

## Semi-Decentralized Generalized Nash Equilibrium Seeking in Monotone Aggregative Games

Belgioioso, Giuseppe; Grammatico, Sergio

**DOI**

[10.1109/TAC.2021.3135360](https://doi.org/10.1109/TAC.2021.3135360)

**Publication date**

2023

**Document Version**

Final published version

**Published in**

IEEE Transactions on Automatic Control

**Citation (APA)**

Belgioioso, G., & Grammatico, S. (2023). Semi-Decentralized Generalized Nash Equilibrium Seeking in Monotone Aggregative Games. *IEEE Transactions on Automatic Control*, 68(1), 140-155.  
<https://doi.org/10.1109/TAC.2021.3135360>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.



***Green Open Access added to TU Delft Institutional Repository***

***'You share, we take care!' - Taverne project***

**<https://www.openaccess.nl/en/you-share-we-take-care>**

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

# Semi-Decentralized Generalized Nash Equilibrium Seeking in Monotone Aggregative Games

Giuseppe Belgioioso  and Sergio Grammatico 

**Abstract**—We address the generalized Nash equilibrium seeking problem for a population of agents playing aggregative games with affine coupling constraints. We focus on semi-decentralized communication architectures, where there is a central coordinator able to gather and broadcast signals of aggregative nature to the agents. By exploiting the framework of monotone operator theory and operator splitting, we first critically review the most relevant available algorithms and then design two novel schemes: 1) a single-layer, fixed-step algorithm with convergence guarantee for general (noncocoercive, nonstrictly) monotone aggregative games and 2) a single-layer proximal-type algorithm for a class of monotone aggregative games with linearly coupled cost functions. We also design novel accelerated variants of the algorithms via (alternating) inertial and over-relaxation steps. Finally, we show via numerical simulations that the proposed algorithms outperform those in the literature in terms of convergence speed.

**Index Terms**—Distributed algorithms, multi-agent systems, optimization methods, scalability.

## I. INTRODUCTION

### A. Aggregative Games

An aggregative game is a set of coupled optimization problems, each associated with an autonomous agent, i.e., an independent decision maker, where the cost function of each agent depends on some aggregate effect of all the agents in the population [1]–[3]. Namely, the aggregative feature arises whenever each agent is affected by the overall population behavior, hence, not by some specific agents. In general, such a special feature is typical of incentive-based control in competitive markets [4] and in fact engineering applications of aggregative games span from

Manuscript received 5 July 2021; accepted 3 December 2021. Date of publication 14 December 2021; date of current version 28 December 2022. This work was supported in part by the NWO under Research Projects OMEGA under Grant 613.001.702 and Grant P2P-TALES647.003.003 and in part by the ERC under Research Project COSMOS under Grant 802348. Recommended by Associate Editor Raphaël M. Jungers. (Corresponding author: Giuseppe Belgioioso.)

Giuseppe Belgioioso is with the Automatic Control Laboratory, Swiss Federal Institute of Technology (ETH), 8092 Zürich, Switzerland (e-mail: gbelgioioso@ethz.ch).

Sergio Grammatico is with the Delft Center for Systems and Control, TU Delft 2628CD, Delft, The Netherlands (e-mail: grammatico.css@gmail.com).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TAC.2021.3135360>.

Digital Object Identifier 10.1109/TAC.2021.3135360

demand side management in the smart grid [5], [6] and charging control for plug-in electric vehicles [7], [8] to spectrum sharing in wireless networks [9] and network congestion control [10]. With these motivating applications in mind, aggregative games have been receiving high research interest, within the operations research [11] and especially the automatic control [12]–[17] communities. Researchers have in fact studied and proposed solutions to the generalized Nash equilibrium problem (GNEP) in aggregative games, which is the problem to compute a set of decisions such that each is individually optimal given the others. Remarkably, the aggregative structure has been exploited to mitigate the computational complexity for large population size, and in fact the proposed solution algorithms are primarily non-centralized, i.e., almost (semi-) decentralized and distributed, where the computations by the agents are fully decoupled. Essentially, in semi-decentralized algorithms, the agents do not communicate with each other, but rely on a reliable central coordinator (e.g., an aggregator) that gathers the local decisions in aggregative form and then broadcasts (incentive) signals, e.g., dual variables, to all the agents [13]. On the other hand, in distributed algorithms, there is no central coordinator, so the agents communicate with each other to cooperatively estimate or reach consensus on the signals of common interest, e.g., dual and auxiliary variables. The latter algorithmic setup is also called partial-decision information [18], [19], because the agents do not have direct access to the aggregative effect on their cost functions, thus, they should estimate it via reliable, truthful peer-to-peer communications, e.g., via cooperative consensus protocols. These features motivate us to focus on the semi-decentralized algorithmic structure in this article.

### B. Literature Review

The literature on semi-decentralized GNE seeking in aggregative games is quite recent. Belgioioso and Grammatico [13] designed the first semi-decentralized GNE seeking algorithm for (nonstrictly/strongly, noncocoercive) *monotone* aggregative games<sup>1</sup>, where the algorithm derivation relies on the so-called forward-backward-forward (FBF) operator splitting. In parallel, for the class of *strongly monotone* games, Yi and Pavel [20], [21] proposed the first preconditioned forward-backward (pFB) operator splitting method, which is applicable to aggregative

<sup>1</sup>For ease of reading, with (strict/strongly) monotone game, we mean game with (strict/strongly) monotone pseudo-subdifferential mapping (§II-C).

games with semi-decentralized algorithmic structure—as shown in [22], the outcome of the pFB method for aggregative games is in fact the so-called asymmetric project algorithm (APA) [23, §12.5.1], proposed for aggregative games in [15]. Effectively, [13] and [20] are the first works to adopt an elegant and general mathematical approach based on monotone operator theory [24] to explicitly model GNEPs, to decouple the coupling constraints via Lagrangian duality, and in turn to exploit operator splitting methods for systematically designing (noncentralized) GNE seeking algorithms. Next, we refer to some other relevant GNE seeking algorithms for or applicable to aggregative games. For a class of *unconstrained strictly monotone* games, Ye and Hu [6] proposed continuous-time saddle-point dynamics. For *strictly monotone* games with *equality* coupling constraints, in Liang *et al.* [14] proposed continuous-time projected pseudo-gradient dynamics paired with discontinuous dynamics for dual and auxiliary variables. For *unconstrained, strongly monotone* aggregative games, Deng and Liang [16] proposed continuous-time, integral consensus-based dynamics. Recently, De Persis and Grammatico [17] proposed continuous-time, integral dynamics for a class of *strongly monotone* aggregative games.

From the literature on (semi-decentralized) GNE seeking in aggregative games, several critical issues emerge. First, the solution methods available for general (nonstrictly, noncocoercive) monotone aggregative games are limited to algorithms that require at least two demanding computations (projections) and two communications (between the agents and the coordinator) at each iteration, see, e.g., the FBF [13] and extra-gradient (EG) based methods [23, §12.6.1]; instead, computationally convenient algorithms, e.g., the pFB [21], require strong monotonicity of the game. Surprisingly, there is currently no single-communication-per-iteration, fixed-step algorithm for merely monotone aggregative games. For instance, the pFB method does not always converge in merely monotone games, not even under vanishing step sizes [25]. From a practical perspective, the available algorithms may require a large number of iterations, and in particular of communications between the agents and the central coordinator, to converge. For example, algorithms based on the iterative Tikhonov regularization (ITR) [26] require double-layer vanishing step sizes, which considerably slows down convergence. Finally, often, the local step sizes of the algorithms are based on global properties of the game data, that however should be unknown to the local agents in practice—on the contrary, little or no coordination among agents should be necessary to set the step sizes with guaranteed convergence.

### C. Contribution

In this article, we fully exploit monotone operator theory and operator splitting methodologies to study and address the main technical and computational issues that currently afflict (semi-decentralized) GNE seeking in aggregative games. Specifically, our main contributions are summarized in the following.

- 1) We characterize the available (semi-decentralized) algorithms with a general operator-theoretic perspective, which allows us to establish basically the most general convergence results and draw a fair technical comparison

among these algorithms (Section III), as well as to possibly improve convergence speed, e.g., via accelerations.

- 2) We present the first single-layer, single-communication-per-iteration, fixed-step algorithm for (nonstrictly, noncocoercive) monotone aggregative games (see Section IV-A)—essentially, the most desirable algorithmic features for the most general class of monotone aggregative games one could hope for.
- 3) We present a very fast, single-layer, single-communication-per-iteration, fixed-step, proximal algorithm for a class of (nonstrictly) monotone aggregative games with linearly coupled cost functions (see Section IV-B)—essentially, the most desirable algorithmic features with the fastest convergence ever experienced for a special, popular, class of monotone aggregative games.
- 4) We design an alternating inertial acceleration scheme, which is applicable to some algorithms (see Section IV-C) and that, remarkably, in some particular cases outperforms the classic inertial acceleration in terms of numerical convergence—mathematically, we prove that our alternating inertia preserves averagedness of operators, thus, the convergence is desirably Fejér monotone.

### D. Notation and Definitions

*Basic Notation:*  $\mathbb{R}$  denotes the set of real numbers, and  $\overline{\mathbb{R}} := \mathbb{R} \cup \{\infty\}$  the set of extended real numbers.  $\mathbf{0}$  ( $\mathbf{1}$ ) denotes a matrix/vector with all elements equal to 0 ( $\mathbf{1}$ ); to improve clarity, we may add the dimension of these matrices/vectors as subscript.  $A \otimes B$  denotes the Kronecker product between the matrices  $A$  and  $B$ . For a square matrix  $A \in \mathbb{R}^{n \times n}$ , its transpose is  $A^\top$ ,  $[A]_{i,j}$  represents the element on the row  $i$  and column  $j$ .  $A \succ 0$  ( $\succeq 0$ ) stands for positive definite (semidefinite) matrix. Given  $A \succ 0$ ,  $\|\cdot\|_A$  denotes the  $A$ -induced norm, such that  $\|x\|_A = x^\top A x$ .  $\|A\|$  denotes the largest singular value of  $A$ ;  $\text{eig}_{\max}(A)$  and  $\text{eig}_{\min}(A)$  denote, respectively, the largest and the smallest eigenvalues of  $A$ . Given  $N$  scalars,  $a_1, \dots, a_N$ ,  $\text{diag}(a_1, \dots, a_N)$  denotes the diagonal matrix with  $a_1, \dots, a_N$  on the main diagonal. Given  $N$  vectors  $x_1, \dots, x_N \in \mathbb{R}^n$ ,  $\mathbf{x} := \text{col}(x_1, \dots, x_N) = [x_1^\top, \dots, x_N^\top]^\top$ .

*Operator-Theoretic Definitions:*  $\text{Id}(\cdot)$  denotes the identity operator. The mapping  $\iota_S : \mathbb{R}^n \rightarrow \{0, \infty\}$  denotes the indicator function for the set  $S \subseteq \mathbb{R}^n$ , i.e.,  $\iota_S(x) = 0$  if  $x \in S$ ,  $\infty$  otherwise. For a closed set  $S \subseteq \mathbb{R}^n$ , the mapping  $\text{proj}_S : \mathbb{R}^n \rightarrow S$  denotes the projection onto  $S$ , i.e.,  $\text{proj}_S(x) = \text{argmin}_{y \in S} \|y - x\|$ . The set-valued mapping  $N_S : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$  denotes the normal cone operator for the set  $S \subseteq \mathbb{R}^n$ , i.e.,  $N_S(x) = \emptyset$  if  $x \notin S$ ,  $\{v \in \mathbb{R}^n \mid \sup_{z \in S} v^\top(z - x) \leq 0\}$  otherwise. For a function  $\psi : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$ ,  $\text{dom}(\psi) := \{x \in \mathbb{R}^n \mid \psi(x) < \infty\}$ ;  $\partial\psi : \text{dom}(\psi) \rightrightarrows \mathbb{R}^n$  denotes its subdifferential set-valued mapping, defined as  $\partial\psi(x) := \{v \in \mathbb{R}^n \mid \psi(z) \geq \psi(x) + v^\top(z - x) \text{ for all } z \in \text{dom}(\psi)\}$ ;  $\text{prox}_\psi(x) = \text{argmin}_{y \in \mathbb{R}^n} \psi(y) + \frac{1}{2}\|y - x\|^2$  denotes its proximal operator. A set-valued mapping  $\mathcal{F} : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$  is (strictly) monotone if  $(u - v)^\top(x - y) \geq (>) 0$  for all  $x \neq y \in \mathbb{R}^n$ ,  $u \in \mathcal{F}(x)$ ,  $v \in \mathcal{F}(y)$ ;  $\mathcal{F}$  is  $\eta$ -strongly

monotone, with  $\eta > 0$ , if  $(u - v)^\top(x - y) \geq \eta \|x - y\|^2$  for all  $x \neq y \in \mathbb{R}^n$ ,  $u \in \mathcal{F}(x)$ ,  $v \in \mathcal{F}(y)$ .  $J_{\mathcal{F}} := (\text{Id} + \mathcal{F})^{-1}$  denotes the resolvent operator of  $\mathcal{F}$ ;  $\text{fix}(\mathcal{F}) := \{x \in \mathbb{R}^n \mid x \in \mathcal{F}(x)\}$  and  $\text{zer}(\mathcal{F}) := \{x \in \mathbb{R}^n \mid 0 \in \mathcal{F}(x)\}$  denote the set of fixed points and of zeros, respectively.

## II. GENERALIZED NASH EQUILIBRIUM PROBLEM IN AGGREGATIVE GAMES

### A. Problem Statement

We consider a set of  $N$  agents, where each agent  $i \in \mathcal{I} := \{1, \dots, N\}$  shall choose its decision variable (i.e., strategy)  $x_i$  from the local decision set  $\Omega_i \subseteq \mathbb{R}^n$  with the aim of minimizing its local cost function  $J_i(x_i, \mathbf{x}_{-i})$ , which depends on the local variable  $x_i$  and on the decision variables of the other agents,  $\mathbf{x}_{-i} := \text{col}(\{x_j\}_{j \in \mathcal{I} \setminus \{i\}}) \in \mathbb{R}^{n(N-1)}$ .

In this article, we focus on the class of *aggregative games*, where the cost function of each agent depends on the local decision variable and on the value of the aggregation, i.e.,

$$\text{avg}(\mathbf{x}) := \frac{1}{N} \sum_{i=1}^N x_i. \quad (1)$$

Specifically, we consider local cost functions of the form

$$J_i(x_i, \mathbf{x}_{-i}) := g_i(x_i) + f_i(x_i, \text{avg}(\mathbf{x})) \quad (2)$$

where  $g_i$  and  $f_i$  satisfy the following assumptions.

*Assumption 1:* For each  $i \in \mathcal{I}$ , the function  $g_i$  is continuous (possibly nondifferentiable) and convex, and  $f_i(\cdot, \frac{1}{N} \cdot + y)$  is continuously differentiable and convex, for any  $y \in \mathbb{R}^n$ .  $\square$

Cost functions as in (2) are the most general considered in the literature of monotone games [27, Rem. 1], [28, § 12].

Furthermore, we consider *generalized games*, where the coupling among the agents arises not only via the cost functions, but also via their feasible decision sets. In our setup, the coupling constraints are described by an affine function,  $\mathbf{x} \mapsto A\mathbf{x} - b$ , where  $A := [A_1 | \dots | A_N] \in \mathbb{R}^{m \times nN}$ ,  $b := \sum_{i=1}^N b_i \in \mathbb{R}^m$ . Thus, the global feasible set reads as

$$\mathcal{X} = \left( \prod_{i \in \mathcal{I}} \Omega_i \right) \cap \{ \mathbf{x} \in \mathbb{R}^{nN} \mid A\mathbf{x} - b \leq \mathbf{0}_m \} \subseteq \mathbb{R}^{nN} \quad (3)$$

while the feasible decision set of each agent  $i \in \mathcal{I}$  is characterized by the set-valued mapping  $\mathcal{X}_i$ , defined as

$$\mathcal{X}_i(\mathbf{x}_{-i}) := \left\{ y_i \in \Omega_i \mid A_i y_i \leq b_i + \sum_{j \neq i} (b_j - A_j x_j) \right\}$$

where  $A_i \in \mathbb{R}^{m \times n}$  and  $b_i$  are local parameters that define how agent  $i$  is involved in the coupling constraints.

*Remark 1 (Affine constraints):* Affine coupling constraints, as considered in (3), are the most common in the literature of monotone games, see for example [12], [14], [15], [21]. For the sake of compactness, we did not include coupling equality constraints in (3). However, all the results in the remainder of the paper can be straightforwardly adapted to cover this case.  $\square$

Next, let us formalize standard convexity and closedness assumptions for the constraint sets.

*Assumption 2:* For each  $i \in \mathcal{I}$ , the local set  $\Omega_i \subseteq \mathbb{R}^n$  is nonempty, closed and convex. Moreover, the global set  $\mathcal{X}$  satisfies Slater's constraint qualification.  $\square$

In summary, the aim of each agent  $i$ , given the aggregate decision  $\text{avg}(\mathbf{x})$ , is to choose a strategy,  $x_i^*$ , that solves its local convex optimization problem according to the game setup previously described, i.e., for all  $i \in \mathcal{I}$

$$\begin{cases} \underset{x_i \in \mathbb{R}^n}{\text{argmin}} & J_i(x_i, \mathbf{x}_{-i}) = g_i(x_i) + f_i(x_i, \text{avg}(\mathbf{x})) \\ \text{s.t.} & x_i \in \Omega_i \\ & A_i x_i \leq b_i + \sum_{j \neq i} (b_j - A_j x_j) \end{cases} \quad (4)$$

where the last constraint is equivalent to  $A\mathbf{x} - b \leq \mathbf{0}$ . From a game-theoretic perspective, we consider the problem to compute a Nash equilibrium [29], as formalized next.

*Definition 1 (Generalized  $\varepsilon$ -Nash equilibrium):* A collective strategy  $\mathbf{x}^* \in \mathcal{X}$  is a generalized  $\varepsilon$ -Nash equilibrium ( $\varepsilon$ -GNE) of the game in (4) if, for all  $i \in \mathcal{I}$

$$J_i(x_i^*, \mathbf{x}_{-i}^*) \leq \inf \{ J_i(y, \mathbf{x}_{-i}^*) + \varepsilon \mid y \in \mathcal{X}_i(\mathbf{x}_{-i}^*) \}. \quad (5)$$

If (5) holds with  $\varepsilon = 0$ , then  $\mathbf{x}^*$  is a GNE.  $\square$

In other words, a set of strategies is a Nash equilibrium if no agent can improve its objective function by unilaterally changing its strategy to another feasible one.

*Remark 2 (Existence of a GNE):* If Assumption 2 holds with bounded local strategy sets  $\Omega_i$ 's, the existence of a GNE follows from Brouwer's fixed-point theorem [28, Prop. 12.7], while uniqueness does not hold in general.  $\square$

### B. Nash Versus Aggregative (or Wardrop) Equilibria

In aggregative games with cost functions as in (2), the condition in (5) specializes as: for all  $i \in \mathcal{I}$  and  $y \in \mathcal{X}_i(\mathbf{x}_{-i}^*)$

$$g_i(x_i^*) + f_i(x_i^*, \text{avg}(\mathbf{x}^*)) \leq g_i(y) + f_i\left(y, \frac{1}{N}y + \frac{1}{N} \sum_{j \neq i} x_j^*\right)$$

where the decision variable of agent  $i$ , i.e.,  $x_i^*$ , appears also in the second argument of  $f_i$ , since  $x_i^*$  contributes to form the average strategy, i.e.,  $\text{avg}(\mathbf{x}^*) = \frac{1}{N}x_i^* + \frac{1}{N} \sum_{j \neq i} x_j^*$ .

The concept of *aggregative (or Wardrop) equilibrium* (formalized in Definition 2) springs from the intuition that the contribution of each agent to the average strategy decreases as the population size grows. Technically, the influence of the decision variable of agent  $i$  on the second argument of its cost function  $f_i$  vanishes as  $N$  grows unbounded.

*Definition 2 (Generalized aggregative equilibrium):* A collective strategy  $\mathbf{x}^* \in \mathcal{X}$  is a generalized aggregative equilibrium (GAE) of the game in (4) if, for all  $i \in \mathcal{I}$

$$\begin{aligned} g_i(x_i^*) + f_i(x_i^*, \text{avg}(\mathbf{x}^*)) &\leq \\ \inf \{ g_i(y) + f_i(y, \text{avg}(\mathbf{x}^*)) \mid y \in \mathcal{X}_i(\mathbf{x}_{-i}^*) \}. & \end{aligned} \quad \square$$

We note that Nash and aggregative equilibria are strictly connected. In fact, under some mild assumptions, it can be proven that every GAE equilibrium is an  $\varepsilon$ -GNE equilibrium,

with  $\varepsilon$  vanishing as  $N$  diverges [15, §4]. Thus, in large-scale games where the agents are unaware of the population size, e.g., [30], a GAE represents a good approximation of a GNE.

### C. Variational Equilibria

In this article, we focus on the subclass of *variational GNE* (v-GNE) that corresponds to the solution set of an appropriate generalized variational inequality, i.e.,  $\text{GVI}(P, \mathcal{X})$ , namely, the problem of finding  $\mathbf{x}^* \in \mathcal{X}$  such that

$$\langle \mathbf{z}^*, \mathbf{x} - \mathbf{x}^* \rangle \geq 0, \quad \forall \mathbf{x} \in \mathcal{X}, \mathbf{z}^* \in P(\mathbf{x}^*)$$

where the mapping  $P : \mathbb{R}^{nN} \rightrightarrows \mathbb{R}^{nN}$  denotes the so-called *pseudo-subdifferential* (PS) of the game in (4), defined as

$$P(\mathbf{x}) := \prod_{i=1}^N \partial_{x_i} J_i(x_i, \mathbf{x}_{-i}). \quad (6)$$

Namely, the mapping  $P$  is obtained by stacking together the subdifferentials of the agents' cost functions with respect to their local decision variables. Given the splitting structure of the cost functions in (2), it follows by invoking [24, Cor. 16.48 (iii)] componentwise that the PS can be written as the sum of a set-valued mapping and a single-valued one:

$$P = G + F$$

where

$$G(\mathbf{x}) := \prod_{i=1}^N \partial g_i(x_i) \quad (7)$$

$$F(\mathbf{x}) := \text{col} \left( \left\{ \nabla_{x_i} f_i(x_i, \text{avg}(\mathbf{x})) \right\}_{i=1}^N \right). \quad (8)$$

Note that, since the local decision variable  $x_i$  of agent  $i$  enters also in the second argument of the cost function  $f_i(\cdot, \frac{1}{N} \cdot + \frac{1}{N} \sum_{j \neq i} x_j)$ , with Leibniz notation, we have that

$$\begin{aligned} \nabla_{x_i} f_i(x_i, \text{avg}(\mathbf{x})) \\ = \left( \nabla_{x_i} f_i(x_i, z) + \frac{1}{N} \nabla_z f(x_i, z) \right) \Big|_{z=\text{avg}(\mathbf{x})}. \end{aligned} \quad (9)$$

In the remainder of the article, let us refer to  $F$  as *pseudo-gradient* mapping (with a little abuse of terminology).

Under Assumptions 1 and 2, it follows by [28, Prop. 12.4] that any solution to  $\text{GVI}(P, \mathcal{X})$  is a (variational) Nash equilibrium of the game in (4). The inverse implication is not true in general, and actually in passing from the Nash equilibrium problem to the GVI problem most solutions are lost [28, § 12.2.2]; indeed, a game may have a Nash equilibrium while the corresponding GVI has no solution. Note that, if  $J_i$  in (2) is continuously differentiable for all  $i \in \mathcal{I}$ , then  $P$  is a single-valued mapping and  $\text{GVI}(P, \mathcal{X})$  reduces to  $\text{VI}(P, \mathcal{X})$ , which is commonly addressed in the context of game theory via projected pseudo-gradient algorithms, e.g., [11], [15], [22].

Next, we assume monotonicity of the PS mapping  $P$ , which “is one of the weakest conditions under which global convergence can be proved” for VI-type methods [29, § 5.2].

*Assumption 3 (Monotone and Lipschitz pseudo-gradient):* The mapping  $F$  in (8) is maximally monotone and  $\ell$ -Lipschitz continuous over  $\Omega := \prod_{i \in \mathcal{I}} \Omega_i$ , for some  $\ell > 0$ .  $\square$

It directly follows that also the PS  $P$  is maximally monotone since it is the sum of two maximally monotone operators [24,

Cor. 25.5], i.e.,  $P = G + F$ , where  $G$  is maximally monotone as concatenation of maximally monotone operators [24, Prop. 20.23] (i.e., the subdifferentials of the continuous and convex functions  $g_i$ 's [24, Th. 20.25]), and  $F$  is maximally monotone by Assumption 3.

The following lemma recalls some sufficient conditions for the existence and uniqueness of a variational GNE (v-GNE).

*Lemma 1 (Existence and Uniqueness of v-GNE):* Let Assumption 2 be satisfied. The following conditions hold.

- (i) If  $\Omega_i$  is bounded, for all  $i \in \mathcal{I}$ , and  $P$  is (strictly) monotone, then there exists a (unique) solution to  $\text{GVI}(P, \mathcal{X})$ .
- (ii) If  $P$  is strongly monotone, then there exists a unique solution to  $\text{GVI}(P, \mathcal{X})$ .  $\square$

*Proof:* (i) [24, Prop. 23.36]; (ii) [24, Cor. 23.37].  $\blacksquare$

Hereafter, we assume that a v-GNE of the game in (4) exists.

*Assumption 4 (Existence of a v-GNE):* The set of solutions to  $\text{GVI}(P, \mathcal{X})$  is nonempty.  $\square$

*Remark 3 (Approximate pseudo-gradient):* Let all cost functions  $(f_i)_{i \in \mathcal{I}}$  be uniformly bounded (on their respective feasible sets) for all population sizes,  $N$ . As the latter grows, the second term in the right hand side of (9) vanishes. In fact, if  $\lim_{N \rightarrow \infty} \text{avg}(\mathbf{x}) < \infty$ , we have that

$$\lim_{N \rightarrow \infty} \nabla_{x_i} f_i(x_i, \text{avg}(\mathbf{x})) = \nabla_{x_i} f_i(x_i, z) \Big|_{z=\lim_{N \rightarrow \infty} \text{avg}(\mathbf{x})}. \quad (10)$$

Thus, let us define an approximate version of the PG in (8) for large-scale games, i.e.,

$$\tilde{F}(\mathbf{x}) := \text{col} \left( \left\{ \nabla_{x_i} f_i(x_i, z) \Big|_{z=\text{avg}(\mathbf{x})} \right\}_{i=1}^N \right) \quad (11)$$

and the correspondent approximate PS, i.e.,

$$\tilde{P} := G + \tilde{F}. \quad (12)$$

As for v-GNE, one can show that any solution to  $\text{GVI}(\tilde{P}, \mathcal{X})$  is a (variational) GAE (v-GAE) of the game in (4) [17].  $\square$

### D. Nash Equilibria as Zeros of a Monotone Operator

In this section, we exploit operator theory to recast the Nash equilibrium problem into a monotone inclusion, namely, the problem of finding a zero of a set-valued monotone operator. As first step, we characterize a GNE of the game in terms of KKT conditions of the interdependent optimization problems in (4). For each agent  $i \in \mathcal{N}$ , let us introduce the Lagrangian function  $L_i$ , defined as

$$L_i(\mathbf{x}, \lambda_i) := J_i(x_i, \mathbf{x}_{-i}) + \iota_{\Omega_i}(x_i) + \lambda_i^\top (A\mathbf{x} - b)$$

where  $\lambda_i \in \mathbb{R}_{\geq 0}^m$  is the Lagrangian multiplier associated with the coupling constraints. It follows from [28, §12.2.3] that the set of strategies  $\mathbf{x}^*$ , where the Mangasarian–Fromovitz constraint qualification holds at  $x_i^*$  for the set  $\mathcal{X}_i(\mathbf{x}_{-i}^*)$ , for all  $i \in \mathcal{I}$ , is a GNE of the game in (4) if and only if there exist some dual variables  $\lambda_1^*, \dots, \lambda_N^* \in \mathbb{R}_{\geq 0}^m$  such that the following coupled KKT conditions are satisfied:

$$\forall i \in \mathcal{I} : \begin{cases} 0 \in \partial_{x_i} J_i(x_i^*, \mathbf{x}_{-i}^*) + N_{\Omega_i}(x_i^*) + A_i^\top \lambda_i^* \\ 0 \leq \lambda_i^* \perp -(A\mathbf{x}^* - b) \geq 0. \end{cases} \quad (13)$$

Similarly, we characterize a v-GNE in terms of KKT conditions by exploiting the Lagrangian duality scheme for the corresponding GVI problem, see [31, §3.2]. Specifically, if  $\mathcal{X}$  satisfies the Slater's condition (Assumption 1), it follows by [31, Th. 3.1] that  $\mathbf{x}^*$  is a solution to GVI( $\mathcal{X}, P$ ) if and only if there exists a dual variable  $\lambda^* \in \mathbb{R}_{\geq 0}^m$  such that

$$\begin{cases} 0 \in \partial_{x_i} J_i(x_i^*, \mathbf{x}_{-i}^*) + N_{\Omega_i}(x_i^*) + A_i^\top \lambda^*, \forall i \in \mathcal{I} \\ 0 \leq \lambda^* \perp -(A\mathbf{x}^* - b) \geq 0. \end{cases} \quad (14)$$

To cast (14) in compact form, we introduce the set-valued mapping  $T : \Omega \times \mathbb{R}_{\geq 0}^m \rightrightarrows \mathbb{R}^{nN} \times \mathbb{R}^m$ , defined as

$$T : \begin{bmatrix} \mathbf{x} \\ \lambda \end{bmatrix} \mapsto \begin{bmatrix} N_{\Omega}(\mathbf{x}) + P(\mathbf{x}) + A^\top \lambda \\ N_{\mathbb{R}_{\geq 0}^m}(\lambda) - (A\mathbf{x} - b) \end{bmatrix}. \quad (15)$$

The role of the mapping  $T$  in (15) is that its zeros correspond to the v-GNE of the game in (4), or, equivalently, to the solutions to the KKT system in (13) with equal dual variables, i.e.,  $\lambda_i = \lambda^*$  for all  $i \in \mathcal{I}$ , as formalized in the following statement.

*Proposition 1:* Let Assumptions 1, 2 hold. Then, the following statements are equivalent:

- (i)  $\mathbf{x}^*$  is a v-GNE of the game in (4);
- (ii)  $\exists \lambda^* \in \mathbb{R}_{\geq 0}^m$  such that, the pair  $(x_i^*, \lambda^*)$  is a solution to the KKT in (13), for all  $i \in \mathcal{I}$ ;
- (iii)  $\mathbf{x}^*$  is a solution to GVI( $P, \mathcal{X}$ ); and
- (iv)  $\exists \lambda^* \in \mathbb{R}_{\geq 0}^m$  such that  $\text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(T)$ .  $\square$

*Proof:* The equivalence (i) $\Leftrightarrow$ (iii) is proven in [28, Prop. 12.4]. (iii) $\Leftrightarrow$ (iv) follows by [31, Th. 3.1]. (iv) $\Leftrightarrow$ (ii) follows by noting that (13), with  $\lambda_1^* = \dots = \lambda_N^* = \lambda^*$ , is equivalent to (14), whose solutions corresponds to the zeros of  $T$  [31, §3.2].

A similar equivalence can be derived for v-GAE.  $\blacksquare$

*Proposition 2:* Let Assumptions 1, 2 hold. Then, the following statements are equivalent:

- (i)  $\mathbf{x}^*$  is a v-GAE of the game in (4);
- (ii)  $\exists \lambda^* \in \mathbb{R}_{\geq 0}^m$  such that, the pair  $(x_i^*, \lambda^*)$  is a solution to the KKT in (13) with  $\partial_{x_i} J_i(x_i^*, \mathbf{x}_{-i}^*)$  replaced by  $\partial g_i(x_i^*) + \nabla_{x_i} f_i(x_i^*, z)|_{z=\text{avg}(\mathbf{x}^*)}$ , for all  $i \in \mathcal{I}$ ;
- (iii)  $\mathbf{x}^*$  is a solution to GVI( $\tilde{P}, \mathcal{X}$ );
- (iv)  $\exists \lambda^* \in \mathbb{R}_{\geq 0}^m$  such that  $\text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(\tilde{T})$ , where  $\tilde{T}$  is analogous to  $T$  in (15) with  $P$  replaced by its approximation  $\tilde{P}$  in (12).  $\square$

*Proof:* The proof is similar to that of Proposition 1.  $\blacksquare$

### III. GENERALIZED NASH EQUILIBRIUM SEEKING: OPERATOR-THEORETIC CHARACTERIZATION

#### A. Zero Finding Methods for GNE Seeking

In Section II-D, we show that the original GNE seeking problem corresponds to the following generalized equation:

$$\text{find } \omega^* := \text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(T). \quad (16)$$

Next, we show that the mapping  $T$  can be written as the sum of two operators, i.e.,  $T = T_1 + T_2$ , where

$$T_1 : \omega \mapsto \text{col}(F(\mathbf{x}), b) \quad (17)$$

$$T_2 : \omega \mapsto (N_{\Omega}(\mathbf{x}) + G(\mathbf{x})) \times N_{\mathbb{R}_{\geq 0}^m}(\lambda) + S\omega \quad (18)$$

and  $S$  is a skew symmetric matrix, i.e.,  $S^\top = -S$ , defined as

$$S := \begin{bmatrix} 0 & A^\top \\ -A & 0 \end{bmatrix}. \quad (19)$$

The formulation  $T = T_1 + T_2$  is called *splitting* of  $T$ , and we exploit it in different ways later on. We show next that the mappings  $T_1$  and  $T_2$  are both maximally monotone, which paves the way for operator splitting algorithms [24, § 26].

*Lemma 2:* Let Assumptions 1–3 hold. The mappings  $T_1$  in (17),  $T_2$  in (18) and  $T$  in (15) are maximally monotone.  $\square$

*Proof:*  $T_1$  is maximally monotone since  $F$  is such by Assumption 3,  $b$  is a constant, thus maximally monotone, and the concatenation of maximally monotone operator remains maximally monotone [24, Prop. 20.23]. The first term of  $T_2$ , i.e.,  $(N_{\Omega} + G) \times N_{\mathbb{R}_{\geq 0}^m}$ , is maximally monotone, since normal cones of closed convex sets are maximally monotone and the concatenation preserves maximality [24, Prop. 20.23]; the second term, i.e.,  $S$ , is linear and skew symmetric, i.e.,  $S^\top = -S$ , thus maximally monotone [24, Ex. 20.35]. Then, the sum of the previous terms, namely,  $T_2$ , is maximally monotone by [24, Cor. 25.5], since  $\text{dom } S = \mathbb{R}^{nN+m}$ . Equivalently, the maximal monotonicity of  $T = T_1 + T_2$  follows from [24, Cor. 25.5], since  $\text{dom } T_1 = \mathbb{R}^{nN+m}$ .  $\blacksquare$

In the remainder of this section, we characterize the main features and limitations of some existing semi-decentralized algorithms for aggregative games with coupling constraints from a general operator-theoretic perspective.

*Remark 4 (Generalized aggregative equilibrium seeking):* In light of Proposition 2, the same operator-theoretic approach can be exploited to recast the GAE seeking problem as a monotone inclusion problem. It follows that all the GNE seeking algorithms introduced next can be adopted for seeking a GAE. Specifically, for gradient-based algorithms, it is sufficient to replace  $\nabla_{x_i} f_i(x_i, \text{avg}(\mathbf{x}))$  in (8) with its approximate version, i.e.,  $\nabla_{x_i} f_i(x_i, z)|_{z=\text{avg}(\mathbf{x})}$ .  $\square$

#### B. Preconditioned Forward-Backward Algorithm

The main idea of the preconditioned forward-backward algorithm (pFB, Algorithm 1) is that the zeros of the mapping  $T$  in (15) correspond to the fixed points of a certain operator, which depends on the chosen splitting (17)–(18) [24, §26.5] and on an arbitrary symmetric, positive definite matrix  $\Phi$ , known as preconditioning matrix [20]. The pFB method, proposed in [20] for strongly monotone games, is applicable to aggregative games with semi-decentralized algorithmic structure [22], in which case it reduces to the APA [23, §12.5.1], also proposed in [15]. A critical assumption for the convergence of this method is the cocoercivity of the pseudo-gradient mapping  $F$  in (8).

*Assumption 5 (Cocoercive pseudo-gradient):* The mapping  $F$  in (8) is  $\gamma$ -cocoercive on  $\Omega$ , for some  $\gamma > 0$ .  $\square$

*Remark 5 (Sufficient conditions for cocoercivity of  $F$ ):* If  $F$  is  $\mu$ -strongly monotone and  $\ell$ -Lipschitz,  $\ell \geq \mu > 0$ , then  $F$  is  $(\mu/\ell^2)$ -cocoercive. On the contrary, cocoercive mappings are not necessarily strongly monotone, e.g., the gradient of a nonstrictly convex smooth function. Some sufficient conditions

**Algorithm 1: Preconditioned forward-backward (pFB).**

**Initialization:**  $\delta > \frac{1}{2\gamma}; \forall i \in \mathcal{I}, x_i^0 \in \mathbb{R}^n, 0 < \alpha_i \leq (\|A_i\| + \delta)^{-1}; \lambda^0 \in \mathbb{R}_{\geq 0}^m, 0 < \beta \leq \left(\frac{1}{N} \sum_{i=1}^N \|A_i\| + \frac{1}{N} \delta\right)^{-1}$ .

**Iterate until convergence:**

1. Local: Strategy update, for all  $i \in \mathcal{I}$ :
 
$$y_i^k = x_i^k - \alpha_i (\nabla_{x_i} f_i(x_i^k, \text{avg}(\mathbf{x}^k)) + A_i^\top \lambda^k)$$

$$x_i^{k+1} = \text{prox}_{\alpha_i g_i + \iota_{\Omega_i}}(y_i^k)$$

$$d_i^{k+1} = 2A_i x_i^{k+1} - A_i x_i^k - b_i$$
2. Central coordinator: dual variable update
 
$$\lambda^{k+1} = \text{proj}_{\mathbb{R}_{\geq 0}^m}(\lambda^k + \beta \text{avg}(\mathbf{d}^{k+1}))$$

for cocoercivity of  $F$  based on the local cost functions  $(f_i)_{i \in \mathcal{I}}$  are provided in Appendix A1.  $\square$

*Remark 6:* (i) The local auxiliary variables  $y_i$ 's and  $d_i$ 's are introduced to cast Algorithm 1 in a more compact form. The quantity  $\text{avg}(\mathbf{d}^{k+1}) := \frac{1}{N} \sum_{i=1}^N (2A_i x_i^{k+1} - A_i x_i^k - b_i)$  measures the violation of the coupling constraints, technically, it is the ‘‘reflected violation’’ of the constraints at iteration  $k$ .

(ii) The proximal operator in Algorithm 1 reads as

$$\text{prox}_{\alpha_i g_i + \iota_{\Omega_i}}(y) = \begin{cases} \underset{z \in \mathbb{R}^n}{\text{argmin}} & g_i(z) + \frac{1}{2\alpha_i} \|z - y\|^2 \\ \text{s.t.} & z \in \Omega_i. \end{cases}$$

If  $g_i = 0$ , then the primal update in Algorithm 1 becomes a projection, i.e.,  $\text{prox}_{\alpha_i g_i + \iota_{\Omega_i}} = \text{proj}_{\Omega_i}$ .  $\square$

If Assumption 5 holds and the step sizes  $\{\alpha_i\}_{i \in \mathcal{I}}$  and  $\beta$  are small enough, then the sequence  $(\text{col}(\mathbf{x}^k, \lambda^k))_{k \in \mathbb{N}}$  generated by Algorithm 1 converges to some  $\text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(T)$ , where  $\mathbf{x}^*$  is a v-GNE, see [22, Th. 1] for a formal proof of convergence.

Algorithm 1 is semi-decentralized. In fact, at each iteration  $k$ , a central coordinator is needed to:

- (i) gather and broadcast the average strategy  $\text{avg}(\mathbf{x}^k)$ ;
- (ii) gather the reflected violation of the constraints  $\text{avg}(\mathbf{d}^k)$ ;
- (iii) update and broadcast the dual variable  $\lambda^k$ .

Specifically, after each central and local update in Algorithm 1, a communication stage follows. The central coordinator broadcasts to all the agents the current values of the aggregation  $\text{avg}(\mathbf{x}^k)$  and the multiplier vector  $\lambda^k$ . In return, each agent  $i \in \mathcal{I}$  updates its own strategy  $x_i$ , based on the received signals, and forwards it to the central coordinator. Moreover, at each iteration only two vectors, in  $\mathbb{R}^n$  and  $\mathbb{R}^m$ , respectively, are broadcast, independently on the population size  $N$ . Each decentralized computation consists of solving a finite-dimensional convex optimization problem, for which efficient algorithms are available.

*Remark 7:* The primal-dual iterations of Algorithm 1 are sequential, namely, while the local primal updates  $x_i^{k+1}$  can be performed in parallel, the dual update,  $\lambda^{k+1}$ , exploits the most recent value of the agents' strategies,  $d_i^{k+1}$ . This feature is convenient since it follows the natural information flow in the considered semi-decentralized communication structure.  $\square$

*Algorithm 1 as a fixed-point iteration:* The dynamics generated by Algorithm 1 can be cast in a compact form as the

fixed-point iteration

$$\omega^{k+1} = R_{\text{FB}}(\omega^k) \quad (20)$$

where  $\omega^k = \text{col}(\mathbf{x}^k, \lambda^k)$  is the vector of the primal-dual variables and  $R_{\text{FB}}$  is the so-called FB operator [24, (26.7)]

$$R_{\text{FB}} := (\text{Id} + \Phi^{-1}T_2)^{-1} \circ (\text{Id} - \Phi^{-1}T_1) \quad (21)$$

where  $T_1$  and  $T_2$  as in (17) and (18) and  $\Phi$  is a preconditioning matrix, here defined as

$$\Phi := \begin{bmatrix} \bar{\alpha}^{-1} \otimes I_n & -A^\top \\ -A & N\beta^{-1}I_m \end{bmatrix} \quad (22)$$

with  $\bar{\alpha} = \text{diag}(\alpha_1, \dots, \alpha_N)$ . When the mapping  $T_1$  is cocoercive (see Assumption 5),  $T_2$  is maximally monotone (see Lemma 2) and the step sizes in the main diagonal of  $\Phi$  are set as in Algorithm 1, then the preconditioned mappings  $\Phi^{-1}T_1$  and  $\Phi^{-1}T_2$  satisfy the following properties with respect to the  $\Phi$ -induced norm ([21, Lemma 7]):

- (i)  $\Phi^{-1}T_1$  is  $\gamma\delta$ -cocoercive w.r.t.  $\|\cdot\|_\Phi$ ;
- (ii)  $\Phi^{-1}T_2$  is maximally monotone w.r.t.  $\|\cdot\|_\Phi$ .

It follows from [24, Prop. 26.1(iv)–(d)] that the FB operator  $R_{\text{FB}}$  in (21) is averaged with respect to the same norm, i.e.,

- iii)  $R_{\text{FB}}$  is  $(\frac{2\delta\gamma}{4\delta\gamma-1})$ -averaged w.r.t.  $\|\cdot\|_\Phi$ .

Hence, the Banach–Picard fixed-point iteration in (20) converges to some  $\omega^* := \text{col}(\mathbf{x}^*, \lambda^*) \in \text{fix}(R_{\text{FB}})$  [24, Prop. 5.16], where  $\text{fix}(R_{\text{FB}}) = \text{zer}(T)$  [24, Prop. 26.1(iv)–(a)],  $\text{zer}(T) \neq \emptyset$  (Assumption 4) and, therefore,  $\mathbf{x}^*$  is a v-GNE by Prop. 1. We refer to [21], [22] for a complete convergence analysis.

*Inertial pFB Algorithm:* To conclude this section, we recall the inertial version of the pFB (see Algorithm 1), originally proposed for the more general context of generalized network games in [21, Algorithm 2] and summarized here in Algorithm 1B.

We note that the inertial extrapolation phase, at the end of the local and central updates, improves the converge properties of the pFB algorithm. The convergence of Algorithm 1B can be studied via fixed-point theory [32], or by relying on the inertial version of the FB splitting method [33]. We refer to [21, Th. 2] for a complete convergence proof of this algorithm.

**Algorithm 1B: Inertial pFB (I-pFB).**

**Initialization:**  $\theta \in [0, 1/3)$  and  $\delta > \frac{(1-\theta)^2}{2\gamma(1-3\theta)}$ , with  $\gamma$  as in Assumption 5; for all  $i \in \mathcal{I}, x_i^0 = \tilde{x}_i^0 \in \mathbb{R}^n, 0 < \alpha_i \leq (\|A_i\| + \delta)^{-1}; \lambda^0 = \tilde{\lambda}^0 \in \mathbb{R}_{\geq 0}^m, 0 < \beta \leq \left(\frac{1}{N} \sum_{i=1}^N \|A_i\| + \frac{1}{N} \delta\right)^{-1}$ .

**Iterate until convergence:**

1. Local: Strategy update, for all  $i \in \mathcal{I}$ :
 
$$y_i^k = \tilde{x}_i^k - \alpha_i (\nabla_{x_i} f_i(\tilde{x}_i^k, \text{avg}(\tilde{\mathbf{x}}^k)) + A_i^\top \tilde{\lambda}^k)$$

$$x_i^{k+1} = \text{prox}_{\alpha_i g_i + \iota_{\Omega_i}}(y_i^k)$$

$$\tilde{x}_i^{k+1} = x_i^{k+1} + \theta(x_i^{k+1} - x_i^k)$$

$$d_i^{k+1} = 2A_i x_i^{k+1} - A_i \tilde{x}_i^k - b_i$$
2. Central coordinator: Dual variable update:
 
$$\lambda^{k+1} = \text{proj}_{\mathbb{R}_{\geq 0}^m}(\tilde{\lambda}^k + \beta \text{avg}(\mathbf{d}^{k+1}))$$

$$\tilde{\lambda}^{k+1} = \lambda^{k+1} + \theta(\lambda^{k+1} - \lambda^k)$$



*Algorithm 1B as a Fixed-Point Iteration:* The dynamics generated by Algorithm 1B can be cast in a compact form as the following inertial fixed-point iteration:

$$\tilde{\omega}^k = \omega^k + \theta(\omega^k - \omega^{k-1}) \quad (23a)$$

$$\omega^{k+1} = R_{\text{FB}}(\tilde{\omega}^k) \quad (23b)$$

where  $\omega^k = \text{col}(\mathbf{x}^k, \lambda^k)$  and  $\tilde{\omega}^k = \text{col}(\tilde{\mathbf{x}}^k, \tilde{\lambda}^k)$  are the stacked vectors of the iterates and  $R_{\text{FB}}$  is the FB operator defined in (21). The convergence analysis of inertial schemes as in (23) are studied in [32]; while more precise conditions for the convergence of (23) are derived in [33, Th. 1].

### C. Algorithms for Merely Monotone Aggregative Games

When the pseudo-gradient mapping  $F$  is non-cocoercive, nonstrictly monotone, then Algorithm 1 may fail to converge, see [25] for an example of nonconvergence. Few algorithms are available in the literature for solving merely monotone (aggregative) games with coupling constraints, each with important technical or computational limitations.

*Iterative Tikhonov Regularization (Algorithm 2):* To be applicable to games with (non-cocoercive, non-strictly) monotone pseudo-gradient mapping, the forward-backward algorithm should be augmented with a vanishing regularization. This approach is known as ITR and generates a forward-backward algorithm with double-layer vanishing step sizes [26, §1.3(a)]

$$\forall i : \begin{cases} y_i^k = x_i^k - \gamma^k (\nabla_{x_i} f_i(x_i^k, \text{avg}(\mathbf{x}^k)) + A_i^\top \lambda^k + \epsilon^k x_i^k) \\ x_i^{k+1} = \text{proj}_{\Omega_i}(y_i^k), \quad d_i^{k+1} = A_i x_i^{k+1} - b_i \\ \lambda^{k+1} = \text{proj}_{\mathbb{R}_{\geq 0}^m}(\lambda^k + \gamma^k (N \text{avg}(\mathbf{d}^k) - \epsilon^k \lambda^k)). \end{cases}$$

The convergence proof is based on the fact that the actual step size  $\gamma^k$  must vanish faster than the vanishing regularization parameter  $\epsilon^k$ , [26, (A2.2), §2.1]. The extension of ITR schemes to nonsmooth games can be possibly achieved by discretizing the algorithm proposed in [34, § 4].

*Inexact Preconditioned Proximal-Point (Algorithm 3):* Recently, the inexact preconditioned proximal-point (PPP) method [27] was proposed to solve monotone (aggregative) games, virtually with no additional technical assumption other than monotonicity of the PS mapping  $P$ . When applied to the game in (4), the PPP [27, Algorithm 2] generates a double-layer algorithm, in which at each (outer) iteration  $k$ , the inner loop consists of solving (inexactly) an aggregative game without coupling constraints and with cost functions  $\bar{J}_i$ 's defined as

$$\bar{J}_i^k(x_i, \mathbf{x}_{-i}) = J_i(x_i, \mathbf{x}_{-i}) + (A_i^\top \lambda^k)^\top x_i + \alpha_i \|x_i - x_i^k\|^2$$

where  $\lambda^k$ , i.e., the dual variable, and  $x_i^k$ , i.e., the so-called centroid, stay fixed during the inner iterations. When the subgame is solved with the desired precision  $\epsilon^k$ , namely, an  $\epsilon^k$ -NE profile  $\bar{\mathbf{x}}^k$  is reached, the agents update their centroids

$$(\forall i \in \mathcal{I}) : x_i^{k+1} = \bar{x}_i^k, \quad d_i^{k+1} = 2A_i x_i^{k+1} - A_i x_i^k - b_i.$$

Finally, the central coordinator updates the dual variable as

$$\lambda^{k+1} = \text{proj}_{\mathbb{R}_{\geq 0}^m}(\lambda^k + \beta \text{avg}(\mathbf{d}^{k+1})).$$

The primal-dual dynamics generated by the PPP can be cast in compact form as the fixed-point iteration

$$\omega^{k+1} = J_{\Phi^{-1}T}(\omega^k) + e^k \quad (24)$$

where  $\omega^k = \text{col}(\mathbf{x}^k, \lambda^k)$  is the vector of primal-dual iterates,  $J_{\Phi^{-1}T}$  is the so-called resolvent operator of the mapping  $\Phi^{-1}T$ , defined as  $J_{\Phi^{-1}T} := (\text{Id} + \Phi^{-1}T)^{-1}$ , and  $e^k$  is an error term that accounts for the inexact computations of  $J_{\Phi^{-1}T}(\omega^k)$ .

When the mapping  $T$  is maximally monotone (Lemma 2) and the step sizes in the main diagonal of  $\Phi$  are set such that  $\Phi \succ 0$ , then the resolvent  $J_{\Phi^{-1}T}$  is firmly nonexpansive [24, Prop. 23.8] (1/2-averaged) w.r.t. the  $\Phi$ -induced norm, i.e.,  $\|\cdot\|_\Phi$ . Moreover, if the error sequence  $(e^k)_{k \in \mathbb{N}}$  is summable (which is guaranteed by solving the regularized subgames with increasing precision, namely  $\sum_{k=0}^{\infty} \epsilon_k < \infty$ ), then the inexact fixed-point iteration (24) converges to some  $\omega^* := \text{col}(\mathbf{x}^*, \lambda^*) \in \text{fix}(J_{\Phi^{-1}T}) = \text{zer}(T) \neq \emptyset$  [24, Prop. 5.34], where  $\mathbf{x}^*$  is a v-GNE. We refer to [27] for a complete convergence analysis of Algorithm 3.

*Remark 8 (Computational Limitations of ITR and PPP):* The solution of each subgame of the PPP (see Algorithm 3), requires nested (inner) iterations, and, therefore, multiple communication stages between the agents and the central coordinator. Similarly to the (doubly) vanishing step sizes of the ITR schemes (see Algorithm 2), that lead to slow speed of convergence in practice, we can regard double-layer or nested iterations as an important computational limitation.  $\square$

*Tseng's Forward-Backward-Forward Splitting:* To solve non-cocoercive, non-strictly monotone aggregative games via non-vanishing iterative steps or nested iterations, the forward-backward-forward (FBF) method [24, §26.6] adds an additional forward step to the FB algorithm. In Algorithm 4, we introduce a modified version of the FBF algorithm for aggregative games, originally proposed in [13, Algorithm 1].

Algorithm 4 improves [13, Algorithm 1] on two main aspects.

- (i) (Partially uncoordinated step sizes) each agent  $i \in \mathcal{I}$  and the central coordinator have decision authority on their own local step sizes.
- (ii) (Additional projection) The local updates in step 3 and the central update in step 4 are projected onto the local feasible sets,  $\Omega_i$ 's and  $\mathbb{R}_{\geq 0}^m$ , respectively. These additional projections make sure that the iterates  $x_i^k$ 's live in the domain of correspondent functions  $f_i$ 's and in fact can improve the convergence speed of the algorithm.

The convergence analysis of Algorithm 4 is (almost) identical to that of [13, Algorithm 1], thus, we discuss it briefly next.

*Algorithm 4 as a fixed-point iteration:* In compact form, the dynamics generated by Algorithm 4 read as

$$\omega^{k+1} = R_{\text{FBF}}(\omega^k) \quad (25)$$

where  $\omega^k = \text{col}(\mathbf{x}^k, \lambda^k)$  is the stacked vector of the primal-dual variables and  $R_{\text{FBF}}$  is the so-called FBF operator, i.e.,

$$R_{\text{FBF}} := \text{proj}_{\Omega \times \mathbb{R}_{\geq 0}^m} \circ ((\text{Id} - \Psi^{-1}U_1) \circ J_{\Psi^{-1}U_2} \circ (\text{Id} - \Psi^{-1}U_1) + \Psi^{-1}U_1)$$

---

**Algorithm 4:** Tseng's forward-backward-forward (FBF).
 

---

**Initialization:** For all  $i \in \mathcal{I}$ ,  $x_i^0 \in \mathbb{R}^n$  and  $0 < \alpha_i < (\ell + \|A\|)^{-1}$ ;  $\lambda^0 \in \mathbb{R}_{\geq 0}^m$  and  $0 < \beta < (\ell + \|A\|)^{-1}$ .

**Iterate until convergence:**

1. Local: Strategy update, for all  $i \in \mathcal{I}$ :
 
$$y_i^k = x_i^k - \alpha_i (\nabla_{x_i} f_i(x_i^k, \text{avg}(\mathbf{x}^k)) + A_i^\top \lambda^k)$$

$$\tilde{x}_i^k = \text{prox}_{\alpha_i g_i + \iota_{\Omega_i}}(y_i^k)$$

$$\tilde{d}_i^k = A_i x_i^k - b_i$$
  2. Central coordinator: dual variable update
 
$$\tilde{\lambda}^k = \text{proj}_{\mathbb{R}_{\geq 0}^m}(\lambda^k + \beta \text{avg}(\tilde{\mathbf{d}}^k))$$
  3. Local: Strategy update, for all  $i \in \mathcal{I}$ :
 
$$r_i^{k+1} = \tilde{x}_i^k - \alpha_i (\nabla_{x_i} f_i(\tilde{x}_i^k, \text{avg}(\tilde{\mathbf{x}}^k)) + A_i^\top \tilde{\lambda}^k)$$

$$x_i^{k+1} = \text{proj}_{\Omega_i}(x_i^k - y_i^k + r_i^{k+1})$$

$$d_i^{k+1} = A_i \tilde{x}_i^k - b_i$$
  4. Central coordinator: dual variable update
 
$$\lambda^{k+1} = \text{proj}_{\mathbb{R}_{\geq 0}^m}(\tilde{\lambda}^k + \beta(\text{avg}(\mathbf{d}^{k+1}) - \text{avg}(\tilde{\mathbf{d}}^k)))$$
- 

where  $U_1$  and  $U_2$  characterize an alternative splitting of the mapping  $T$  in (15), i.e.,  $T = U_1 + U_2$ , where

$$U_1 : \omega \mapsto \text{col}(F(\mathbf{x}), b) + S\omega \quad (26)$$

$$U_2 : \omega \mapsto (N_{\Omega}(\mathbf{x}) + G(\mathbf{x})) \times N_{\mathbb{R}_{\geq 0}^m}(\lambda) \quad (27)$$

and  $\Psi$  is the preconditioning matrix, here defined as

$$\Psi := \begin{bmatrix} \bar{\alpha}^{-1} \otimes I_n & \mathbf{0} \\ \mathbf{0} & N\beta^{-1}I_m \end{bmatrix}. \quad (28)$$

When the mappings  $U_1$  and  $U_2$  are maximally monotone (which can be proven when Assumption 3 holds true by following a similar technical reasoning of that in Lemma 2),  $U_1$  is Lipschitz continuous (see Assumption 3) and the step sizes in the main diagonal of  $\Psi$  are set small enough, then the fixed-point iteration (25) converges to some  $\omega^* := \text{col}(\mathbf{x}^*, \lambda^*) \in \text{fix}(R_{\text{FBF}}) = \text{zer}(T) \cap (\Omega \times \mathbb{R}_{\geq 0}^m) = \text{zer}(T)$  [24, Th. 26.17], where  $\mathbf{x}^*$  is a v-GNE. We refer to [13, Th. 2], for a complete convergence analysis, which is applicable to Algorithm 4.

*Remark 9 (Double communication round):* At each central and local update of Algorithm 4 a communication takes place. Hence, each iteration of Algorithm 4 requires two communication rounds between the agents and central operator.  $\square$

Finally, we note that an inertial version of Algorithm 4, without the extra projections (in steps 3, 4) and with fully coordinated step sizes that match across the agents and the central coordinator can be derived based on [35, Th. 4].

#### IV. GENERALIZED NASH EQUILIBRIUM SEEKING: ADVANCED ALGORITHMS

In this section, we design two novel semi-decentralized GNE seeking algorithms obtained by solving the monotone inclusion in (16) with different zero-finding methods: the forward-reflected-backward splitting [36] and, for a particular subclass of aggregative games with linear-coupling functions, the proximal-point method with (alternated) inertia. The main features of the

proposed algorithms, e.g., convergence guarantees and communication requirements, are summarized and compared with those of the existing methods in Table I.

##### A. (Inertial) Forward-Reflected-Backward Algorithm

In this section, we present a single-layer, single communication round algorithm for monotone generalized aggregative games that overcomes the technical and computational limitations of all the algorithms in Section III-C. The design of the proposed method (i.e., Algorithm 5) is based on the *forward-reflected-backward splitting* (FoRB) recently proposed in [36] to find a zero of the sum of two maximally monotone operators, one of which is single-valued and Lipschitz continuous.

Also for Algorithm 5, we describe the generated dynamics as a compact inertial iteration, step

$$\omega^{k+1} = (\text{Id} + \Phi^{-1}T_2)^{-1}(\omega^k - 2\Phi^{-1}T_1(\omega^k) + \Phi^{-1}T_1(\omega^{k-1}) + \theta(\omega^k - \omega^{k-1})) \quad (29)$$

where  $\omega^k = \text{col}(\mathbf{x}^k, \lambda^k)$  is the stacked vector of primal-dual variables, the components mappings  $T_1, T_2$  as in (17) and (18) and the preconditioning  $\Phi$  as in (22). If the step sizes in the main diagonal of  $\Phi$  are chosen small enough, then the iteration (29), namely, the inertial FoRB splitting [36, Corollary 4.4] on the operators  $\Phi^{-1}T_1$  and  $\Phi^{-1}T_2$ , converges to some  $\omega^* := \text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(T_1 + T_2) = \text{zer}(T)$ , where  $\mathbf{x}^*$  is a v-GNE.

Our first main result is to establish global convergence of Algorithm 5 to a v-GNE when the mapping  $F$  is maximally monotone and Lipschitz continuous (see Assumption 3) and the step sizes are chosen small enough.

*Theorem 1 (Convergence of FoRB (Algorithm 5)):* Let Assumptions 1–4 hold true. The sequence  $(\text{col}(\mathbf{x}^k, \lambda^k))_{k \in \mathbb{N}}$  generated by Algorithm 5, globally converges to some  $\text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(T) \neq \emptyset$ , where  $\mathbf{x}^*$  is a v-GNE.  $\square$

*Proof:* See Appendix VI-B.  $\blacksquare$

*Remark 10 (Single communication round):* Algorithms 4 (FBF) and 5 (FoRB) are the only single-layer, fixed-step algorithms for GNE seeking in (noncocoercive, nonstrictly) monotone generalized (aggregative) games. The main advantage of Algorithm 5 is that it requires only one communication round

---

**Algorithm 5:** Inertial FoRB (I-FoRB).
 

---

**Initialization:**  $\theta \in [0, 1/3)$  and  $\delta > 2\ell/(1 - 3\theta)$ , with  $\ell$  as in Assumption 3;  $\forall i \in \mathcal{I}$ ,  $x_i^0, \lambda_i^{-1} \in \mathbb{R}^n$  and

$$0 < \alpha_i \leq (\|A_i\| + \delta)^{-1}; \lambda^0, \lambda^{-1} \in \mathbb{R}_{\geq 0}^m,$$

$$0 < \beta \leq \left( \frac{1}{N} \sum_{i=1}^N \|A_i\| + \frac{1}{N} \delta \right)^{-1}.$$

**Iterate until convergence:**

1. Local: Strategy update, for all  $i \in \mathcal{I}$ :
 
$$r_i^k = 2\nabla_{x_i} f_i(x_i^k, \text{avg}(\mathbf{x}^k)) - \nabla_{x_i} f_i(x_i^{k-1}, \text{avg}(\mathbf{x}^{k-1}))$$

$$x_i^{k+1} = \text{prox}_{\alpha_i g_i + \iota_{\Omega_i}}(x_i^k - \alpha_i(r_i^k + A_i^\top \lambda^k) + \theta(x_i^k - x_i^{k-1}))$$

$$d_i^{k+1} = 2A_i x_i^{k+1} - A_i x_i^k - b_i$$
  2. Central coordinator: dual variable update
 
$$\lambda^{k+1} = \text{proj}_{\mathbb{R}_{\geq 0}^m}(\lambda^k + \beta \text{avg}(\mathbf{d}^{k+1}) + \theta(\lambda^k - \lambda^{k-1}))$$
-

TABLE I

COMPARISON AMONG v-GNE ALGORITHMS. LEGEND: C STANDS FOR COORDINATED STEP SIZES, P-UC FOR PARTIALLY UNCOORDINATED, F-UC FOR FULLY UNCOORDINATED; MON FOR MONOTONE, SMON FOR STRONGLY MONOTONE, COCO STANDS FOR (MONOTONE AND) COCOERCIVE

	APA [15, Alg. 2]	pFB [22] (Alg. 1)	ITR [26] (Alg. 2)	PPP [27] (Alg. 3)	FBF [13] (Alg. 4)	FoRB (Alg. 5, $\theta = 0$ )	cPPP (Alg. 6, $\theta^k = 0$ )
Communications / iteration	1	1	1	$\infty$	2	1	1
Local step sizes	fixed, C	fixed, P-UC	vanishing, P-UC	fixed, P-UC	fixed, P-UC	fixed, P-UC	fixed, F-UC
Pseudo-subdifferential	SMON	COCO	MON	MON	MON	MON	MON
Local cost functions	diff.	non-diff. as in (2)	diff.	non-diff. as in (2)	non-diff. as in (2)	non-diff. as in (2)	linear coupling as in (30)
Inertia		Alg. 1B		✓	[35, Th. 4]	✓	✓
Alternating inertia		Cor. 1		✓			Cor. 2
Over-relaxation		✓		✓			Alg. 6B
Acceleration parameters	-	C	-	F-UC	C	P-UC	F-UC

(between the agents and the central coordinator) per iteration instead of the two required by the FBF, see also Remark 9.  $\square$

### B. Customized Preconditioned Proximal-Point Algorithm

In this section, we focus on a particular class of aggregative games, where the cost functions have the form

$$J_i(x_i, \mathbf{x}_{-i}) = g_i(x_i) + (C \text{ avg}(\mathbf{x}))^\top x_i \quad (30)$$

where  $C = C^\top$  is a symmetric matrix. We emphasize that this particular structure arises in several engineering applications, where  $x_i$  denotes the usage level of a certain commodity, whose disutility is modeled by the cost function  $g_i(x_i)$ , while the term  $C \text{ avg}(\mathbf{x})$  represents a price function that linearly depends on the average usage level of the population, see [5], [30], [37]–[40] for some application examples.

The following lemma shows that aggregative games with such special structure are generalized potential games [41, Def. 2.1].

*Lemma 3:* Consider monotone aggregative games with agent cost functions as in (30) and  $C = C^\top$ . There exists a continuous function  $\phi : \mathbb{R}^{nN} \rightarrow \mathbb{R}$  such that  $\partial\phi = P$ , with  $P = \prod_{i=1}^N \partial_{x_i} J_i(x_i, \mathbf{x}_{-i})$ .  $\square$

*Proof:* For aggregative games with linear coupling functions as in (30), the pseudo-subdifferential  $P$  in (6) reads as

$$P = \prod_{i=1}^N \partial_{x_i} g_i + \frac{1}{N} (I_N + \mathbf{1}_N \mathbf{1}_N^\top) \otimes C. \quad (31)$$

Let  $\phi(\mathbf{x}) := \sum_{i=1}^n g_i(x_i) + \frac{1}{2} \mathbf{x}^\top (\frac{1}{N} (I_N + \mathbf{1}_N \mathbf{1}_N^\top) \otimes C) \mathbf{x}$ , then it is easy to verify that  $\partial\phi = P$ .  $\blacksquare$

It follows by Lemma 3 that a v-GNE corresponds to a solution to the optimization problem  $\text{argmin} \phi(\mathbf{x})$  s.t.  $\mathbf{x} \in \mathcal{X}$ . However, in many practical setups, a centralized solution to this problem is not viable since it would require a high degree of coordination among selfish agents and also an “unbearable overload of information exchange” [41, §3.3]. Moreover, distributed optimization algorithms, see, e.g., [42], can only deal with feasible sets  $\mathcal{X}$  in (7) with Cartesian product structure (namely, the case of non-generalized games) and cost functions with a separable form (i.e., dependent only on local decision variables). This motivates us to investigate a customized algorithm for aggregative games with cost functions as in (30), which we summarize in Algorithm 6 and denote as I-cPPP (or cPPP, when  $\theta^k \equiv 0$ ).

#### Algorithm 6: Inertial customized PPP (I-cPPP).

**Initialization:**  $0 \leq \theta^k \leq \theta^{k+1} \leq \bar{\theta} < 1/3$  for all  $k \geq 0$ ;  
for all  $i \in \mathcal{I}$ ,  $x_i^0 \in \mathbb{R}^n$ ,  $0 < \alpha_i < \|A_i\| + \frac{N-1}{N} \|C\|$ ;  $\lambda^0 \in \mathbb{R}_{\geq 0}^m$ ,  $0 < \beta < \left(\frac{1}{N} \sum_{i=1}^N \|A_i\|\right)^{-1}$ .

**Iterate until convergence:**

1. Local: Strategy update, for all  $i \in \mathcal{I}$ :

$$y_i^k = \tilde{x}_i^k - \alpha_i (C \text{ avg}(\tilde{\mathbf{x}}^k) + A_i^\top \tilde{\lambda}^k)$$

$$x_i^{k+1} = \text{argmin}_{z \in \Omega_i} g_i(z) + \frac{1}{2\alpha_i} \|z - y_i^k\|^2 + \frac{1}{N} (C(z - \tilde{x}_i^k))^\top z$$

$$\tilde{x}_i^{k+1} = x_i^{k+1} + \theta^k (x_i^{k+1} - x_i^k) \quad (32)$$

$$d_i^{k+1} = 2A_i x_i^{k+1} - A_i \tilde{x}_i^k - b_i$$

2. Central Coordinator: dual variable update

$$\lambda^{k+1} = \text{proj}_{\mathbb{R}_{\geq 0}^m} (\tilde{\lambda}^k + \beta \text{ avg}(d^{k+1}))$$

$$\tilde{\lambda}^{k+1} = \lambda^{k+1} + \theta^k (\lambda^{k+1} - \lambda^k) \quad (33)$$

The following theorem establishes global convergence of Algorithm 6 to a v-GNE of aggregative games with linear coupling functions as in (30), when the associated PS  $P$  is maximally monotone, as postulated next.

*Assumption 6:* The pseudo-subdifferential mapping  $P$  in (31) is maximally monotone over  $\Omega$ .  $\square$

We remark that this assumption is less strict than Assumption 3, since the monotonicity of the coupled part  $F$  is not required. Necessary and sufficient conditions for the (strong) monotonicity of  $P$  for this class of aggregative games are discussed in [43, Cor. 1]. For instance,  $C \succcurlyeq 0$  is sufficient to guarantee a maximally monotone PS mapping.

*Theorem 2 (Convergence of I-cPPP (Algorithm 6)):* Consider the game in (4) with cost functions in (30). Let Assumptions 1, 2, 4, 6 hold. Then, the sequence  $(\text{col}(x^k, \lambda^k))_{k \in \mathbb{N}}$  generated by Algorithm 6 globally converges to some  $\text{col}(x^*, \lambda^*) \in \text{zer}(T)$ , where  $x^*$  is a v-GNE.  $\square$

*Proof:* See Appendix VI-C.  $\blacksquare$

*Remark 11 (v-GAE seeking via I-cPPP):* As for gradient-based methods, to compute a v-GNE via Algorithm 6, the agents must know the population size  $N$ . However, if an approximate solution, i.e., a v-GAE, is equally desirable, this requirement can be relaxed by removing the correction term  $\frac{1}{N}(C(z - \tilde{x}_i^k))^\top z$  in the local primal update of each agent  $i$ .  $\square$

*Algorithm 6 as a fixed-point iteration:* In compact form, the dynamics generated by Algorithm 6 read as

$$\tilde{\omega}^k = \omega^k + \theta^k(\omega^k - \omega^{k-1}) \quad (34a)$$

$$\omega^{k+1} = J_{\Phi_C^{-1}T}(\tilde{\omega}^k) \quad (34b)$$

where  $\omega^k = \text{col}(\mathbf{x}^k, \lambda^k)$  is the stacked vector of primal-dual iterates,  $J_{\Phi_C^{-1}T} = (\text{Id} + \Phi_C^{-1}T)^{-1}$  is the generalized resolvent operator of the mapping  $T$  in (15) with preconditioning matrix

$$\Phi_C := \Phi + \begin{bmatrix} \frac{1}{N}(I_N - \mathbf{1}_N \mathbf{1}_N^\top) \otimes C & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (35)$$

with  $\Phi$  as in (22). The iteration in (34) corresponds to the inertial proximal-point method in [44] applied to the mapping  $T$  preconditioned with  $\Phi_C$ . When  $T$  is maximally monotone (which follows by Assumption 6) and the step sizes in the main diagonal of  $\Phi$  are set such that  $\Phi_C \succ 0$ , then  $J_{\Phi_C^{-1}T}$  is firmly nonexpansive ( $\frac{1}{2}$ -averaged) w.r.t. the  $\Phi_C$ -induced norm, i.e.,  $\|\cdot\|_{\Phi_C}$ . Moreover, if the inertial parameter  $\theta^k$  is nondecreasing and small enough, then the inertial fixed-point iteration (34) converges to some  $\omega^* := \text{col}(\mathbf{x}^*, \lambda^*) \in \text{fix}(J_{\Phi_C^{-1}T}) = \text{zer}(T)$  [44, Th. 2.1, Prop. 2.1], where  $\mathbf{x}^*$  is a v-GNE. We provide the full convergence analysis in Appendix A3.

*Remark 12 (cPPP is a single-layer algorithm):* Both the PPP (see Algorithm 3) and our cPPP (see Algorithm 6) rely on the same fixed-point iteration, which is generated by the proximal-point method. However, while the PPP is double-layer, namely, it requires the solution of a subgame at each iteration, cPPP is single-layer. The idea behind the cPPP is in fact to exploit the special structure of the pseudo-subdifferential  $P$  in (31) to customize the preconditioning matrix,  $\Phi_C$ , and in turn solve the inner loop of the PPP with a single implicit iteration, namely, the parallel solution of  $N$  local, decoupled, strongly convex optimization problems. Our cPPP is devised specifically for the subclass of aggregative games presented in this section, thus, its applicability is mainly limited to this class of games.  $\square$

*Remark 13 (Fully-uncoordinated step sizes):* Unlike all the previously presented gradient-based algorithms, the choice of the local step sizes and inertial parameters in Algorithm 6 is based on local information only.<sup>2</sup> To the best of our knowledge, this is the first and only inertial, fixed-step v-GNE seeking algorithm that enjoys this important property.  $\square$

*Over-Relaxed cPPP (Algorithm 6B):* To conclude this section, we present the over-relaxed variant of cPPP, i.e., or-cPPP. This new method is obtained by substituting the inertial steps of primal and dual variables in Algorithm 6, i.e., (32) and (33), respectively, with the relaxation steps

$$\tilde{x}_i^{k+1} = \tilde{x}_i^k + \theta^k(x_i^{k+1} - \tilde{x}_i^k) \quad (36)$$

<sup>2</sup>Except for the population size  $N$ , which is implicitly necessary for computing an exact v-GNE.

$$\tilde{\lambda}_i^{k+1} = \tilde{\lambda}_i^k + \theta^k(\lambda_i^{k+1} - \tilde{\lambda}_i^k) \quad (37)$$

where the relaxation sequence  $(\theta^k)_{k \in \mathbb{N}}$  must be chosen s.t.

$$\theta^k \in [0, 2] \quad \forall k \in \mathbb{N}, \quad \sum_{k \in \mathbb{N}} \theta^k(2 - \theta^k) = \infty. \quad (38)$$

Similarly to Algorithm 6, or-cPPP can be compactly cast as the following Krasnosel'skii–Mann fixed-point iteration:

$$\tilde{\omega}^{k+1} = \tilde{\omega}^k + \theta^k(J_{\Phi_C^{-1}T}(\tilde{\omega}^k) - \tilde{\omega}^k). \quad (39)$$

Thus, its convergence readily follows by [24, Prop. 5.16], since the generalized resolvent  $J_{\Phi_C^{-1}T}$  is  $\frac{1}{2}$ -averaged w.r.t. the  $\Phi_C$ -induced norm [24, Prop. 23.8]. While there is no interest in doing under-relaxation with  $\theta^k$  less than 1, over-relaxation with  $\theta^k$  larger than 1 (close to 2) may be beneficial for the convergence speed, as often observed in practice.

### C. Alternating Inertial Steps for Averaged Operators

In this section, we propose an alternating inertial scheme, which is applicable to the algorithms in Sections III and IV, and whose updates can be described as a special fixed-point iteration of an averaged operator. An advantage of this scheme is that the generated even subsequence is contractive (Fejér monotone) towards a v-GNE. Furthermore, the inertial extrapolation step sizes,  $\theta_k$ , can freely vary in  $[0, 1)$ , namely, they do not need to be monotonically nondecreasing. These requirements are less restrictive than those in [32] and [44].

Next, we first introduce the idea of alternated inertia in operator-theoretic terms, and then apply it to two v-GNE seeking algorithms, the I-pFB (see Algorithm 1B) and the I-cPPP (see Algorithm 6). Let  $R$  be an averaged mapping. The alternating inertial Banach–Picard iteration then reads as follows:

$$\tilde{\omega}^k := \begin{cases} (1 + \theta)\omega^k - \theta\omega^{k-1}, & \text{if } k \text{ odd} \\ \omega^k, & \text{if } k \text{ even} \end{cases} \quad (40a)$$

$$\omega^{k+1} = R(\tilde{\omega}^k) \quad (40b)$$

where  $\omega^{-1} = \omega^0$  is the initialization.

*Lemma 4:* Let  $R$  be  $\eta$ -averaged, with  $\text{fix}(R) \neq \emptyset$ . Then, the even subsequence  $(\omega^{2k+2})_{k \in \mathbb{N}}$  generated by (40), with  $\theta \in (0, \frac{1-\eta}{\eta})$ , converges to some  $\bar{\omega} \in \text{fix}(R)$ .  $\square$

*Proof:* The odd and even subsequences in (40b) read as

$$\forall k \in \mathbb{N} : \begin{cases} \omega^{2k+1} = R(\omega^{2k}) \\ \omega^{2k+2} = R((1 + \theta)\omega^{2k+1} - \theta\omega^{2k}) \\ \quad = R \circ ((1 + \theta)R - \theta\text{Id})(\omega^{2k}). \end{cases} \quad (41)$$

Let us define the mapping  $R_\theta := R \circ ((1 + \theta)R - \theta\text{Id})$ . The next lemma shows that, for  $\theta$  small enough,  $R_\theta$  is averaged and has the same fixed points of  $R$ .  $\blacksquare$

*Lemma 5:* Let  $R$  be  $\eta$ -averaged, with  $\eta \in (0, 1)$ , and set  $\theta \in (0, (1 - \eta)/\eta)$ . The following statements hold:

- (i)  $(1 + \theta)R - \theta\text{Id}$  is  $\mu$ -averaged, with  $\mu = \eta(1 + \theta)$ ;
- (ii)  $\text{fix}((1 + \theta)R - \theta\text{Id}) = \text{fix}(R)$ ;
- (iii)  $R_\theta$  is  $\nu$ -averaged, with  $\nu = \frac{\eta + \mu - 2\eta\mu}{1 - \eta\mu} \in (0, 1)$ ;
- (iv)  $\text{fix}(R_\theta) = \text{fix}(R)$ .  $\square$

*Proof:* (i) It directly follows from [24, Prop. 4.40]. (ii)  $\omega \in \text{fix}((1 + \theta)R - \theta \text{Id}) \Leftrightarrow (1 + \theta)R(\omega) - \theta(\omega) = \omega \Leftrightarrow (1 + \theta)R(\omega) = (1 + \theta)\omega \Leftrightarrow \omega \in \text{fix}(R)$ . (iii) It follows by [24, Prop. 4.44], since  $R_\theta$  is the composition of  $R$  and  $(1 + \theta)R - \theta \text{Id}$ , that are  $\eta$ - and  $\mu$ -averaged, respectively. (iv) It follows by [24, Cor. 4.51] that  $\text{fix}(R_\theta) = \text{fix}(R \circ ((1 + \theta)R - \theta \text{Id})) = \text{fix}(R) \cap \text{fix}((1 + \theta)R - \theta \text{Id}) = \text{fix}(R)$ .

In view of (41) and Lemma 5, the even subsequence in (40b) can be recast as

$$\omega^{2k+2} = R_\theta(\omega^{2k}), \quad \forall k \in \mathbb{N} \quad (42)$$

where  $R_\theta$  is  $\nu$ -averaged, with  $\nu \in (0, 1)$  given by Lemma 5 (iii). Thus, the convergence of the sequence  $(\omega^{k+2})_{k \in \mathbb{N}}$  to some  $\bar{\omega} \in \text{fix}(R_\theta) = \text{fix}(R)$  follows by [24, Prop. 5.16]. ■

Finally, we propose some explicit rules to choose the alternating inertial extrapolation step sizes for the pFB (see Algorithm 1B) and for the cPPP (see Algorithm 6). In fact, in Section V-B, we observe via numerical simulations that in some cases these alternating-inertial variants outperform the standard-inertial algorithms in terms of convergence speed. Let us then conclude the section with the associated convergence results.

*Corollary 1 (Convergence of alternating inertial pFB (aI-pFB)):* Let Assumptions 1, 2, 4, and 5 hold true. Then, the sequence  $(\text{col}(\mathbf{x}^k, \lambda^k))_{k \in \mathbb{N}}$  generated by Algorithm 1B with extrapolation steps set as

$$\theta^k = \begin{cases} \theta \in \left[0, \frac{2\delta\gamma-1}{2\delta\gamma}\right), & \text{if } k \text{ odd} \\ 0 & \text{if } k \text{ even} \end{cases} \quad (43)$$

globally converges to some  $\text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(T) \neq \emptyset$ , where  $\mathbf{x}^*$  is a v-GNE. □

*Proof:* The pFB algorithm (see Algorithm 1) reads as the fixed-point iteration in (20), where the mapping  $R_{\text{FB}}$  is  $\eta := (\frac{2\delta\gamma}{4\delta\gamma-1})$ -averaged w.r.t. the  $\Phi$ -induced norm. Therefore, the iteration with alternated inertia and extrapolation step sizes  $0 \leq \theta < \frac{1-\eta}{\eta} = \frac{2\delta\gamma-1}{2\delta\gamma}$  converges by Lemma 4. ■

*Corollary 2 (Convergence of alternating inertial cPPP (aI-cPPP)):* Consider the game in (4) with cost functions as in (30). Let Assumptions 1, 2, 4, 6 hold true. Then, the sequence  $(\text{col}(\mathbf{x}^k, \lambda^k))_{k \in \mathbb{N}}$  generated by Algorithm 6 with extrapolation steps

$$\theta_k = \begin{cases} \theta \in [0, 1), & \text{if } k \text{ odd} \\ 0, & \text{if } k \text{ even} \end{cases} \quad (44)$$

globally converges to some  $\text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(T) \neq \emptyset$ , where  $\mathbf{x}^*$  is a v-GNE. □

*Proof:* cPPP reads as the fixed-point iteration in (34) with  $\theta^k = 0$  for all  $k > 0$ , where the resolvent mapping  $J_{\Phi_C^{-1}T}$  is firmly-nonexpansive, i.e.,  $\eta := \frac{1}{2}$ -averaged, w.r.t. the  $\Phi_C$ -induced norm. Therefore, the iteration with alternating inertia and extrapolation step sizes  $0 \leq \theta < \frac{1-\eta}{\eta} = 1$  converges by Lemma 4. ■

*Remark 14 (Convergence rate):* We recall that pFB, cPPP and their alternating-inertial variants can be compactly cast as fixed-point iterations of some averaged operators. It follows by [45, Th. 1] that their sequences of fixed-point residuals, i.e.,  $\|\omega^{k+1} - \omega^k\|^2$ , converge with rate  $o(1/(k+1))$ . □

## V. ILLUSTRATIVE APPLICATION: CHARGING CONTROL OF PLUG-IN ELECTRIC VEHICLES

To study the performance of the proposed algorithms, we formulate a charging coordination problem for a large population of noncooperative plug-in electric vehicles (PEV) as a generalized aggregative game, as in [15, §6]. In Section V-A, we introduce the model for the PEV agents, formalize the charging control game and verify that the necessary technical assumptions are satisfied. In Section V-B, we compare the performance of our algorithm against some standard methods.

### A. Game Formulation

We adopt the same model in [15, §6]. Consider the charging coordination problem for a large population of  $N \gg 1$  noncooperative PEV over a time horizon made of multiple charging intervals  $\{1, 2, \dots, n\}$ . The state of vehicle  $i$  at time  $t$  is denoted by the variable  $s_i(t)$ . The time evolution of  $s_i(t)$  is described by the discrete-time system

$$s_i(t+1) = s_i(t) + b_i x_i(t), \quad t = 1, \dots, n$$

where  $x_i(t)$  denotes the charging control input and  $b_i$  the charging efficiency.

*Constraints:* At each time instant  $t$ , the charging input  $x_i(t)$  must be nonnegative and cannot exceed an upper bound  $\bar{x}_i(t) \geq 0$ . Moreover, the final state of charge must satisfy  $s_i(n+1) \geq \eta_i$ , where  $\eta_i \geq 0$  is the desired state of charge of vehicle  $i$ . We assume that each PEV agent  $i$  decides on its charging strategy  $x_i = \text{col}(x_i(1), \dots, x_i(n)) \in \Omega_i \subset \mathbb{R}^n$ , where the set  $\Omega_i$  can be expressed as

$$\Omega_i := \left\{ x_i \in \mathbb{R}^n \mid 0 \leq x_i(t) \leq \bar{x}_i(t), \forall t = 1, \dots, n \right. \\ \left. \text{and } \sum_{t=1}^n x_i(t) \geq l_i \right\} \quad (45)$$

where  $l_i = b_i^{-1}(\eta_i - s_i(1))$  and  $s_i(1)$  is the state of charge at the beginning of the time horizon.

Furthermore, for each time instant  $t$ , the overall power that the grid can deliver to the PEV is denoted by  $NK(t)$ , thus introducing the following coupling constraints:

$$\frac{1}{N} \sum_{i=1}^N x_i(t) \leq K(t), \quad \text{for all } t = 1, \dots, n \quad (46)$$

which in compact form can be cast as  $(\mathbf{1}_N^\top \otimes I_n) \mathbf{x} \leq NK$ , with  $K = [K(1), \dots, K(n)]^\top$ .

*Cost Functions:* The cost function of each PEV represents its electricity bill over the horizon of length  $n$  plus a local penalty term  $g_i$  (e.g., the battery degradation cost [7], [46]), i.e.,

$$J_i(x_i, \mathbf{x}_{-i}) = \sum_{t=1}^n g_{i,t}(x_i(t)) + p_t \left( \frac{d(t) + \text{avg}(\mathbf{x}(t))}{\kappa(t)} \right) x_i(t) \\ =: g_i(x_i) + p(\text{avg}(\mathbf{x}))^\top x_i \quad (47)$$

where  $g_i$  is convex and the energy price for each time interval  $p_t : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{> 0}$  is monotonically increasing, continuously differentiable, and depends on the ratio between the total consumption and the total capacity, i.e.,  $(d(t) + \text{avg}(\mathbf{x}(t)))/\kappa(t)$ , where

$d(t)$  and  $\text{avg}(\mathbf{x}(t)) := \frac{1}{N} \sum_{i=1}^N x_i(t)$  represent the non-PEV and PEV demand at time  $t$  divided by  $N$  and  $\kappa(t)$  is the total production capacity divided by  $N$  as in [47, (6)].

*Aggregative Game:* Overall, each PEV  $i$ , given the charging inputs of the other PEV, aims at solving the following optimization problem:

$$(\forall i \in \mathcal{I}) : \begin{cases} \underset{x_i \in \mathbb{R}^n}{\text{argmin}} & g_i(x_i) + p(\text{avg}(\mathbf{x}))^\top x_i \\ \text{s.t.} & x_i \in \Omega_i, \\ & (\mathbf{1}_N^\top \otimes I_n) \mathbf{x} \leq NK. \end{cases} \quad (48)$$

Next, we show that the proposed charging control game in (48) does satisfy our technical setup. The local cost functions  $J_i$ 's in (47) are convex w.r.t. the local variable  $x_i$ , the local constraint sets  $\Omega_i$ 's in (45) are nonempty (for an appropriate choice of the parameters), convex and compact, the coupling constraints in (46) are affine and their intersection with the local constraints nonempty (for an appropriate choice of the parameters), namely, the Slater's condition holds true. Hence, Assumptions 1 and 2 are satisfied. In particular, there exist at least one GNE of the game in (48), see Remark 2.

The correspondent PG in (8) and approximate PG in (11) read more explicitly as follows:

$$\tilde{F}(\mathbf{x}) = \text{col}(\{p(\text{avg}(\mathbf{x}))\}_{i \in \mathcal{I}}) \quad (49)$$

$$F(\mathbf{x}) = \tilde{F}(\mathbf{x}) + \frac{1}{N} \text{col}(\{\nabla_z p(z)|_{z=\text{avg}(\mathbf{x})} x_i\}_{i \in \mathcal{I}}). \quad (50)$$

The following lemma shows the properties of these mappings depending on the choice of the price function  $p$  in (47).

*Lemma 6 ([15, Lemma 3]):* The following conditions hold:

- (i) for all  $i \in \mathcal{I}$ , let  $g_i$  be convex and the price function  $p$  be monotone, then  $\tilde{F}$  in (49) is maximally monotone;
- (ii) for all  $i \in \mathcal{I}$ , let  $g_i$  be convex and the price function  $p$  be affine, i.e.,  $p(\text{avg}(\mathbf{x})) = C \text{avg}(\mathbf{x}) + c$ , with  $C \in \mathbb{R}^{n \times n}$ , and strongly monotone, i.e.,  $(C + C^\top)/2 \succ 0$ , then  $F$  in (50) is strongly monotone.  $\square$

*Proof:* (i) and (ii) follow from [15, Lemma 3 (i)].  $\blacksquare$

## B. Numerical Analysis

In our numerical study, we consider an heterogeneous population of PEV playing over a time horizon of  $n = 24$  charging intervals. All the parameters of the game are drawn from uniform distributions and fixed over the course of a simulation. Specifically, for all  $i \in \mathcal{I}$ , we set: the desired final state of charge  $l_i$  in (45) according to  $l_i \sim (0.5, 1.5)$ , where  $\sim (\tau_1, \tau_2)$  denotes the uniform distribution over an interval  $(\tau_1, \tau_2)$  with  $\tau_1 < \tau_2$ ; for all  $t \in \{1, \dots, n\}$ , the upper charging input bound as  $\bar{x}_i(t) \sim (1, 5)$ , with probability 0.8,  $\bar{x}_i(t) = 0$  otherwise. For all  $t$ , the non-PEV demand  $d(t)$  is taken as the typical base demand over a summer day in the United States [47, Fig. 1];  $\kappa(t) = 12$  kW, and the upper bound  $K(t) = 0.55$  kW is chosen such that the coupling constraints in (46) are active in the middle of the night.

In the remainder of this section, we study the convergence properties of the proposed algorithms on two different scenarios characterized by a different choice of the price function  $p$  and local cost functions  $g_i$ , in (47).

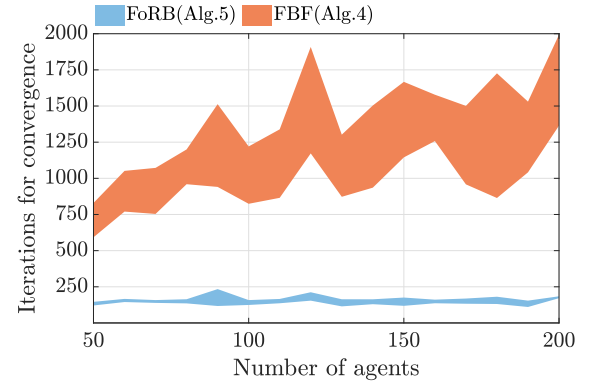


Fig. 1. Number of iterations to achieve convergence for FoRB (Algorithm 5,  $\theta = 0$ ) and FBF (see Algorithm 4) versus populations size  $N$ . The areas contain the outcome of 10 random simulations for each  $N \in \{50, 60, \dots, 200\}$ . Convergence is considered achieved when  $\|\mathbf{x}^k - \mathbf{x}^*\|/\|\mathbf{x}^*\| \leq 10^{-4}$ , where  $\mathbf{x}^*$  is a v-GAE.

1) *Monotone Price Function:* Consider the price function

$$p_t(\text{avg}(\mathbf{x}(t))) := 0.15 \left( \frac{d(t) + \text{avg}(\mathbf{x}(t))}{\kappa(t)} \right)^{1.5}, \quad \forall t. \quad (51)$$

As in [47, §VII.B] and a local cost function  $g_i$  defined as

$$g_i(x_i) = \pi_i \left( \sum_{t=1}^n x_i(t) \right)^2 + a_i^\top x_i, \quad \forall i \in \mathcal{I} \quad (52)$$

where  $\pi_i \sim (0.1, 0.8)$  and  $a_i(t) \sim (0.1, 0.4)$ , for all  $t \in \{1, \dots, n\}$ . Under these choices, it follows from Lemma 6 (i) that the approximate PG in (49) is maximally monotone. Therefore, a v-GAE of the game in (48) can be found with the algorithms in Section III-C and the FoRB (see Algorithm 5).

In Fig. 1, we compare the total number of iterations required by FBF (see Algorithm 4) and FoRB (see Algorithm 5) to achieve convergence to a v-GAE (i.e.,  $\|\mathbf{x}^k - \mathbf{x}^*\|/\|\mathbf{x}^*\| \leq 10^{-4}$ ), over different population sizes  $N$  varying from 50 to 200 agents. For each  $N$ , we run 10 simulations with random parameters. On average, FoRB converges at least 5 times faster than FBF in terms of number of iterations, and, thus, 10 times faster in terms of communication rounds between PEVs and central coordinator. Moreover, unlike FBF, the convergence speed of FoRB seems not affected by increasing the number of agents and randomly varying the parameters of the problem.

2) *Affine Price Function:* Consider the price function

$$p(\text{avg}(\mathbf{x})) := C \text{avg}(\mathbf{x}) + c \quad (53)$$

where  $C = I_n$ ,  $c = \text{col}(d(1), \dots, d(n))$ , and the local convex cost function  $g_i$ , for all  $i \in \mathcal{I}$ , as

$$g_i(x_i) = \frac{1}{2} x_i^\top Q_i x_i + p_i^\top x_i. \quad (54)$$

For instance, the local penalty term  $g_i$  in (54) can model a convex quadratic battery degradation cost as in [7, (5)], [46, (8)], possibly plus a quadratic penalty  $\|x_i - x_i^{\text{ref}}\|^2$  on the deviation from a preferred charging strategy  $x_i^{\text{ref}} \in \Omega_i$ .

Under these choices, the pseudo-gradient mapping  $F$  in (50) is strongly monotone, by Lemma 6 (ii), and Lipschitz continuous, since affine. Thus, it follows by Remark 5 that  $F$  is cocoercive. The unique v-GNE of the game in (48) can be found with the

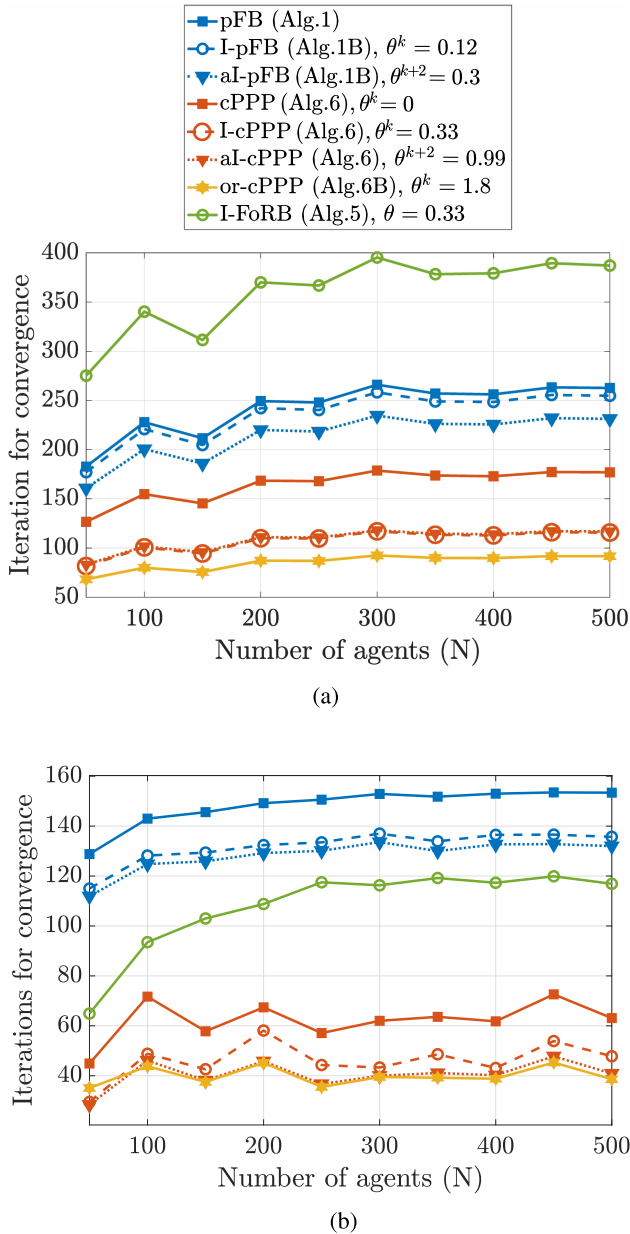


Fig. 2. Iterations to achieve convergence for different population sizes. Each polygon is the average over 120 random simulations. Convergence is considered achieved when  $\|\mathbf{x}^k - \mathbf{x}^*\|/\|\mathbf{x}^*\| \leq 10^{-6}$ . (a) Heterogeneous population of PEVs. The step sizes of pFB and cPPP have been set 1% smaller than their theoretical upper bounds. (b) Homogeneous population of PEVs. The step sizes of pFB and cPPP have been set 1% smaller than their theoretical upper bounds.

algorithms in Section III-B and, since the cost functions have the same structure as in (30), with the cPPP.

First, we consider a heterogeneous population of PEV's, by setting the parameters of the penalty terms  $g_i$  in (54) as follows:  $Q_i = \text{diag}(q(1), \dots, q(n))$ ,  $p_i = \text{col}(p_i(1), \dots, p_i(n))$ , with  $q_i(t) \sim (0.1, 4)$  and  $p_i(t) \sim (0.2, 2)$ , for all  $t$ . In Fig. 2(a), we compare the average number of iterations required to achieve convergence (i.e.,  $\|\mathbf{x}^k - \mathbf{x}^*\|/\|\mathbf{x}^*\| \leq 10^{-6}$ ) for pFB (see Algorithm 1), cPPP (see Algorithm 6) and their inertial variants, for different population sizes  $N$ . For each  $N$ , we run 10 simulations

with random parameters and considered the average number of iterations for convergence. The step sizes of all the algorithms are set 1% smaller than their theoretical upper bounds. On average, cPPP outperforms pFB. For both pFB and cPPP, their inertial variants show better performances with respect to the vanilla-flavour algorithms. Overall, the over-relaxed cPPP is the fastest among all the considered methodologies. We note that, the convergence speed of all the algorithms seems only mildly affected by the population size.

In Fig. 2(b), we repeat the same analysis for an homogeneous population of PEVs. Specifically, we set the parameters of the local penalty term in (54) as  $Q_i = 0.1$  and  $p_i = 0.2$ , for all  $i \in \mathcal{N}$ . The performances of all the algorithms improve w.r.t. to the case with heterogeneous agents. On average, cPPP requires less than half the iterations/communication rounds of pFB. For both pFB and cPPP, their inertial variants show better performances with respect to the standard algorithms. Overall, the alternated inertial cPPP (aI-cPPP) and the over-relaxed cPPP (or-cPPP) are the fastest among all the considered methodologies (less than 50 communication rounds with the central coordinator to achieve a precision of  $10^{-6}$ , independently on the total number of PEVs).

## VI. CONCLUSION AND OUTLOOK

Generalized Nash equilibrium problems in monotone aggregative games can be efficiently solved via accelerated, semi-decentralized, single-layer, single-communication-per-iteration, fixed-step algorithms. For this class of equilibrium problems, the over-relaxation seems the most effective provably-convergent decentralized way to speed up convergence. The study of adaptive step sizes is left for future work.

## APPENDIX

### A. Sufficient Conditions for Cocoercivity of the Pseudo-Gradient

In this appendix, we provide some sufficient conditions – inspired by [48, Section III-B] – for cocoercivity of the pseudo-gradient  $F$  in (8) based on properties of the functions  $(f_i)_{i \in \mathcal{I}}$ .

*Lemma 7:* For all agent  $i \in \mathcal{I}$ , assume that the function  $h_i(\mathbf{x}) : \mathbf{x} \mapsto f_i(x_i, \text{avg}(\mathbf{x}))$  is twice continuously differentiable, and let the following conditions hold:

- $\nabla_{x_i} h_i(\mathbf{x})$  is  $\ell_i$ -Lipschitz continuous on  $\Omega$ ;
- there exists a positive constant  $\mu_i$  such that

$$\inf_{\mathbf{x} \in \Omega} \text{eig}_{\min}(\nabla_{x_i}^2 h_i(\mathbf{x})) \geq \sum_{j \in \mathcal{I} \setminus \{i\}} \sup_{\mathbf{x} \in \Omega} \|\nabla_{x_j, x_i}^2 h_i(\mathbf{x})\| + \mu_i.$$

Then, the pseudo-gradient  $F(\mathbf{x}) = \text{col}((\nabla_{x_i} h_i(\mathbf{x}))_{i \in \mathcal{I}})$  in (8) is  $\gamma$ -cocoercive, with  $\gamma = \min_{i \in \mathcal{I}} \{\mu_i\} / (\max_{i \in \mathcal{I}} \{\ell_i\})^2$ .  $\square$

*Proof:* Define the matrix  $\Upsilon_F \in \mathbb{R}^{N \times N}$ , with entries

$$[\Upsilon_F]_{i,j} := \begin{cases} \inf_{\mathbf{x} \in \Omega} \text{eig}_{\min}(\nabla_{x_i}^2 h_i(\mathbf{x})), & \text{if } i = j \\ -\sup_{\mathbf{x} \in \Omega} \|\nabla_{x_j, x_i}^2 h_i(\mathbf{x})\|, & \text{otherwise.} \end{cases} \quad (55)$$

Under the conditions in Lemma 7 (ii), it follows by the Gershgorin's circle theorem, e.g. [49, Th. 2], that  $\Upsilon_F$  is positive definite with  $\text{eig}_{\min}(\Upsilon_F) = \min_{i \in \mathcal{I}} \{\mu_i\}$ . In turn, it follows

from [48, Prop. 5.(c)] that the pseudo-gradient  $F$  is  $\mu$ -strongly monotone, with  $\mu = \min_{i \in \mathcal{I}} \{\mu_i\}$ . Moreover, under the conditions in Lemma 7 (i) it follows that  $F(\mathbf{x}) = \text{col}((\nabla_{x_i} h_i(\mathbf{x}))_{i \in \mathcal{I}})$  is  $\ell$ -Lipschitz continuous with  $\ell = \max_{i \in \mathcal{I}} \{\ell_i\}$ . Finally, from Remark 5 it follows that  $F$  is  $\gamma$ -cocoercive with  $\gamma = \min_{i \in \mathcal{I}} \{\mu_i\} / (\max_{i \in \mathcal{I}} \{\ell_i\})^2$ . ■

### B. Proof of Theorem 1

To establish global convergence, we show that the following.

- (i) Algorithm 5 corresponds to the (preconditioned) inertial FoRB splitting method [36, (4.12)] in (29).
- (ii) If the step sizes  $\{\alpha_i\}_{i \in \mathcal{I}}$ ,  $\beta$  and the extrapolation parameter  $\theta$  are chosen as in Algorithm 5, then the assumptions of [36, Corollary 4.4] are satisfied, hence  $(\text{col}(\mathbf{x}^k, \lambda^k))_{k \in \mathbb{N}}$  globally converges to some  $\text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(T) \neq \emptyset$ , where  $\mathbf{x}^*$  is a v-GNE.
- (i): Let us recast Algorithm 5 in a more compact form as

$$\begin{aligned} \mathbf{x}^{k+1} &= \text{diag}(\text{prox}_{\alpha_1 g_1 + \iota_{\Omega_1}}, \dots, \text{prox}_{\alpha_N g_N + \iota_{\Omega_N}}) \\ &\quad \circ (\tilde{\mathbf{x}}^k - \bar{\alpha}(2F(\mathbf{x}^k) - F(\mathbf{x}^{k-1}) + A^\top \lambda^k)) \end{aligned} \quad (56)$$

$$\lambda^{k+1} = \text{proj}_{\mathbb{R}_{\geq 0}^m} (\tilde{\lambda}^k + \beta(2A\mathbf{x}^{k+1} - A\mathbf{x}^k - b)) \quad (57)$$

where  $\tilde{\mathbf{x}}^k := \mathbf{x}^k + \theta(\mathbf{x}^k - \mathbf{x}^{k-1})$  and  $\tilde{\lambda}^k := \lambda^k + \theta(\lambda^k - \lambda^{k-1})$ . Since  $\text{diag}(\text{prox}_{\alpha_1 g_1 + \iota_{\Omega_1}}, \dots, \text{prox}_{\alpha_N g_N + \iota_{\Omega_N}}) = (\text{Id} + N_\Omega + \bar{\alpha}G)^{-1}$ , it follows by (56) that  $(\text{Id} + N_\Omega + \bar{\alpha}G)(\mathbf{x}^{k+1}) \in \tilde{\mathbf{x}}^k - \bar{\alpha}(2F(\mathbf{x}^k) - F(\mathbf{x}^{k-1}) + A^\top \lambda^k)$ , which leads to

$$\begin{aligned} - (2F(\mathbf{x}^k) - F(\mathbf{x}^{k-1})) &\in (N_\Omega + G)(\mathbf{x}^{k+1}) + A^\top \lambda^{k+1} \\ &\quad + \bar{\alpha}^{-1}(\mathbf{x}^{k+1} - \tilde{\mathbf{x}}^k) - A^\top (\lambda^{k+1} - \lambda^k) \end{aligned} \quad (58)$$

where we used  $\bar{\alpha}^{-1}N_\Omega(\mathbf{x}^{k+1}) = N_\Omega(\mathbf{x}^{k+1})$ . Equivalently, it follows from (57) that  $(\text{Id} + N_{\mathbb{R}_{\geq 0}^m})(\lambda^{k+1}) \in \tilde{\lambda}^k + \beta \frac{1}{N}(2A\mathbf{x}^{k+1} - A\mathbf{x}^k - b)$ , which leads to:

$$\begin{aligned} -b &\in N_{\mathbb{R}_{\geq 0}^m}(\lambda^{k+1}) - A\mathbf{x}^{k+1} \\ &\quad - A(\mathbf{x}^{k+1} - \mathbf{x}^k) + N\beta^{-1}(\lambda^{k+1} - \tilde{\lambda}^k). \end{aligned} \quad (59)$$

Let  $\omega^k := \text{col}(\mathbf{x}^k, \lambda^k)$  be the stacked vector of the iterates and  $\tilde{\omega}^k = \omega^k + \theta(\omega^k - \omega^{k-1})$ . The inclusions in (58) and (59) can be cast in a more compact form as

$$-(2T_1(\omega^k) - T_1(\omega^{k-1})) \in T_2(\omega^{k+1}) + \Phi(\omega^{k+1} - \tilde{\omega}^k)$$

where  $T_1$ ,  $T_2$ , and  $\Phi$  as in (17), (18), and (19), respectively. By making  $\omega^{k+1}$  explicit in the last inclusion, we obtain

$$\begin{aligned} \omega^{k+1} &= (\text{Id} + \Phi^{-1}T_2)^{-1} \\ &\quad \circ (\tilde{\omega}^k - 2\Phi^{-1}T_1(\omega^k) + \Phi^{-1}T_1(\omega^{k-1})) \end{aligned} \quad (60)$$

which corresponds to (29), thus concluding the proof.

(ii): Before studying the convergence of iteration (60), we show that, if the step sizes are chosen as in Algorithm 5, then the preconditioning matrix  $\Phi$  is positive definite.

**Lemma 8:** Let  $\{\alpha_i\}_{i \in \mathcal{I}}$  and  $\beta$  be set as in Algorithm 5. Then, the following statements hold:

- (i)  $\Phi - \delta I \succeq 0$ ;

(ii)  $\|\Phi^{-1}\| \leq \delta^{-1}$ . □

*Proof:* (i): By the generalized Gershgorin circle theorem [49, Th. 2], each eigenvalue  $\mu$  of the matrix  $\Phi$  in (19) satisfies at least one of the following inequalities:

$$\mu \geq \alpha_i^{-1} - \|A_i^\top\|, \quad \forall i \in \mathcal{I} \quad (61)$$

$$\mu \geq N\beta^{-1} - \sum_{j=1}^N \|A_j^\top\|. \quad (62)$$

Hence, if we set the step sizes  $\{\alpha_i\}_{i \in \mathcal{I}}, \beta$  as in Algorithm 5, the inequalities (61), (62) yield to  $\mu \geq \delta$ , where  $\delta > 0$  by design choice. It follows that the smallest eigenvalue of  $\Phi$ , i.e.,  $\text{eig}_{\min}(\Phi)$ , satisfies  $\text{eig}_{\min}(\Phi) \geq \delta > 0$ . Hence,  $\Phi - \delta I \succeq 0$ .

(ii): Let  $\text{eig}_{\max}(\Phi)$  be the largest eigenvalue of  $\Phi$ . We have that  $\text{eig}_{\max}(\Phi) \geq \text{eig}_{\min}(\Phi) \geq \delta$ . Moreover,  $\|\Phi\| = \text{eig}_{\max}(\Phi) \geq \text{eig}_{\min}(\Phi) = \frac{1}{\|\Phi^{-1}\|} \geq \delta$ . Hence,  $\|\Phi^{-1}\| \leq \delta^{-1}$ . ■

Since  $\Phi^{-1}$  is  $\delta^{-1}$ -Lipschitz, by Lemma 8 (ii), and  $T_1$  is  $\ell$ -Lipschitz, by Assumption 3, then their composition, i.e.,  $\Phi^{-1} \circ T_1$ , is  $\tau$ -Lipschitz continuous, with  $\tau := \delta^{-1}\ell < (1 - 3\theta)/2$ , since  $\delta > 2\ell/(1 - 3\theta)$ , by design choice.

The fixed-point iteration (60), that corresponds to Algorithm 5 by the first part of this proof, is the inertial FoRB splitting algorithm on the mappings  $\Phi^{-1}T_1$  and  $\Phi^{-1}T_2$ . The convergence of (60) to some  $\omega^* := \text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(T_1 + T_2)$  follows by [36, Corollary 4.4, Remark 2.7], since  $\Phi^{-1}T_1$  and  $\Phi^{-1}T_2$  are maximally monotone in the  $\Phi$ -induced norm and  $\Phi^{-1}T_1$  is  $\tau$ -Lipschitz continuous, with  $\tau < (1 - 3\theta)/2$ . To conclude, we note that  $\omega^* \in \text{zer}(\Phi^{-1}T_1 + \Phi^{-1}T_2) = \text{zer}(T)$ , since  $\Phi \succ 0$ , by Lemma 8 (i), and  $T_1 + T_2 = T$ . Since the limit point  $\omega^* \in \text{zer}(T) \neq \emptyset$ , then  $\mathbf{x}^*$  is a v-GNE of the game in (4), by Proposition 1, thus concluding the proof. ■

### C. Proof of Theorem 2

To establish global convergence, we show that the following:

- (i) Algorithm 6 corresponds to the inertial proximal-point method [44, Th. 2.1] in (34);
- (ii) if the step sizes  $\{\alpha_i\}_{i \in \mathcal{I}}$ ,  $\beta$  and the inertial parameters  $\theta^k$  are chosen as in Algorithm 6, then the assumptions of [44, Th. 2.1, Prop. 2.1] are satisfied, hence  $(\text{col}(\mathbf{x}^k, \lambda^k))_{k \in \mathbb{N}}$  globally converges to some  $\text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(T)$ , where  $\mathbf{x}^*$  is a v-GNE.

(i): With some cosmetic manipulations, we can rewrite the local primal update of agent  $i$  as the solution to

$$x_i^{k+1} = \arg\min_{z \in \Omega_i} J_i(z, \tilde{x}_{-i}^k) + (A_i^\top \tilde{\lambda}_i^k)^\top z + \frac{1}{2\alpha_i} \|z - \tilde{x}_i^k\|^2$$

with  $J_i$  as in (30). Equivalently,  $x_i^{k+1}$  must satisfy

$$\begin{aligned} \mathbf{0}_n &\in \partial_{x_i} \left( J_i(x_i^{k+1}, \tilde{x}_{-i}^k) + (A_i^\top \tilde{\lambda}_i^k)^\top x_i^{k+1} \right. \\ &\quad \left. + \frac{1}{2\alpha_i} \|x_i^{k+1} - \tilde{x}_i^k\|^2 \right). \end{aligned}$$

Since  $\partial_{x_i} J_i(x_i^{k+1}, \tilde{x}_{-i}^k) = \partial_{x_i} g_i(x_i^{k+1}) + \frac{1}{N} \sum_{j \neq i}^N Cx_j^k + \frac{2}{N} Cx_i^{k+1}$ , then the previous inclusion can be rewritten as

$$\begin{aligned} \mathbf{0}_n &\in \partial_{x_i} g_i(x_i^{k+1}) + \frac{2}{N} Cx_i^{k+1} + \frac{1}{N} [(\mathbf{1}_N^\top \otimes C)\mathbf{x}^{k+1} - Cx_i^{k+1}] \\ &\quad + A_i^\top \lambda_i^{k+1} + \frac{1}{\alpha_i} (x_i^{k+1} - \tilde{x}_i^k) - \frac{1}{N} [(\mathbf{1}_N^\top \otimes C)(\mathbf{x}^{k+1} - \tilde{\mathbf{x}}^k) \end{aligned}$$



$$+ C(\mathbf{x}_i^{k+1} - \tilde{\mathbf{x}}_i^k) - A_i^\top(\lambda^{k+1} - \lambda^k). \quad (63)$$

By stacking-up the inclusions (63), for all  $i \in \mathcal{I}$ , we obtain

$$\begin{aligned} \mathbf{0}_{n_N} \in G(\mathbf{x}^{k+1}) + \frac{2}{N}(I_N \otimes C)\mathbf{x}^{k+1} - \\ \frac{1}{N}((I_N - \mathbf{11}^\top) \otimes C)\mathbf{x}^{k+1} + A^\top\lambda^{k+1} + \bar{\alpha}^{-1}(\mathbf{x}^{k+1} - \tilde{\mathbf{x}}^k) \\ + \frac{1}{N}((I_N - \mathbf{11}^\top) \otimes C)(\mathbf{x}^{k+1} - \tilde{\mathbf{x}}^k) - A^\top(\lambda^{k+1} - \tilde{\lambda}^k) \end{aligned} \quad (64)$$

where the first 3 terms on the right-hand side correspond to the pseudo-subdifferential mapping  $P(\mathbf{x}^{k+1})$ , i.e.,  $G(\mathbf{x}^{k+1}) + \frac{2}{N}(I_N \otimes C)(\mathbf{x}^{k+1}) - \frac{1}{N}((I_N - \mathbf{11}^\top) \otimes C)(\mathbf{x}^{k+1}) = P(\mathbf{x}^{k+1})$ .

It follows by the dual update in Algorithm 6 that  $(\text{Id} + N_{\mathbb{R}_{\geq 0}^m})(\lambda^{k+1}) \in \lambda^k + \beta \frac{1}{N}(2A\mathbf{x}^{k+1} - A\tilde{\mathbf{x}}^k - b)$ , which yields

$$\begin{aligned} \mathbf{0}_m \in N_{\mathbb{R}_{\geq 0}^m}(\lambda^{k+1}) - (A\mathbf{x}^{k+1} - b) \\ - A(\mathbf{x}^{k+1} - \tilde{\mathbf{x}}^k) + N\beta^{-1}(\lambda^{k+1} - \tilde{\lambda}^k). \end{aligned} \quad (65)$$

Let  $\omega^k := \text{col}(\mathbf{x}^k, \lambda^k)$  be the stacked vector of the iterates. The inclusions in (64) and (65) can be cast in a compact form as

$$\mathbf{0} \in T(\omega^{k+1}) + \Phi_C(\omega^{k+1} - \tilde{\omega}^k)$$

where  $T$  and  $\Phi_C$  as in (15) and (35), respectively. By making  $\omega^{k+1}$  explicit in the last inclusion, we obtain

$$\omega^{k+1} = (\text{Id} + \Phi_C^{-1}T)^{-1}(\tilde{\omega}^k) \quad (66)$$

where the auxiliary updates can be cast in a compact form as

$$\tilde{\omega}^k = \omega^k + \theta^k(\omega^k - \omega^{k-1}). \quad (67)$$

By combining (66) and (67), we obtain the fixed-point iteration in (34), that corresponds to the iteration in [44, Th. 2.1] applied on  $\Phi_C^{-1}T$  and, thus, concludes the proof.

(ii): The following Lemma shows that, if the step sizes are chosen as in Algorithm 6, then the preconditioning matrix  $\Phi_C$  is positive definite.

*Lemma 9:* Let  $\{\alpha_i\}_{i \in \mathcal{I}}$  and  $\beta$  be set as in Algorithm 6, then,  $\Phi_C \succ 0$ .  $\square$

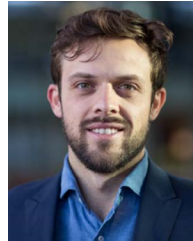
*Proof:* This proof follows the same technical reasoning of the proof of Lemma 8 (i) and, thus, is omitted.  $\blacksquare$

The fixed-point iteration (34), that corresponds to Algorithm 6 by the first part of this proof, is the inertial proximal-point algorithm applied on the operator  $\Phi_C^{-1}T$ . The convergence of (34) to some  $\omega^* = \text{col}(\mathbf{x}^*, \lambda^*) \in \text{zer}(\Phi_C^{-1}T)$  follows by [44, Th. 2.1, Prop. 2.1], since  $\Phi_C^{-1}T$  is maximally monotone in the  $\Phi_C$ -induced norm and  $0 \leq \theta^k \leq \theta^{k+1} \leq \bar{\theta} < 1/3$ , for all  $k > 0$ . To conclude, we note that  $\omega^* \in \text{zer}(\Phi_C^{-1}T) = \text{zer}(T)$ , since  $\Phi_C \succ 0$ , by Lemma 9. Since the limit point  $\omega^* \in \text{zer}(T) \neq \emptyset$ , then  $\mathbf{x}^*$  is a v-GNE of the game in (4), by Proposition 1, concluding the proof.  $\blacksquare$

## REFERENCES

- [1] N. S. Kukushkin, "Best response dynamics in finite games with additive aggregation," *Games Econ. Behav.*, vol. 48, no. 1, pp. 94–10, 2004.
- [2] M. Jensen, "Aggregative games and best-reply potentials," in *Economic Theory*, vol. 43. Berlin, Germany: Springer, 2010, pp. 45–66.
- [3] R. Cornes and R. Hartley, "Fully aggregative games," *Econ. Lett.*, vol. 116, no. 3, pp. 631–633, 2012.
- [4] K. Ma, G. Hu, and C. Spanos, "Distributed energy consumption control via real-time pricing feedback in smart grid," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 5, pp. 1907–1914, Sep. 2014.
- [5] H. Chen, Y. Li, R. Louie, and B. Vucetic, "Autonomous demand side management based on energy consumption scheduling and instantaneous load billing: An aggregative game approach," *IEEE Trans. Smart Grid*, vol. 5, no. 4, pp. 1744–1754, Jul. 2014.
- [6] M. Ye and G. Hu, "Game design and analysis for price-based demand response: An aggregate game approach," *IEEE Trans. Cybern.*, vol. 47, no. 3, pp. 720–730, Mar. 2017.
- [7] Z. Ma, S. Zou, L. Ran, X. Shi, and I. A. Hiskens, "Efficient decentralized coordination of large-scale plug-in electric vehicle charging," *Automatica*, vol. 69, pp. 35–47, 2016.
- [8] Z. Liu, Q. Wu, S. Huang, L. Wang, M. Shahidehpour, and Y. Xue, "Optimal day-ahead charging scheduling of electric vehicles through an aggregative game model-based," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 5173–5184, Sep. 2018.
- [9] P. Zhou, W. Wei, K. Bian, D. O. Wu, Y. Hu, and Q. Wang, "Private and truthful aggregative game for large-scale spectrum sharing," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 2, pp. 463–477, Feb. 2017.
- [10] J. Barrera and A. Garcia, "Dynamic incentives for congestion control," *IEEE Trans. Autom. Control*, vol. 60, no. 2, pp. 299–310, Feb. 2015.
- [11] J. Koshal, A. Nedić, and U. Shanbhag, "Distributed algorithms for aggregative games on graphs," *Operations Res.*, vol. 64, no. 3, pp. 680–704, 2016.
- [12] 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.
- [13] G. Belgioioso and S. Grammatico, "Semi-decentralized Nash equilibrium seeking in aggregative games with coupling constraints and non-differentiable cost functions," *IEEE Control Syst. Lett.*, vol. 1, no. 2, pp. 400–405, Oct. 2017.
- [14] S. Liang, P. Yi, and Y. Hong, "Distributed Nash equilibrium seeking for aggregative games with coupled constraints," *Automatica*, vol. 85, pp. 179–185, 2017.
- [15] D. Paccagnan, B. Gentile, F. Parise, M. Kamgarpour, and J. Lygeros, "Nash and Wardrop equilibria in aggregative games with coupling constraints," *IEEE Trans. Autom. Control*, vol. 64, no. 4, pp. 1373–1388, Apr. 2019.
- [16] Z. Deng and S. Liang, "Distributed algorithms for aggregative games of multiple heterogeneous eulerlagrange systems," *Automatica*, vol. 99, pp. 246–252, 2019.
- [17] C. D. Persis and S. Grammatico, "Continuous-time integral dynamics for a class of aggregative games with coupling constraints," *IEEE Trans. Autom. Control*, vol. 65, no. 5, pp. 2171–2176, May 2020.
- [18] D. Gadjov and L. Pavel, "Single-timescale distributed GNE seeking for aggregative games over networks via forward-backward operator splitting," *IEEE Trans. Autom. Control*, vol. 66, no. 7, pp. 3259–3266, Jul. 2021.
- [19] G. Belgioioso, A. Nedić, and S. Grammatico, "Distributed generalized Nash equilibrium seeking in aggregative games on time-varying networks," *IEEE Trans. Autom. Control*, vol. 66, no. 5, pp. 2061–2075, May 2021.
- [20] P. Yi and L. Pavel, "A distributed primal-dual algorithm for computation of generalized Nash equilibria via operator splitting methods" in *Proc. IEEE 56th Annu. Conf. Decis. Control.*, 2017, pp. 3841–3846.
- [21] P. Yi and L. Pavel, "An operator splitting approach for distributed generalized Nash equilibria computation," *Automatica*, vol. 102, pp. 111–121, 2019.
- [22] 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.
- [23] F. Facchinei and J. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Berlin, Germany: Springer Verlag, 2003.
- [24] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, vol. 408. Berlin, Germany: Springer, 2011.
- [25] S. Grammatico, "Comments on distributed robust adaptive equilibrium computation for generalized convex games" *Automatica*, vol. 97, pp. 186–188, 2018.
- [26] A. Kannan and U. V. Shanbhag, "Distributed computation of equilibria in monotone Nash games via iterative regularization techniques," *SIAM J. Optim.*, vol. 22, no. 4, pp. 1177–1205, 2012.

- [27] P. Yi and L. Pavel, "Distributed generalized Nash equilibria computation of monotone games via double-layer preconditioned proximal-point algorithms," *IEEE Control Netw. Syst.*, vol. 6, no. 1, pp. 299–311, Mar. 2019.
- [28] D. P. Palomar and Y. C. Eldar, *Convex Optimization in Signal Processing and Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2010.
- [29] F. Facchinei and C. Kanzow, "Generalized Nash equilibrium problems," *Ann. Operations Res.*, vol. 175, no. 1, pp. 177–211, 2010.
- [30] L. Deori, K. Margellos, and M. Prandini, "Price of anarchy in electric vehicle charging control games: When Nash equilibria achieve social welfare," *Automatica*, vol. 96, pp. 150–158, 2018.
- [31] A. Auslender and M. Teboulle, "Lagrangian duality and related multiplier methods for variational inequality problems," *SIAM J. Optim.*, vol. 10, no. 4, pp. 1097–1115, 2000.
- [32] P.-E. Maingé, "Convergence theorems for inertial KM-type algorithms," *J. Comput. Appl. Math.*, vol. 219, no. 1, pp. 223–236, 2008.
- [33] D. A. Lorenz and T. Pock, "An inertial forward-backward algorithm for monotone inclusions," *J. Math. Imag. Vis.*, vol. 51, no. 2, pp. 311–325, 2015.
- [34] R. I. Boş, S.-M. Grad, D. Meier, and M. Staudigl, "Inducing strong convergence of trajectories in dynamical systems associated to monotone inclusions with composite structure," *Adv. Nonlinear Anal.*, vol. 10, no. 1, pp. 450–476, 2020.
- [35] R. I. Boş and E. R. Csetnek, "An inertial forward-backward-forward primal-dual splitting algorithm for solving monotone inclusion problems," *Numer. Algorithms*, vol. 71, no. 3, pp. 519–540, 2016.
- [36] Y. Malitsky and M. K. Tam, "A forward-backward splitting method for monotone inclusions without cocoercivity," *SIAM J. Optim.*, vol. 30, no. 2, pp. 1451–1472, 2020.
- [37] T. K. Kristoffersen, K. Capion, and P. Meibom, "Optimal charging of electric drive vehicles in a market environment," *Appl. Energy*, vol. 88, no. 5, pp. 1940–1948, 2011.
- [38] I. Atzeni, L. G. Ordóñez, G. Scutari, D. P. Palomar, and J. R. Fonollosa, "Demand-side management via distributed energy generation and storage optimization," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 866–876, Jun. 2013.
- [39] R. Estrella, G. Belgioioso, and S. Grammatico, "A shrinking-horizon, game-theoretic algorithm for distributed energy generation and storage in the smart grid with wind forecasting," *IFAC-PapersOnLine*, vol. 52, no. 3, pp. 126–131, 2019.
- [40] G. Belgioioso, W. Ananduta, S. Grammatico, and C. Ocampo-Martinez, "Energy management and peer-to-peer trading in future smart grids: A distributed game-theoretic approach," in *Proc. Eur. Control Conf.*, 2020, pp. 1324–1329.
- [41] F. Facchinei, V. Piccialli, and M. Sciarone, "Decomposition algorithms for generalized potential games," *Comput. Optim. Appl.*, vol. 50, no. 2, pp. 237–262, 2011.
- [42] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, vol. 23. Englewood Cliffs, NJ, USA: Prentice-Hall, 1989.
- [43] G. Belgioioso and S. Grammatico, "On convexity and monotonicity in generalized aggregative games," *Proc. IFAC World Congr.*, vol. 50, no. 1, pp. 14 338–14 343, 2017.
- [44] F. Alvarez and H. Attouch, "An inertial proximal method for maximal monotone operators via discretization of a nonlinear oscillator with damping," *Set-Valued Anal.*, vol. 9, no. 1/2, pp. 3–11, 2001.
- [45] D. Davis and W. Yin, "Convergence rate analysis of several splitting schemes," in *Splitting Methods in Communication, Imaging, Science, and Engineering*. Berlin, Germany: Springer, 2016, pp. 115–163.
- [46] Z. Ma, S. Zou, and X. Liu, "A distributed charging coordination for large-scale plug-in electric vehicles considering battery degradation cost," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 5, pp. 2044–2052, Sep. 2015.
- [47] Z. Ma, D. Callaway, and I. Hiskens, "Decentralized charging control of large populations of plug-in electric vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 1, pp. 67–78, Jan. 2013.
- [48] G. Scutari, F. Facchinei, J.-S. Pang, and D. P. Palomar, "Real and complex monotone communication games," *IEEE Trans. Inf. Theory*, vol. 60, no. 7, pp. 4197–4231, Jul. 2014.
- [49] D. G. Feingold *et al.*, "Block diagonally dominant matrices and generalizations of the gerschgorin circle theorem," *Pacific J. Math.*, vol. 12, no. 4, pp. 1241–1250, 1962.



**Giuseppe Belgioioso** was born in Venice, Italy, in 1990. He received the bachelor's degree in information engineering and the master's degree (*summa cum laude*) in automatic control engineering from the University of Padova, Padua, Italy, in 2012 and 2015, respectively, and the Ph.D. degree in automatic control from the Eindhoven University of Technology (TU/e), Eindhoven, The Netherlands, in 2020.

He is a currently Postdoctoral Researcher with the Automatic Control Laboratory (ifa), Federal Institute of Technology (ETH) Zürich, Switzerland. In 2015 and 2016, he was also a Research Scientist (assegnista di ricerca) with the Control Systems Group (Automatica), University of Padova, Italy. In 2019, he visited the School of Electrical, Computer and Energy Engineering at Arizona State University, USA. His research interests include game theory, operator theory and distributed optimization for studying and solving coordination, decision, and control problems that arise in complex network systems.



**Sergio Grammatico** was born in Italy, in 1987. He received the bachelor's degree (*summa cum laude*) in computer engineering, the master's degree (*summa cum laude*) in automatic control engineering, and the Ph.D. degree in automatic control, all from the University of Pisa, Pisa, Italy, in 2008, 2009, and 2013, respectively, and the master's degree (*summa cum laude*) in engineering science from the Sant'Anna School of Advanced Studies, Pisa, Italy, in 2011.

He is currently an Associate Professor with the Delft Center for Systems and Control, Delft University of Technology, The Netherlands.

In 2010–2012, he also visited the Department of Mathematics, University of Hawaii at Manoa, USA, and the Department of Electrical and Computer Engineering, University of California at Santa Barbara, USA. In 2013–2015, he was a Postdoctoral Research Fellow with the Automatic Control Laboratory, ETH Zurich, Switzerland, and in 2015–2017, he was an Assistant Professor with the Department of Electrical Engineering, Control Systems, TU Eindhoven, The Netherlands.

Prof. Grammatico was the recipient of a 2005 F. Severi B.Sc. Scholarship by the Italian High-Mathematics National Institute, and a 2008 M.Sc. Fellowship by the Sant'Anna School of Advanced Studies. He was awarded 2013 and 2014 "TAC Outstanding Reviewer" by the Editorial Board of the IEEE TAC. He was recipient of the Best Paper Award at the 2016 ISDG International Conference on Network Games, Control and Optimization, and of the 2021 Roberto Tempo Best CDC Paper Award. He is currently an Associate Editor for the IEEE TRANSACTIONS ON AUTOMATIC CONTROL and for IFAC *Automatica*.