A distributed Bregman forward-backward algorithm for a class of Nash equilibrium problems

Ananduta, Wicak; Grammatico, Sergio

DOI
10.1016/j.ejcon.2022.100686

Publication date
2022

Document Version
Final published version

Published in
European Journal of Control

European Journal of Control 68 (2022) 100686

Contents lists available at ScienceDirect

European Journal of Control

journal homepage: www.elsevier.com/locate/ejcon

A distributed Bregman forward-backward algorithm for a class of Nash equilibrium problems☆

Wicak Ananduta*, Sergio Grammatico

The Delft Center of Systems and Control (DCSC), TU Delft, the Netherlands

ARTICLE INFO

Article history:
Received 16 April 2022
Accepted 6 June 2022
Available online 14 June 2022

Recommended by Prof. T Parisini

Keywords:
Nash equilibrium seeking problems
Distributed algorithms
Non-cooperative games

ABSTRACT

We present a distributed Nash equilibrium seeking method based on the Bregman forward-backward splitting, which allows us to have a mirror mapping instead of the standard projection as the backward operator. Our main technical contribution is to show convergence to a Nash equilibrium when the game has cocoercive pseudogradient mapping. Furthermore, when the feasible sets of the agents are simplices, a suitable choice of a Legendre function results in an exponentiated pseudogradient method, which, in our numerical experience, performs out the standard projected pseudogradient and dual averaging methods.

1. Introduction

Game theory has been widely used in analyzing and computing solutions of prominent problems not only in economics but also in engineering domains, such as traffic flow networks [1,17], power systems [20], and energy management systems [2,7,19,24], since it deals with optimization problems of multi-agent systems, where the objective function of each agent is coupled through the decision variables of other agents. In such decision-making problems, a Nash equilibrium (NE) is a collective decision where no agent has an incentive to unilaterally deviate.

A considerable effort has been given to designing distributed NE seeking algorithms, e.g. [9,13,14,18,23,25]. Typically, these algorithms perform a Euclidean projection at each iteration to ensure the feasibility of decisions with respect to their local constraints, usually assumed to be compact and convex. In practice, a projection step requires each agent to solve a convex programming problem.

The main research question studied in this paper is on how to exploit the structure of the feasible sets in order to reduce the computational effort of performing the projection step. In addressing this question, we turn our attention to the mirror descent approach for convex optimization, pioneered by Nemirovski and Yudin [6,15]. In this method, the gradient step is mapped onto the feasible set via a so-called mirror mapping, which is based on the Bregman distance of a well-behaved function. In fact, mirror mappings are a generalization of the Euclidean projection. An application of this approach to convex optimization problems with simplex feasible sets is studied in [6], where the negative entropy function is exploited to obtain a mirror map onto a simplex in a closed-form.

A generalization of the mirror descent method for monotone inclusion problems is presented in [10], which proposes the Bregman forward-backward (B-FB) splitting. In fact, the B-FB subsumes the standard forward-backward (FB) splitting [5, Section 26.5] and provides an extra degree of freedom, that is the choice of the regularizer function used to define the Bregman distance, which closely relates to the (possibly non-Euclidean) projection step. On the other hand, a mirror-based approach to solve NE problems has also been proposed in [15], with a method based on the dual averaging (DA) [16], also called the lazy mirror descent (technically, not a special instance of the B-FB).

In this paper, we propose a distributed implementation of the B-FB splitting method suitable to compute an NE of cocoercive games, which include strongly monotone games as special cases [9,18,23]. In particular, when the feasible sets of the agents are simplices, e.g., in a Nash–Cournot market [25], an atomic splittable congestion game [1,17], or a peer-to-peer energy market [24], we can obtain an *exponentiated* pseudogradient algorithm, which has a closed-form non-Euclidean projection onto simplices. In Section 4, we provide a numerical simulation study to illustrate the benefits of our algorithm.

☆ This work was partially funded by the ERC under research project COSMOS (802348).

* Corresponding author.
E-mail addresses: w.ananduta@tudelft.nl (W. Ananduta), s.grammatico@tudelft.nl (S. Grammatico).

https://doi.org/10.1016/j.ejcon.2022.100686

1.1. Notation and definitions

The set of real numbers is denoted by $\mathbb{R}$ whereas that of the extended real numbers is denoted by $\overline{\mathbb{R}} := \mathbb{R} \cup \{+\infty\}$. The vector of all 1 (or 0) with dimension n is denoted by $\mathbf{1}_n$ ($\mathbf{0}_n$). We omit the subscript when the dimension is clear from the context. The operator $\mathrm{col}(\cdot)$ stacks the arguments column-wise. The cardinality of a set is denoted by $|\cdot|$. The interior and relative interior of a set are denoted by int and rint, respectively. Let $P \succ 0$ be symmetric. For $x, y \in \mathbb{R}^n$, $\langle x, y\rangle_P = \langle x, Py\rangle$ and $\|x\|_P$ denote the P-weighted Euclidean inner product and norm, respectively. The operator $\mathrm{proj}_{\mathcal{A}} : \mathbb{R}^n \to \mathcal{A}$ is the Euclidean projection onto a closed convex set $\mathcal{A} \in \mathbb{R}^n$, i.e., $\mathrm{proj}_{\mathcal{A}}(x) := \mathrm{argmin}_{y\in\mathcal{A}} \|y - x\|^2$.

The graph of an operator $A : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is denoted by $\mathrm{gph}(A)$. $\mathrm{zer}(A)$ defines the set of zeros of operator A, i.e., $\mathrm{zer}(A) = \{x \in \mathrm{dom}(A) \mid 0 \in A(x)\}$. An operator $A : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is ξ-strongly monotone if, for any $(x, y) \in \mathrm{gph}(A)$ and $(x', y') \in \mathrm{gph}(A)$, $\langle y - y', x - x'\rangle \geq \xi \|x - x'\|^2$ [5, Def. 22.1.e]. Moreover, an operator $A : \mathbb{R}^n \to \mathbb{R}^n$ is L-Lipschitz continuous if there exists a constant $L > 0$, such that, for all $x, x' \in \mathbb{R}^n$, $\|A(x) - A(x')\| \leq L\|x - x'\|$ [5, Def. 1.47] and A is β-cocoercive if there exists $\beta > 0$ such that for all $x, x' \in \mathbb{R}^n$, $\langle A(x) - A(x'), x - x'\rangle \geq \beta\|A(x) - A(x')\|^2$ [5, Def. 4.10]. A continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is ξ-strongly convex with $\xi > 0$, if, for all $x, x' \in \mathrm{dom}(f)$, $f(x') \geq f(x) + \langle \nabla f(x), x' - x\rangle + \frac{\xi}{2}\|x' - x\|^2$. Moreover, it also holds that $\langle \nabla f(x') - \nabla f(x), x' - x\rangle \geq \xi \|x' - x\|^2$. Additionally, f is convex if the two previous inequalities hold for $\xi = 0$. For a nonempty convex set $\mathcal{C}$, $\mathrm{N}_{\mathcal{C}}$ denotes its normal cone operator [5, Eq. (6.35)].

Definition 1 [3, Def. 3.1]**.** Let φ be a closed, convex, and proper function on $\mathbb{R}^n$ with $\mathrm{int}\,\mathrm{dom}(\varphi) \neq \varnothing$, and differentiable on $\mathrm{int}\,\mathrm{dom}(\varphi)$. The Bregman distance associated with φ, denoted by $\mathrm{dist}_\varphi : \mathbb{R}^n \times \mathrm{int}\,\mathrm{dom}(\varphi) \to \overline{\mathbb{R}}$, is defined as

$$\mathrm{dist}_\varphi(x, y) = \varphi(x) - \varphi(y) - \langle \nabla\varphi(y), x - y\rangle.$$

Definition 2 ([From [3]]). Let φ be a closed, convex, and proper function on $\mathbb{R}^n$. Then, φ is Legendre if

1. φ is essentially smooth, i.e., continuously differentiable on $\mathrm{int}\,\mathrm{dom}(\varphi)$ [3, Def. 2.1];
2. φ is essentially strictly convex, i.e., strictly convex on every convex subset of $\mathrm{int}\,\mathrm{dom}(\varphi)$ [3, Def. 2.3].

2. Cocoercive Nash equilibrium problems

2.1. Nash equilibrium problems

Let $\mathcal{I} := \{1, 2, \ldots, N\}$ denote the index set of agents playing a game in the form:

$$\forall i \in \mathcal{I}: \quad \min_{x_i \in \mathcal{X}_i} \; J_i(x_i, \boldsymbol{x}_{\mathcal{N}_i}) \tag{1}$$

where $x_i \in \mathbb{R}^{n_i}$ denotes the decision variable of agent i whereas $\boldsymbol{x}_{\mathcal{N}_i} \in \mathbb{R}^{\sum_{j\in\mathcal{N}_i} n_j}$ denotes the collection of decision variables of a subset of other agents $\mathcal{N}_i \subseteq \mathcal{I}\backslash\{i\}$, which is called the set of neighbors of agent i. We also denote by $\boldsymbol{x} = \mathrm{col}(\{x_i\}_{i\in\mathcal{I}}) \in \mathbb{R}^n$ the collection of decision variables of all agents, where $n = \sum_{i\in\mathcal{I}} n_i$. Furthermore, the cost function of agent i is denoted by $J_i(\cdot)$, which not only depends on x_i but also $\boldsymbol{x}_{\mathcal{N}_i}$, whereas $\mathcal{X}_i \subset \mathbb{R}^{n_i}$ denotes the action space of agent i and $\boldsymbol{\mathcal{X}} = \prod_{i\in\mathcal{I}} \mathcal{X}_i$ denotes the overall action space.

We aim at computing an NE of the game in (1), which is formally defined as follows.

Definition 3. A point $\boldsymbol{x}^\star = \mathrm{col}(\{x_i^\star\}_{i\in\mathcal{I}}) \in \boldsymbol{\mathcal{X}}$ is a Nash equilibrium if, for each $i \in \mathcal{I}$,

$$J_i(x_i^\star, \boldsymbol{x}_{\mathcal{N}_i}^\star) \leq \inf_{y\in\mathcal{X}_i} J_i(y, \boldsymbol{x}_{\mathcal{N}_i}^\star).$$

Furthermore, we consider the following assumptions:

Assumption 1. The cost function $J_i(\cdot, \boldsymbol{x}_{\mathcal{N}_i})$ in (1), for each $i \in \mathcal{I}$ and for any $\boldsymbol{x}_{\mathcal{N}_i}$, is convex and continuously differentiable.

Assumption 2. For each $i \in \mathcal{I}$, the set $\mathcal{X}_i$ is nonempty, compact, and convex.

Assumption 3. The pseudogradient mapping,

$$F(\boldsymbol{x}) := \mathrm{col}(\{\nabla_{x_i} J_i(x_i, \boldsymbol{x}_{\mathcal{N}_i})\}_{i\in\mathcal{I}}), \tag{2}$$

is β-cocoercive, for some $\beta > 0$.

Under Assumptions 1 and 2, an NE of the game in (1), $\boldsymbol{x}^\star$, exists [12, Cor. 2.2.5], and it is a solution to the variational inequality $\mathrm{VI}(F, \boldsymbol{\mathcal{X}})$, i.e., it holds that $\langle F(\boldsymbol{x}^\star), \boldsymbol{x} - \boldsymbol{x}^\star\rangle \geq 0$, for all $\boldsymbol{x} \in \boldsymbol{\mathcal{X}}$ [12, Prop. 1.4.2]. While cocoercivity is stronger than monotonicity, Assumption 3 is the weakest assumption for FB splitting methods [10], [5, Thm. 26.14].

Remark 1. If F is ξ-strongly monotone and L-Lipschitz continuous, then F is (ξ/L^2)-cocoercive [12, page 164]. Strong monotonicity of F in (2) is assumed in [9,23,25].

As a motivation of this work, we consider games with a special structure of its local feasible sets $\mathcal{X}_i$, namely simplex sets, which satisfy Assumption 2.

Assumption 4. For each $i \in \mathcal{I}$, the set $\mathcal{X}_i$ is defined as

$$\mathcal{X}_i := \{y \in \mathbb{R}^{n_i}_{\geq 0} \mid \mathbf{1}^\top y = 1\}. \tag{3}$$

In the following, let us provide some examples of cocoercive games with such feasible sets.

2.2. Nash–Cournot market with production constraint

In a Nash–Cournot market [25], a set of producers (agents) $\mathcal{I}$ supplies (divisible) commodities to a set of markets $\mathcal{M}$. Let us suppose that agent i distributes its product to a subset $\mathcal{M}_i \subseteq \mathcal{M}$ and the total production output, denoted by g_i, is fixed. Therefore, the decision variable of agent i is $x_i := \mathrm{col}(\{x_i^m\}_{m\in\mathcal{M}_i})$, where $x_i^m \in [0, 1]$ denotes the proportion of the total product distributed to market $m \in \mathcal{M}_i$, i.e., agent i sells $x_i^m g_i$ quantity to market m. Given a fixed production, then it must hold that $x_i \in \mathcal{X}_i$, where $\mathcal{X}_i$ satisfies Assumption 4. The cost function of each agent i is defined by [25, Eq. (36)]:

$$J_i(\boldsymbol{x}) = c_i(x_i) - \sum_{m\in\mathcal{M}_i} p_m(\boldsymbol{x}) g_i x_i^m,$$

where $c_i(x_i)$ is a strongly convex production cost with Lipschitz continuous gradient and $p_m(\boldsymbol{x})$ is the unit price function of market m which follows a linear inverse demand function, i.e.,

$$p_m(\boldsymbol{x}) = \bar{p}_m - d_m \sum_{i\in\mathcal{I} \text{ s.t. } m\in\mathcal{M}_i} g_i x_i^m, \quad \forall m \in \mathcal{M},$$

where $\bar{p}_m, d_m > 0$ are the price parameters. As shown in [25, Section 7.1], the pseudogradient F (2) of this game is strongly monotone and Lipschitz continuous. Thus, by Remark 1, it satisfies Assumption 3.

2.3. Atomic splittable congestion game

An atomic splittable congestion game is a non-cooperative model of agents routing flow in a network (e.g. communication network) [17], [15, Ex. 2.3]. In this game, a set of agents $\mathcal{I}$ uses a network with a common source node, a common sink node, and a set of parallel communication links connecting the nodes, denoted by $\mathcal{L}$. Each agent $i \in \mathcal{I}$ transports its demand, denoted by $d_i > 0$

from the source to the sink by splitting it and by using a subset of available links denoted by $\mathcal{L}_i \subseteq \mathcal{L}$. The decision variable of agent i is $x_i = \mathrm{col}(\{x_i^l\}_{l\in\mathcal{L}_i})$, where $x_i^l \in [0,1]$ denotes the demand distribution for link $l \in \mathcal{L}_i$. Thus, it must hold that $\sum_{l\in\mathcal{L}_i} x_i d_i = d_i$. In other words, $x_i \in \mathcal{X}_i$, where $\mathcal{X}_i$ is as in Assumption 4.

For each agent i, the cost of using the links is defined as

$$J_i(\boldsymbol{x}) = c_f(x_i) + \sum_{l\in\mathcal{L}_i} d_i x_i^l c_l(\boldsymbol{x}),$$

where $c_f(x_i) = -a\log(x_i + d)$, for some positive a and d, is a strongly convex utility function, and $c_l(\boldsymbol{x})$ denotes the unit price function of using link (edge) $l \in \mathcal{L}$ and depends on the decisions of the other agents that use the same link. Specifically, following [1, Eq. (1)], for all $l \in \mathcal{L}$, we define

$$c_l(\boldsymbol{x}) = a_l\Big(\sum_{i\in\mathcal{I} \text{ s.t. } l\in\mathcal{L}_i} d_i x_i^l\Big)^{p_l} + b_l,$$

where $a_l, b_l, p_l > 0$ are cost parameters for link l. Theorem 1 in [1] shows that if $p_l < p^\star$, where $p^\star = (3N-1)/(N-1)$, then the terms $\sum_{l\in\mathcal{L}_i} d_i x_i^l c_l(\boldsymbol{x})$ in J_i, for all $i \in \mathcal{I}$, generate a strictly monotone pseudogradient. Since $c_f(x_i)$ is strongly convex, the pseudogradient F (2) of this game is strongly monotone and Lipschitz continuous, thus satisfying Assumption 3.

2.4. Peer-to-peer energy market

High penetration of distributed power generation units incites the concept of direct trading between consumers and small-scale producers (a peer-to-peer energy market) [24]. In several pilot projects, consumers are allowed to choose their own energy producers, denoted by the set $\mathcal{P}_i$, among a pool of producers, denoted by $\mathcal{P}$, i.e., $\mathcal{P}_i \subset \mathcal{P}$. Assuming that energy production of each producer $p \in \mathcal{P}$ is large enough to meet their consumers, a non-cooperative game from the consumer side can be obtained by considering the decision of agent i as $x_i = \mathrm{col}(\{x_i^p\}_{p\in\mathcal{P}_i})$, where $x_i^p l_i$ is the load proportion of agent i (l_i) that is met by buying energy from producer $p \in \mathcal{P}_i$. Therefore, it must hold that $x_i \in \mathcal{X}_i$, with $\mathcal{X}_i$ as defined in (3). Furthermore, the cost of electricity of each agent can be defined by

$$J_i(\boldsymbol{x}) = \sum_{p\in\mathcal{P}_i} c_p(e_p)\frac{l_i x_i^p}{e_p}, \quad \forall i \in \mathcal{I},$$

where e_p denotes the total energy generated by producer p, i.e.,

$$e_p = \sum_{i\in\mathcal{I} \text{ s.t. } p\in\mathcal{P}_i} l_i x_i^p,$$

and $c_p(e_p)$ denotes the cost function of generating power by producer p, which is typically assumed to be a quadratic function, i.e., $c_p = d_p e_p^2$, for some positive scalar d_p [2]. The pseudogradient F (2) of this game can be written as $F(\boldsymbol{x}) = Q\boldsymbol{x}$, for some symmetric and positive semi-definite matrix Q, implying that F is cocoercive [12, Section 1.5.2].

Remark 2. The game in (1) can also describe an economic dispatch problem for microgrids that have multiple distributed generation units with convex quadratic cost functions and sufficiently large production capacity [19, pp. 83–86].

3. Nash equilibrium seeking based on distributed Bregman forward-backward

In this section, we discuss how we can compute an NE of the game in (1) via a splitting method, namely the B-FB splitting, recently introduced in [10].

3.1. Distributed Bregman forward-backward splitting

The B-FB aims at solving the following inclusion problem:

$$\text{find } \boldsymbol{x} \text{ s.t. } 0 \in (A+B)(\boldsymbol{x}), \tag{4}$$

for maximally monotone A and cocoercive B with the following iteration:

$$\boldsymbol{x}^{(k+1)} = (\nabla\varphi + \gamma A)^{-1}(\nabla\varphi(\boldsymbol{x}^{(k)}) - \gamma B(\boldsymbol{x}^{(k)})), \tag{5}$$

where φ is a Legendre function and $\gamma > 0$ is the step size. We note that the main advantage of the B-FB compared to the standard FB splitting [5, Section 26.5] is the extra degree of freedom on the choice of φ. Indeed, the standard FB is recovered when we choose the Legendre function $\varphi : \mathbb{R}^n \to \mathbb{R} : \boldsymbol{x} \mapsto \frac{1}{2}\|\boldsymbol{x}\|^2$.

By Definition 3 and (2), an NE $\boldsymbol{x}^\star$ satisfies the following inclusion:

$$0 \in \mathrm{N}_{\mathcal{X}}(\boldsymbol{x}^\star) + F(\boldsymbol{x}^\star). \tag{6}$$

By Assumption 2, $\mathrm{N}_{\mathcal{X}}$ is maximally monotone [5, Ex. 20.23], whereas Assumption 3 states that F is β-cocoercive. Based on this observation, the B-FB splitting can indeed solve the above inclusion problem.

Now, we provide a distributed implementation of (5), which is not discussed in [10]. In our problem setup, not only $\boldsymbol{x}$ is composed by the decision variable of each agent i, x_i, but the operator A is also decomposable, i.e, $\boldsymbol{x} \mapsto A(\boldsymbol{x}) : \boldsymbol{x} \mapsto A_1(x_1) \times \cdots \times A_N(x_N)$, where $A_i = \mathrm{N}_{\mathcal{X}_i}$, for each $i \in \mathcal{I}$. In this regard, we choose a separable Legendre function in the form

$$\varphi(\boldsymbol{x}) := \sum_{i\in\mathcal{I}} \varphi_i(x_i), \tag{7}$$

where $\varphi_i : \mathbb{R}^{n_i} \to \mathbb{R}$, for each $i \in \mathcal{I}$, is also separable, i.e., $\varphi_i := \sum_{j=1}^{n_i} \varphi_i^j(x_i^j)$, with x_i^j denoting the j-th component of x_i and $\varphi_i^j : \mathbb{R} \to \mathbb{R}$ being Legendre (Definition 2). Furthermore, instead of using a global step-size γ as in (5), we allow each agent i to choose its own step size, denoted by $\gamma_i > 0$. Thus, by letting $\Gamma = \mathrm{diag}(\{\gamma_i I_{n_i}\}_{i\in\mathcal{I}})$, the distributed B-FB is compactly written as

$$\boldsymbol{x}^{(k+1)} = (\nabla\varphi + \Gamma A)^{-1}(\nabla\varphi - \Gamma B)(\boldsymbol{x}^{(k)}), \tag{8}$$

where φ is defined in (7) and Γ is a positive definite and diagonal step-size matrix, i.e., $\Gamma \succ 0$. Our first result is the convergence of the B-FB with step-size matrix Γ, as formally stated next.

Theorem 1. *Let A be a maximally monotone separable operator, i.e., $\boldsymbol{x} \mapsto A(\boldsymbol{x}) : \boldsymbol{x} \mapsto A_1(x_1) \times \cdots \times A_N(x_N)$, where $\mathrm{dom}(A_i) \subseteq \mathbb{R}^{n_i}$, for all $i \in \mathcal{I}$, and B be β-cocoercive, for some $\beta > 0$. Let φ be defined as in (7), with Legendre function φ_i that is ξ_{φ_i}-strongly convex on $\mathrm{int}\,\mathrm{dom}(\varphi_i) \cap \mathrm{dom}(A_i)$, for each $i \in \mathcal{I}$. Suppose that $\mathrm{zer}(A+B) \cap \mathrm{int}\,\mathrm{dom}(\varphi) \neq \varnothing$, $\gamma_i \in (0, 2\beta\xi_{\varphi_i})$, for all $i \in \mathcal{I}$, and $\boldsymbol{x}^{(0)} \in \mathrm{int}\,\mathrm{dom}(\varphi)$. Then, the sequence $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}}$ generated by (8) is well defined in $\mathrm{int}\,\mathrm{dom}(\varphi) \cap \mathrm{dom}(A)$ and converges to a point in $\mathrm{zer}(A+B) \cap \mathrm{int}\,\mathrm{dom}(\varphi)$.*

Proof. See Appendix A. □

3.2. Exponentiated gradient method for Nash equilibrium problems with simplex feasible sets

As discussed in [12, Section 12.1], for the game (1) that satisfies Assumptions 1–3, the projected pseudogradient method, which is an instance of the standard FB splitting:

$$x_i^{(k+1)} := \mathrm{proj}_{\mathcal{X}_i}(x_i^{(k)} - \gamma_i \nabla_{x_i} J_i(x_i^{(k)}, \boldsymbol{x}_{\mathcal{N}_i}^{(k)})), \ \forall i \in \mathcal{I}, \tag{9}$$

with sufficiently small step sizes γ_i, for all $i \in \mathcal{I}$, generates a sequence that converges to an NE. We can also immediately see this from Theorem 1. Specifically, with $A = \mathrm{N}_{\mathcal{X}}$, $B = F$, and the Legendre

function $\varphi(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{x}\|^2$, the B-FB iteration (8) reads as in (9). In view of Theorem 1, since $\varphi(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{x}\|^2$ is 1-strongly convex, we recover the sufficient condition of the step sizes γ_i's of the standard FB [5, Thm. 26.14], i.e., $\gamma_i \in (0, 2\beta)$, for all $i \in \mathcal{I}$.

However, for the particular case of $\mathcal{X}_i$ with simplicial structure (see Assumption 4), we may choose a different Legendre function. Specifically, inspired by the entropic descent algorithm [6], for each $i \in \mathcal{I}$, we consider the Legendre function in (7) with $\varphi_i(x_i)$ being the negative entropy [21], i.e., $\varphi_i : \mathbb{R}^{n_i}_{\geq 0} \to \mathbb{R}$,

$$\varphi_i(x_i) = \sum_{j=1}^{n_i} x_i^j \ln(x_i^j), \tag{10}$$

where we follow the convention $0\ln(0) = 0$. This Legendre function leads to Algorithm 1, which is an exponentiated pseudogradient method, as formally stated in the next proposition.

Algorithm 1 NE seeking algorithm via exponentiated pseudogradient.

Initialization: For $i \in \mathcal{I}$, set $x_i^{(0)} \in \operatorname{rint}(\mathcal{X}_i)$.
Iteration until convergence
Each agent $i \in \mathcal{I}$ **:**

1. Receives x_j^k from $j \in \mathcal{N}_i$.
2. Updates $x_i^{(k+1)} = ((x_i^1)^{(k+1)}, \dots, (x_i^{n_i})^{(k+1)})$ by

$$(x_i^j)^{(k+1)} = \frac{(x_i^j)^{(k)} \exp(-\gamma_i \nabla_{x_i^j} J_i(x_i^{(k)}, \boldsymbol{x}_{\mathcal{N}_i}^{(k)}))}{\sum_{\ell=1}^{n_i} (x_i^\ell)^{(k)} \exp(-\gamma_i \nabla_{x_i^\ell} J_i(x_i^{(k)}, \boldsymbol{x}_{\mathcal{N}_i}^{(k)}))}, \tag{11}$$

for $j = 1, \dots, n_i$.

end

Proposition 1. *Let Assumptions 1, 3, and 4 hold. Algorithm 1 is equivalent to the B-FB iteration in (8) where $A = \mathrm{N}_{\boldsymbol{\mathcal{X}}}$, $B = F$, F is as in (2), and $\varphi(\boldsymbol{x})$ is as in (7) and (10).*

Proof. See Appendix B. □

Remark 3. Algorithm 1 falls into the class of distributed algorithms under full information, i.e., each agent receives the updated decisions of its neighbors in $\mathcal{N}_i$ (see step 1), necessary to evaluate the pseudogradient $\nabla_{x_i} J_i(x_i, \boldsymbol{x}_{\mathcal{N}_i})$. Therefore, a local communication network is implicitly required by Algorithm 1.

With sufficiently small but not necessarily uniform step sizes, Algorithm 1 generates a sequence that converges to an NE of the game in (1) as formalized next.

Corollary 1. *Let Assumptions 1, 3, and 4 hold. Let $\mathcal{X}^\star$ denote the NE set of the game in (1). If $\mathcal{X}^\star \cap \mathbb{R}^n_{>0} \neq \varnothing$, then the sequence $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}}$ generated by Algorithm 1 with step sizes $\gamma_i \in (0, 2\beta)$, for all $i \in \mathcal{I}$, converges to some NE, $\boldsymbol{x}^\star \in \mathcal{X}^\star$.*

Proof. Assumptions 1,3, and 4 imply that $\mathcal{X}^\star = \operatorname{zer}(\mathrm{N}_{\boldsymbol{\mathcal{X}}} + F)$. Proposition 1 shows that Algorithm 1 is the B-FB algorithm applied to the monotone inclusion (6) with the Legendre function defined in (7) and (10). Furthermore, $\mathrm{N}_{\boldsymbol{\mathcal{X}}}$ is maximally monotone, F is β-cocoercive, and $\operatorname{zer}(\mathrm{N}_{\boldsymbol{\mathcal{X}}} + F) \neq \varnothing$ (Assumptions 3–4). Moreover, φ_i in (10), for each $i \in \mathcal{I}$, is 1-strongly convex on $\operatorname{int}\operatorname{dom}(\varphi_i) \cap \operatorname{dom}(\mathrm{N}_{\mathcal{X}_i}) = \operatorname{rint}(\mathcal{X}_i)$, with $\mathcal{X}_i$ satisfying Assumption 4 [21, Ex. 2.5]. Therefore, by Theorem 1, the sequence $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}}$ generated by Algorithm 1 with $\gamma_i \in (0, 2\beta)$, for all $i \in \mathcal{I}$, converges to a point $\boldsymbol{x}^\star \in \mathcal{X}^\star$. □

The choice of the Legendre function in (7) defines the mirror map of the gradient descent step onto the feasible set $\boldsymbol{\mathcal{X}}$. We have shown that the squared 2-norm results in the Euclidean projection. Meanwhile, the negative entropy function (10) leads to a mapping onto a simplex with a closed-form formula (11) as shown in Proposition 1.

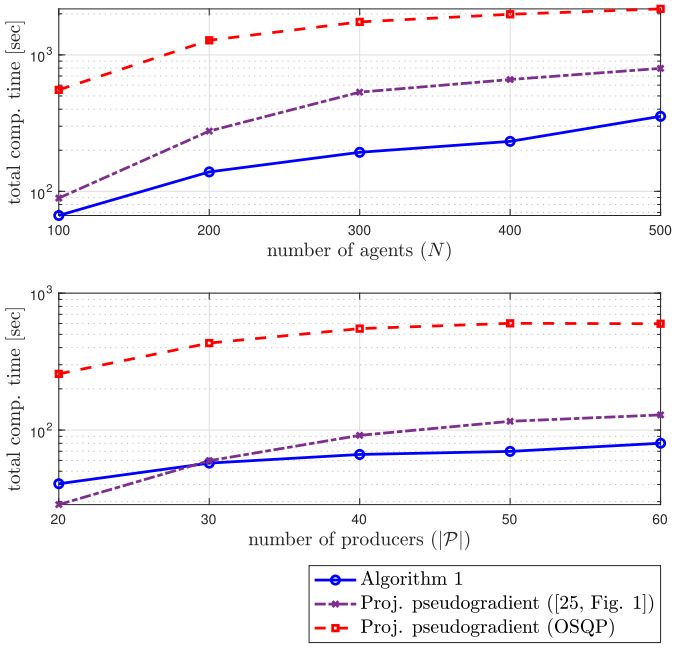

Fig. 1. The total computational time of Algorithm 1 and the standard projected pseudogradient (9) when the number of agents N varies (top plot) and the number of producers $|\mathcal{P}|$ varies (bottom plot). Each data point is the average of 100 simulations.

4. Numerical simulations

We perform a numerical study on Algorithm 1 by using the peer-to-peer market model discussed in Section 2.4. First, we compare the performance of Algorithm 1 with the standard projected pseudogradient (9). To this end, in the first set of simulations, we vary the number of agents (consumers), i.e., $N \in \{100, 200, 300, 400, 500\}$ and fix the number of producers, i.e., $|\mathcal{P}| = 40$. Meanwhile, in the second set, we fix $N = 100$, and vary $|\mathcal{P}| \in \{20, 30, 40, 50, 60\}$. Note that the dimensions of the agents' decision variables grow as $|\mathcal{P}|$ increases. For each pair of $(N, |\mathcal{P}|)$, we run 100 Monte-Carlo simulations, in which l_i, $\mathcal{P}_i \subseteq \mathcal{P}$, for all $i \in \mathcal{I}$, and the cost parameters d_p, for all $p \in \mathcal{P}$, are generated randomly. We set the step sizes of both Algorithm 1 and (9) to be equal since they have the same step-size upper bounds. The stopping criterion of these simulations is $\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\|_\infty \leq 10^{-5}$. Furthermore, we use two methods to perform the Euclidean projection step of (9): (i) the `OSQP` solver [22], which is one of the fastest quadratic programming solvers available, with the absolute and relative tolerance parameters being set to be 10^{-10} while the maximum number of iterations being set to be 10^5, and (ii) an efficient Euclidean projection algorithm onto a simplex proposed in [11, Fig. 1]. All the simulations are carried out in Matlab R2020a on a computer with Intel Xeon E5-2637 processor and 128 GB memory. Fig. 1 summarizes these simulation results and shows that the average total computational time of Algorithm 1 is the lowest due to the closed-form expression of the mirror mapping onto a simplex in almost all cases. For relatively small game instances, i.e., those where $|\mathcal{P}| = 20$ and $N = 100$ (see the bottom plot of Fig. 1), the algorithm in [11], Fig. 1 is slightly more efficient than Algorithm 1. Nevertheless, Algorithm 1 scales better with N and $|\mathcal{P}|$.

We also test Algorithm 1 against the DA method devised in [15]. We randomly generate a case, with $N = 30$, $|\mathcal{P}| = 30$, such that

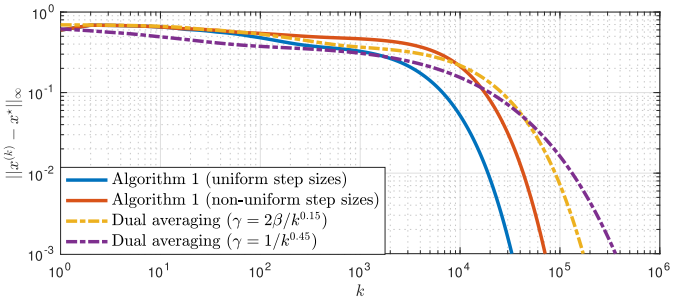

Fig. 2. Comparison between Algorithm 1 and the DA method [15] in terms of $\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^\star\|_\infty$.

the pseudogradient F is strongly monotone to ensure the convergence of the DA method. In this case, the cocoercivity constant of F (2) is $\beta = 0.032$. We run Algorithm 1 with uniform step sizes $\gamma_i = 0.063 < 2\beta$, for all $i \in \mathcal{I}$, and non-uniform step sizes γ_i, which are randomly taken from [0.032,0.063]. On the other hand, for the DA method, we consider the global vanishing step-size rules $\gamma^{(k)} = 2\beta/k^\alpha$ and $\gamma^{(k)} = 1/k^\alpha$ for some $\alpha \in (0, 1]$ [15, Rem. 4.1]. As shown in Fig. 2, Algorithm 1 converges faster than the DA method since its step sizes are constant. The choices of α for the DA method in Fig. 2 represent the step-size rules with the best rate and accuracy we have found. In addition, unlike Algorithm 1, we experienced a numerical issue with the DA method. Specifically, if the score variables, which are exponentiated and normalized to update x_i and can grow due to the gradient descent update, became a too large negative number, a computer with a limited numerical precision considers their exponents to be zero, resulting in numerical errors in the iterations.

5. Conclusion

The Bregman forward-backward splitting provides a framework to solve cocoercive Nash equilibrium problems with special structure of the feasible sets, such as simplex sets, as it allows one to exploit this structure. Our numerical study shows that the resulting exponentiated gradient algorithm converges faster than the standard projected pseudogradient methods. Finding other pairs of compact convex set structures and their associated Legendre functions and dealing with partial information scenarios are open research avenues.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Proof of Theorem 1

Well-defined iterates: Since A is maximally monotone, ΓA is maximally monotone in the Γ^{-1}-induced norm [5, Prop. 20.24]. Furthermore, since φ is a strictly convex function on $\operatorname{int}\operatorname{dom}(\varphi)$ and separable while Γ is a positive diagonal matrix, $\nabla\varphi$ is strictly monotone in the Γ^{-1}-induced norm. Hence, $(\nabla\varphi + \Gamma A)$ is also strictly monotone in the Γ^{-1}-induced norm. Now, we suppose that $(z, \boldsymbol{\omega})$ and $(z, \boldsymbol{\omega}')$ belong to $\operatorname{gph}(\nabla\varphi + \Gamma A)^{-1}$. Then $z \in (\nabla\varphi + \Gamma A)\boldsymbol{\omega}$ and $z \in (\nabla\varphi + \Gamma A)\boldsymbol{\omega}'$. Since $\langle z - z, \boldsymbol{\omega} - \boldsymbol{\omega}'\rangle_{\Gamma^{-1}} = 0$ and $(\nabla\varphi + \Gamma A)$ is strictly monotone under the Γ^{-1}-induced norm, it must hold that $\boldsymbol{\omega} = \boldsymbol{\omega}'$. Thus, we conclude that $(\nabla\varphi + \Gamma A)^{-1}$ is single-valued on $\operatorname{ran}(\nabla\varphi + \Gamma A)$. Moreover, since φ is essentially smooth, it is continuously differentiable on $\operatorname{int}\operatorname{dom}(\varphi)$ and $\operatorname{ran}(\nabla\varphi + \Gamma A)^{-1} = \operatorname{dom}(\nabla\varphi) \cap \operatorname{dom}(A) = \operatorname{int}\operatorname{dom}(\varphi) \cap \operatorname{dom}(A)$. Furthermore, since B is Lipschitz, $\nabla\varphi(\boldsymbol{x}^{(0)}) - \Gamma B(\boldsymbol{x}^{(0)})$ is singleton. We deduce $\boldsymbol{x}^{(1)} \in \operatorname{int}\operatorname{dom}(\varphi) \cap \operatorname{dom}(A)$ is uniquely defined. Therefore, by induction we have that $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}} \in \operatorname{int}\operatorname{dom}(\varphi) \cap \operatorname{dom}(A)$.

By [5, Lem. 2.46], we need to show that the sequence $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}}$ is bounded and has a unique cluster point.

Boundedness of iterates: By premultiplying both sides of the equality in (8) with $(\nabla\varphi + \Gamma A)$ and then rearranging the terms, we have that

$$0 \in \nabla\varphi(\boldsymbol{x}^{(k+1)}) + \Gamma A(\boldsymbol{x}^{(k+1)}) - \nabla\varphi(\boldsymbol{x}^{(k)}) + \Gamma B(\boldsymbol{x}^{(k)}). \tag{A.1}$$

Then, we multiply by Γ^{-1} and rearrange the inclusion to obtain that

$$(\boldsymbol{x}^{(k+1)}, \Gamma^{-1}(\nabla\varphi(\boldsymbol{x}^{(k)}) - \nabla\varphi(\boldsymbol{x}^{(k+1)})) - B(\boldsymbol{x}^{(k)})) \in \operatorname{gph}(A). \tag{A.2}$$

Moreover, let us consider $\boldsymbol{x} \in \operatorname{zer}(A + B) \cap \operatorname{int}\operatorname{dom}(\varphi)$, which implies $(\boldsymbol{x}, -B(\boldsymbol{x})) \in \operatorname{gph}(A)$ and recall that $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}} \in \operatorname{dom}(A)$. By the monotonicity of A and considering the pair $(\boldsymbol{x}, -B(\boldsymbol{x}))$ and that in (A.2), we have that

$$\begin{aligned} 0 &\le \langle A(\boldsymbol{x}) - A(\boldsymbol{x}^{(k+1)}), \boldsymbol{x} - \boldsymbol{x}^{(k+1)}\rangle \\ &= \langle \nabla\varphi(\boldsymbol{x}^{(k+1)}) - \nabla\varphi(\boldsymbol{x}^{(k)}), \boldsymbol{x} - \boldsymbol{x}^{(k+1)}\rangle_{\Gamma^{-1}} + \langle B(\boldsymbol{x}^{(k)}) \\ &\qquad - B(\boldsymbol{x}), \boldsymbol{x} - \boldsymbol{x}^{(k+1)}\rangle. \end{aligned} \tag{A.3}$$

Since Γ is a positive definite and diagonal matrix, so does $\Gamma^{-1} := \operatorname{diag}(\{\gamma_i^{-1} I_{n_i}\}_{i\in\mathcal{I}})$. Therefore, the function $\hat{\varphi}(\boldsymbol{x}) := \sum_{i\in\mathcal{I}} \gamma_i^{-1}\varphi_i(x_i)$ is a Legendre function with $\operatorname{dom}(\hat{\varphi}) = \operatorname{dom}(\varphi)$ [3, Thm. 5.12], and with gradient $\nabla\hat{\varphi} = \Gamma^{-1}\nabla\varphi$. Thus, the first addend on the right-hand side of the equality in (A.3) can be written as:

$$\begin{aligned} &\langle \nabla\varphi(\boldsymbol{x}^{(k+1)}) - \nabla\varphi(\boldsymbol{x}^{(k)}), \boldsymbol{x} - \boldsymbol{x}^{(k+1)}\rangle_{\Gamma^{-1}} \\ &\quad = \langle \nabla\hat{\varphi}(\boldsymbol{x}^{(k+1)}) - \nabla\hat{\varphi}(\boldsymbol{x}^{(k)}), \boldsymbol{x} - \boldsymbol{x}^{(k+1)}\rangle \\ &\quad = \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}, \boldsymbol{x}^{(k)}) - \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}, \boldsymbol{x}^{(k+1)}) - \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}^{(k+1)}, \boldsymbol{x}^{(k)}), \end{aligned}$$

by applying the three-point identity of Bregman distances [4, Prop. 2.3.(ii)]. Moreover, we rewrite the second addend as

$$\begin{aligned} &\langle B(\boldsymbol{x}^{(k)}) - B(\boldsymbol{x}), \boldsymbol{x} - \boldsymbol{x}^{(k+1)}\rangle \\ &\quad = \langle B(\boldsymbol{x}^{(k)}) - B(\boldsymbol{x}), \boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k+1)}\rangle + \langle B(\boldsymbol{x}^{(k)}) - B(\boldsymbol{x}), \boldsymbol{x} - \boldsymbol{x}^{(k)}\rangle. \end{aligned}$$

Therefore, (A.3) can be written as

$$\begin{aligned} 0 \le{}& \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}, \boldsymbol{x}^{(k)}) - \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}, \boldsymbol{x}^{(k+1)}) - \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}^{(k+1)}, \boldsymbol{x}^{(k)}) \\ &+ \langle B(\boldsymbol{x}^{(k)}) - B(\boldsymbol{x}), \boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k+1)}\rangle + \langle B(\boldsymbol{x}^{(k)}) - B(\boldsymbol{x}), \boldsymbol{x} - \boldsymbol{x}^{(k)}\rangle. \end{aligned} \tag{A.4}$$

By the Cauchy-Schwarz inequality, the elementary inequality $\|a\|\|b\| \le \frac{1}{2}(\|a\|^2 + \|b\|^2)$, and the cocoercivity of B, it holds that

$$\begin{aligned} &\langle B(\boldsymbol{x}^{(k)}) - B(\boldsymbol{x}), \boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k+1)}\rangle \\ &\quad \le \Big\|\sqrt{\frac{1}{2\beta}}\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k+1)}\Big\| \big\|\sqrt{2\beta}B(\boldsymbol{x}^{(k)}) - B(\boldsymbol{x})\big\| \\ &\quad \le \frac{1}{4\beta}\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k+1)}\|^2 + \beta\|B(\boldsymbol{x}^{(k)}) - B(\boldsymbol{x})\|^2 \\ &\quad \le \frac{1}{4\beta}\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k+1)}\|^2 + \langle B(\boldsymbol{x}^{(k)}) - B(\boldsymbol{x}), \boldsymbol{x}^{(k)} - \boldsymbol{x}\rangle. \end{aligned}$$

Then, by using this upperbound in (A.4), we have that

$$\begin{aligned} 0 &\le \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}, \boldsymbol{x}^{(k)}) - \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}, \boldsymbol{x}^{(k+1)}) - \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}^{(k+1)}, \boldsymbol{x}^{(k)}) \\ &\quad + \tfrac{1}{4\beta}\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k+1)}\|^2 \\ &\le \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}, \boldsymbol{x}^{(k)}) - \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}, \boldsymbol{x}^{(k+1)}) \\ &\quad - \tfrac{1}{2}\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\|^2_{\Xi\Gamma^{-1}} + \tfrac{1}{4\beta}\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k+1)}\|^2 \\ &= \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}, \boldsymbol{x}^{(k)}) - \operatorname{dist}_{\hat{\varphi}}(\boldsymbol{x}, \boldsymbol{x}^{(k+1)}) - \tfrac{1}{2}\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\|^2_{\Theta}, \end{aligned} \tag{A.5}$$

where $\Theta = \Xi\Gamma^{-1} - \frac{1}{2\beta}I$ and $\Xi = \operatorname{diag}(\{\xi_{\varphi_i} I_{n_i}\}_{i\in\mathcal{I}})$. The second inequality holds because $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}} \in \operatorname{int}\operatorname{dom}(\varphi) \cap \operatorname{dom}(A)$,

$\text{dist}_{\hat{\varphi}}(\boldsymbol{x},\boldsymbol{x}') = \sum_{i\in\mathcal{I}} \gamma_i^{-1}\text{dist}_{\varphi_i}(x_i,x_i')$ [3, Prop. 3.5], and, for each $i\in\mathcal{I}$, $\hat{\varphi}_i$ is 1-strongly convex on $\operatorname{int}\operatorname{dom}(\varphi_i)\cap\operatorname{dom}(A_i)$, thus implying $\text{dist}_{\hat{\varphi}}(\boldsymbol{x}^{(k+1)},\boldsymbol{x}^{(k)}) \geq \frac{1}{2}\|\boldsymbol{x}^{(k+1)}-\boldsymbol{x}^{(k)}\|^2_{\Xi\Gamma^{-1}}$.

By rearranging (A.5) and summming over k, we obtain that

$$\text{dist}_{\hat{\varphi}}(\boldsymbol{x},\boldsymbol{x}^{(k+1)})+\frac{1}{2}\sum_{\ell=0}^{k}\|\boldsymbol{x}^{(\ell+1)}-\boldsymbol{x}^{(\ell)}\|^2_{\Theta}\leq\text{dist}_{\hat{\varphi}}(\boldsymbol{x},\boldsymbol{x}^{(0)}). \tag{A.6}$$

Due to the step size rule, i.e., $\gamma_i\in(0,2\beta\xi_{\varphi_i})$, for all $i\in\mathcal{I}$, $\Theta = \Xi\Gamma^{-1}-\frac{1}{2\beta}I\succ 0$, implying that the second addend on the left-hand side is a sum of non-negative terms. Combining the fact that $\text{dist}_{\hat{\varphi}}(\boldsymbol{x},\boldsymbol{x}^{(k+1)}) \geq \frac{1}{2}\|\boldsymbol{x}^{(k+1)}-\boldsymbol{x}\|^2_{\Xi\Gamma^{-1}}$ and (A.6), we infer that $\|\boldsymbol{x}^{(k+1)}-\boldsymbol{x}\|^2_{\Xi\Gamma^{-1}} \leq \text{dist}_{\hat{\varphi}}(\boldsymbol{x},\boldsymbol{x}^{(0)}) < +\infty$, for all $k\in\mathbb{N}$, implying that the sequence $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}}$ is bounded.

Cluster point is a solution: Let $\hat{\boldsymbol{x}}$ be an arbitrary cluster point of the bounded sequence $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}}$. Next, we show that $\hat{\boldsymbol{x}}$ is contained in $\operatorname{zer}(A+B)\cap\operatorname{int}\operatorname{dom}(\varphi)$. We know that the sequence $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}}$ is contained in $\operatorname{int}\operatorname{dom}(\varphi)$. The inequality (A.6) implies that, for any fixed point $\hat{\boldsymbol{x}}\in\operatorname{zer}(A+B)\cap\operatorname{int}(\hat{\varphi})$, the sequence $(\text{dist}_{\hat{\varphi}}(\hat{\boldsymbol{x}},\boldsymbol{x}^{(k)}))_{k\in\mathbb{N}}$ is bounded. Due to this fact, essential smoothness of $\hat{\varphi}$, and the fact that $\hat{\boldsymbol{x}}\in\operatorname{int}\operatorname{dom}(\hat{\varphi})$, every cluster point belongs to $\operatorname{int}\operatorname{dom}(\hat{\varphi})$ [3, Thm. 3.8(ii)]. Since we assume that $\operatorname{zer}(A+B)\cap\operatorname{int}\operatorname{dom}(\varphi)\neq\varnothing$, it remains to show that $0\in(A+B)(\hat{\boldsymbol{x}})$. From the inclusion in (A.1), we have that

$$\begin{aligned}&\Gamma^{-1}\big(\nabla\varphi(\boldsymbol{x}^{(k)})-\nabla\varphi(\boldsymbol{x}^{(k+1)})\big)-B(\boldsymbol{x}^{(k)})\\&\quad+B(\boldsymbol{x}^{(k+1)})\in(A+B)(\boldsymbol{x}^{(k+1)}),\end{aligned} \tag{A.7}$$

for all $k\geq 0$. Since A and B are maximally monotone and $\operatorname{dom}(B)=\mathbb{R}^n$, $(A+B)$ is maximally monotone [5, Cor. 25.5]. Therefore, $\operatorname{gph}(A+B)$ is closed [5, Prop. 20.38]. Additionally, B is continuous and $\nabla\varphi(\cdot)$ is also continuous on $\operatorname{int}\operatorname{dom}(\varphi)$. Therefore, by taking the limit along a subsequence of $(\boldsymbol{x}^{(k)})$ that converges to $\hat{\boldsymbol{x}}$ in (A.7), the limit of the left-hand side term of the inclusion (A.7) is 0, and indeed $0\in(A+B)(\hat{\boldsymbol{x}})$.

Uniqueness of cluster point: Finally, we need to show that the cluster point set of $(\boldsymbol{x}^{(k)})_{k\in\mathbb{N}}$ is a singleton. First, for any arbitrary cluster point $\hat{\boldsymbol{x}}$, $\text{dist}_{\hat{\varphi}}(\hat{\boldsymbol{x}},\boldsymbol{x}^{(k)})$ converges due to the inequality (A.5) and [5, Lem. 5.31]. Then, we take two subsequences $\boldsymbol{x}^{(k_n)}$ and $\boldsymbol{x}^{(\ell_n)}$ and let their cluster points be denoted by $\hat{\boldsymbol{x}}_1$ and $\hat{\boldsymbol{x}}_2$. Therefore, the sequence $(\langle\hat{\boldsymbol{x}}_1-\hat{\boldsymbol{x}}_2,\nabla\hat{\varphi}(\boldsymbol{x}^{(k)})-\nabla\hat{\varphi}(\hat{\boldsymbol{x}}_2)\rangle)_{k\in\mathbb{N}}$, which is equal to $(\text{dist}_{\hat{\varphi}}(\hat{\boldsymbol{x}}_2,\boldsymbol{x}^{(k)})+\text{dist}_{\hat{\varphi}}(\hat{\boldsymbol{x}}_1,\hat{\boldsymbol{x}}_2)-\text{dist}_{\hat{\varphi}}(\hat{\boldsymbol{x}}_1,\boldsymbol{x}^{(k)}))_{k\in\mathbb{N}}$ by [4, Prop. 2.3.(ii)], converges. Taking the subsequence $\boldsymbol{x}^{(\ell_n)}$ and using the continuity of $\hat{\varphi}$, we obtain that $\eta=\lim_{\ell_n\to\infty}\langle\hat{\boldsymbol{x}}_1-\hat{\boldsymbol{x}}_2,\nabla\hat{\varphi}(\boldsymbol{x}^{(\ell_n)})-\nabla\hat{\varphi}(\hat{\boldsymbol{x}}_2)\rangle=\langle\hat{\boldsymbol{x}}_1-\hat{\boldsymbol{x}}_2,\nabla\hat{\varphi}(\hat{\boldsymbol{x}}_2)-\nabla\hat{\varphi}(\hat{\boldsymbol{x}}_2)\rangle=0$. By the continuity of $\hat{\varphi}$ and considering the subsequence $\boldsymbol{x}^{(k_n)}$, we also have that $\lim_{k_n\to\infty}\langle\hat{\boldsymbol{x}}_1-\hat{\boldsymbol{x}}_2,\nabla\hat{\varphi}(\boldsymbol{x}^{(k_n)})-\nabla\hat{\varphi}(\hat{\boldsymbol{x}}_2)\rangle=\eta$. Since $\eta=0$, we obtain that $\lim_{k_n\to\infty}\langle\hat{\boldsymbol{x}}_1-\hat{\boldsymbol{x}}_2,\nabla\hat{\varphi}(\boldsymbol{x}^{(k_n)})-\nabla\hat{\varphi}(\hat{\boldsymbol{x}}_2)\rangle=\langle\hat{\boldsymbol{x}}_1-\hat{\boldsymbol{x}}_2,\nabla\hat{\varphi}(\hat{\boldsymbol{x}}_1)-\nabla\hat{\varphi}(\hat{\boldsymbol{x}}_2)\rangle=0$. However, $\hat{\varphi}$ is essentially strictly convex. Hence, the preceding equality holds only if $\hat{\boldsymbol{x}}_1=\hat{\boldsymbol{x}}_2$, implying the uniqueness of the cluster point.

Appendix B. Proof of Proposition 1

The B-FB update in (8) applied to $A=\mathrm{N}_{\mathcal{X}}$ and $B=F$ can be rewritten as

$$\begin{aligned}&0\in\nabla\varphi(\boldsymbol{x}^{(k+1)})+\Gamma\mathrm{N}_{\mathcal{X}}(\boldsymbol{x}^{(k+1)})-(\nabla\varphi(\boldsymbol{x}^{(k)})-\Gamma(F(\boldsymbol{x}^{(k)}))\\&\Leftrightarrow 0\in\partial(\hat{\varphi}+\iota_{\mathcal{X}})(\boldsymbol{x}^{(k+1)})-\partial\langle(\nabla\hat{\varphi}(\boldsymbol{x}^{(k)})-(F(\boldsymbol{x}^{(k)})),\boldsymbol{x}^{(k+1)}\rangle,\end{aligned}$$

where $\iota_{\mathcal{X}}$ denotes the indicator function of $\mathcal{X}$ [5], Eq. (1.41), implying that

$$\begin{aligned}\boldsymbol{x}^{(k+1)}&=\operatorname*{argmin}_{\boldsymbol{y}\in\mathcal{X}}\Big(\varphi(\boldsymbol{y})-\langle\nabla\varphi(\boldsymbol{x}^{(k)}),\boldsymbol{y}\rangle+\langle F(\boldsymbol{x}^{(k)}),\boldsymbol{y}\rangle_{\Gamma}\Big)\\&=\operatorname*{argmin}_{\boldsymbol{y}\in\mathcal{X}}\Big(\text{dist}_{\varphi}(\boldsymbol{y},\boldsymbol{x}^{(k)})+\langle F(\boldsymbol{x}^{(k)}),\boldsymbol{y}\rangle_{\Gamma}\Big),\end{aligned} \tag{B.1}$$

since Γ is a diagonal matrix, φ is separable and strongly convex, $F(\boldsymbol{x})$ is a stacked of $\nabla_{x_i}J_i(x_i,\boldsymbol{x}_{\mathcal{N}_i})$, for all $i\in\mathcal{I}$, $\text{dist}_{\varphi}(\boldsymbol{x}',\boldsymbol{x})=\sum_{i\in\mathcal{I}}\text{dist}_{\varphi_i}(x_i',x_i)$ by [3, Prop. 3.5], and $\mathcal{X}=\prod_{i\in\mathcal{I}}\mathcal{X}_i$. The minimization in (B.1) is decomposable, i.e., for each $i\in\mathcal{I}$,

$$x_i^{(k+1)}=\operatorname*{argmin}_{y\in\mathcal{X}_i}\Big(\text{dist}_{\varphi_i}(y,x_i^{(k)})+\gamma_i\langle\nabla_{x_i}J_i(\boldsymbol{x}^{(k)}),y\rangle\Big),$$

which has a closed-form expression for φ_i defined in (10) and $\mathcal{X}_i$ defined in (3) as given by (11) [8, Ex. 4.2].

References

[1] E. Altman, T. Basar, T. Jimenez, N. Shimkin, Competitive routing in networks with polynomial costs, IEEE Trans. Autom. Control 47 (1) (2002) 92–96, doi:10.1109/9.981725.

[2] I. Atzeni, L.G. Ordóñez, G. Scutari, D.P. Palomar, J.R. Fonollosa, Demand-side management via distributed energy generation and storage optimization, IEEE Trans. Smart Grid 4 (2) (2013) 866–876.

[3] H.H. Bauschke, J.M. Borwein, Legendre functions and the method of random Bregman projections, J. Convex Anal. 4 (1) (1997) 27–67.

[4] H.H. Bauschke, J.M. Borwein, P.L. Combettes, Bregman monotone optimization algorithms, SIAM J. Control Optim. 42 (2) (2003) 596–636.

[5] H.H. Bauschke, P.L. Combettes, Convex Analysis and Monotone Operator Theory in Hilbert Spaces, Springer, 2011.

[6] A. Beck, M. Teboulle, Mirror descent and nonlinear projected subgradient methods for convex optimization, Oper. Res. Lett. 31 (3) (2003) 167–175.

[7] G. Belgioioso, W. Ananduta, S. Grammatico, C. Ocampo-Martinez, Operationally-safe peer-to-peer energy trading in distribution grids: a game-theoretic market-clearing mechanism, IEEE Trans. Smart Grid (2022) 1–11, doi:10.1109/TSG.2022.3158442.

[8] E.V. Belmega, P. Mertikopoulos, R. Negrel, L. Sanguinetti, Online convex optimization and no-regret learning: algorithms, guarantees and applications (2018). Available at https://arxiv.org/abs/1804.04529.

[9] M. Bianchi, S. Grammatico, Nash equilibrium seeking under partial-decision information over directed communication networks, in: the 59th IEEE Conference on Decision and Control (CDC), 2020, pp. 3555–3560, doi:10.1109/CDC42340.2020.9304267.

[10] M.N. Bùi, P.L. Combettes, Bregman forward-backward operator splitting, Set-Valued Variational Anal. 29 (3) (2021) 583–603.

[11] J. Duchi, S. Shalev-Shwartz, Y. Singer, T. Chandra, Efficient projections onto the l_1-ball for learning in high dimensions, in: Proceedings of the 25th International Conference on Machine Learning, 2008, pp. 272–279.

[12] F. Facchinei, J.-S. Pang, Finite-dimensional Variational Inequalities and Complementarity Problems, Springer Science & Business Media, 2007.

[13] A. Kannan, U.V. Shanbhag, Distributed computation of equilibria in monotone Nash games via iterative regularization techniques, SIAM J. Optim. 22 (4) (2012) 1177–1205.

[14] J. Koshal, A. Nedić, U.V. Shanbhag, Distributed algorithms for aggregative games on graphs, Operations Research 64 (3) (2016) 680–704.

[15] P. Mertikopoulos, Z. Zhou, Learning in games with continuous action sets and unknown payoff functions, Math. Programm. 173 (2019) 465–507.

[16] Y. Nesterov, Primal-dual subgradient methods for convex problems, Math. Programm. 120 (1) (2009) 221–259.

[17] A. Orda, R. Rom, N. Shimkin, Competitive routing in multiuser communication networks, IEEE/ACM Trans. Netw. 1 (5) (1993) 510–521, doi:10.1109/90.251910.

[18] Y. Pang, G. Hu, Distributed Nash equilibrium seeking with limited cost function knowledge via a consensus-based gradient-free method, IEEE Trans. Autom. Control 66 (4) (2020) 1832–1839.

[19] N. Quijano, C. Ocampo-Martinez, J. Barreiro-Gomez, G. Obando, A. Pantoja, E. Mojica-Nava, The role of population games and evolutionary dynamics in distributed control systems: the advantages of evolutionary game theory, IEEE Control Syst. Mag. 37 (1) (2017) 70–97.

[20] G. Scutari, S. Barbarossa, D.P. Palomar, Potential games: a framework for vector power control problems with coupled constraints, in: IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, 2006, pp. 241–244, doi:10.1109/ICASSP.2006.1660950.

[21] S. Shalev-Shwartz, Online learning and online convex optimization, Found. Trends Mach. Learn. 4 (2) (2011) 107–194.

[22] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, S. Boyd, OSQP: An operator splitting solver for quadratic programs, Math. Programm. Comput. 12 (4) (2020) 637–672.

[23] T. Tatarenko, W. Shi, A. Nedić, Geometric convergence of gradient play algorithms for distributed Nash equilibrium seeking, IEEE Trans. Autom. Control 66 (11) (2021) 5342–5353, doi:10.1109/TAC.2020.3046232.

[24] W. Tushar, C. Yuen, H. Mohsenian-Rad, T. Saha, H.V. Poor, K.L. Wood, Transforming energy networks via peer-to-peer energy trading: the potential of game-theoretic approaches, IEEE Signal Process. Mag. 35 (4) (2018) 90–111.

[25] P. Yi, L. Pavel, An operator splitting approach for distributed generalized Nash equilibria computation, Automatica 102 (2019) 111–121.