. Autom. Control*, vol. 62, no. 9, pp. 4811–4818, Sep. 2017.

[12] D. Gadjov and L. Pavel, "A passivity-based approach to Nash equilibrium seeking over networks," *IEEE Trans. Autom. Control*, vol. 64, no. 3, pp. 1077–1092, Mar. 2019.

[13] 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, May 2019.

[14] T. Tatarenko, W. Shi, and A. Nedić. (2018). *Geometric Convergence of Gradient Play Algorithms for Distributed Nash Equilibrium Seeking*. [Online]. Available: https://arxiv.org/abs/1809.07383

[15] L. Pavel, "Distributed GNE seeking under partial-decision information over networks via a doubly-augmented operator splitting approach," *IEEE Trans. Autom. Control*, vol. 65, no. 4, pp. 1584–1597, Apr. 2020.

[16] T. Tatarenko and A. Nedić. (2019). *Geometric Convergence of Distributed Gradient Play in Games With Unconstrained Action Sets*. [Online]. Available: https://arxiv.org/abs/1907.07144

[17] G. Belgioioso, A. Nedić, and S. Grammatico. (2019). *Distributed Generalized Nash Equilibrium Seeking in Aggregative Games on Time-Varying Networks*. [Online]. Available: https://arxiv.org/abs/1907.00191

[18] F. Salehisadaghiani and L. Pavel, "Nash equilibrium seeking with non-doubly stochastic communication weight matrix," *EAI Endorsed Trans. Collab. Comput.*, vol. 4, no. 13, pp. 1–11, 2019.

[19] F. Facchinei and J. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*. New York, NY, USA: Springer, 2007.

[20] B. Gharesifard and J. Cortés, "Distributed strategies for generating weight-balanced and doubly stochastic digraphs," *Eur. J. Control*, vol. 18, no. 6, pp. 539–557, 2012.

[21] M. Bianchi and S. Grammatico. (2020). *A Continuous-Time Distributed Generalized Nash Equilibrium Seeking Algorithm Over Networks for Double-Integrator Agents*. [Online]. Available: https://arxiv.org/abs/1910.11608

[22] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, vol. 2011. Cham, Switzerland: Springer, 2017.