Equilibrium seeking in games under partial-decision information

Bianchi, M.

DOI
10.4233/uuid:0ef14921-08f4-4f5f-af73-38914755f47f

Publication date
2023

Document Version
Final published version

Citation (APA)
Bianchi, M. (2023). *Equilibrium seeking in games under partial-decision information*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:0ef14921-08f4-4f5f-af73-38914755f47f

EQUILIBRIUM SEEKING IN GAMES UNDER PARTIAL-DECISION INFORMATION

EQUILIBRIUM SEEKING IN GAMES UNDER PARTIAL-DECISION INFORMATION

Dissertation

for the purpose of obtaining the degree of doctor
at the Delft University of Technology,
by the authority of the Rector Magnificus, prof. dr. ir. T.H.J.J. van der Hagen,
chair of the Board for Doctorates,
to be defended publicly on
Friday, 24 February 2023 at 15:00

by

Mattia BIANCHI

Master of Science in Computer and Systems Engineering,
University of L'Aquila, Italy,
born in Guardiagrele, Italy.

This dissertation has been approved by the

promotor: dr. ing. S. Grammatico
copromotor: prof. dr. ir. B. De Schutter

Composition of the doctoral committee:

Rector Magnificus,	chairman
dr. ing. S. Grammatico,	Delft University of Technology, promotor
prof. dr. ir. B. De Schutter,	Delft University of Technology, copromotor
Independent members:	
prof. dr. T. Keviczky,	Delft University of Technology
prof. dr. ir. G.J.T. Leus,	Delft University of Technology
prof. dr. C. De Persis,	University of Groningen
prof. dr. L. Pavel,	University of Toronto
prof. dr. J. Cortés,	University of California San Diego

This research was supported by the Netherlands Organization for Scientific Research (NWO) under research project OMEGA (grant n. 613.001.702) and by the European Research Council under research project COSMOS (grant n. 802348).

ISBN 978-94-6366-654-1

Printed by Gildeprint

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

This dissertation has been completed in fulfillment of the requirements of the Dutch Institute of Systems and Control (DISC) for graduate study.

Millions saw the apple fall, but Newton asked why.

Bernard Baruch

SUMMARY

In modern society, the need has emerged to guarantee the efficient operation of large interconnected systems, such as communication networks, peer-to-peer energy markets, smart cities, social networks. Despite the diverse application domains, a recurring feature is the presence of multiple components, managed by autonomous agents (artificial devices or humans), with shared resources and local, partial knowledge about the whole system. Ensuring a desired emerging behavior requires achieving effective coordination among the agents. This is an arduous task, especially when the agents have diverging interests and objectives, in applications like charging scheduling of electric vehicles, traffic control, bandwidth sharing among self-interested internet users. Mathematically, these competitive settings are modeled by *games*, where the complex interaction among the agents is represented via coupled cost functions and constraints. Often the goal of the agents is to reach an efficient decision, within the operational limits of the system, from which no agent has an incentive to deviate, i.e., a *generalized Nash equilibrium* (GNE).

The main topic of this dissertation is the distributed computation of GNEs in multi-agent games with network structure. In particular, we design and analyze algorithms in the *partial-decision information* scenario (also named fully-distributed algorithms), where each agent can only rely on the information received by some neighbors over a communication graph, although its cost function depends on the actions of possibly all the competitors. This setup is motivated by engineering applications with no central system coordinator, for instance multi-agent autonomous driving or coverage control. While the agents can estimate the unknown variables via local data exchange and consensus protocols, the estimation error introduces critical challenges in the development of algorithms. In fact, the existing schemes for GNE seeking under partial-decision information suffer important limitations, as to performance and conditions to guarantee convergence.

Driven by the need for fast solution methods, in the first part of the thesis we design fully-distributed GNE seeking algorithms based on proximal best-response dynamics – as opposed to known gradient-type iterations. Our convergence analysis is based on an operator-theoretic reformulation and on the choice of a convenient preconditioning; this derivation also facilitates the development of acceleration and inexact schemes, and the customization to the prominent class of aggregative games. Our algorithms outperform the known pseudo-gradient methods, in terms of both communication and (more surprisingly) computation burden. We also study the convergence of our proximal algorithms under a mild assumption, that requires neither Lipschitz continuity nor strong monotonicity of the game mapping.

In control problems arising, e.g., in robotic networks and power systems, the actions of the agents are represented by the outputs of some dynamical systems, to be regulated to an (a priori unknown) GNE. This optimal steady-state problem is addressed in the sec-

ond part of the dissertation. Motivated by mobile sensors applications, we focus on networks of multi-integrator agents; yet, our results can be applied to a wide range of linear or nonlinear systems, via feedback linearization. Specifically, we design fully-distributed continuous-time dynamic feedback controllers that guarantee asymptotic stability of a GNE. The convergence analysis combines stability theory for (projected) dynamical systems, monotonicity properties, and a LaSalle argument to prove convergence of the dual variables. Thanks to the use of adaptive weights in the consensus dynamics, our schemes also allow for totally decentralized tuning.

Finally, in the third part of this thesis, we study complex network structures in the partial-decision information scenario. First, we consider games without coupling constraints, where the agents can only communicate over a time-varying or directed (row stochastic) graph. By leveraging contractivity properties, we develop linearly convergent and fast pseudo-gradient algorithms. Second, we propose a framework to improve efficiency and scalability of fully-distributed GNE seeking algorithms. Such methods generally require each agent to keep (and exchange) an estimate of the actions of all other agents. We propose a graph-theoretic notation that, while introducing virtually no complication in the convergence analysis, allows for more efficient estimate allocations when the coupling among the agents exhibits some sparsity. We demonstrate our framework on generalized games, by designing methods where each agent only estimates some components of the primal/dual variable (or of an aggregation function).

CONTENTS

1

OVERVIEW

The first chapter sells the book; the last chapter sells the next book.

Mickey Spillane

Abandon all hope, ye who enter here

Dante Alighieri (J. Ciardi, Trans.)

The aim of this dissertation is to develop distributed solution methods for multi-agent (generalized) Nash equilibrium problems. In this introductory chapter, we provide the background and motivation of this research. First, we present current challenges in engineering applications (related to communication, traffic and power networks), and we discuss the relevance of game-theoretic models for their undertaking. In particular, we introduce the partial-decision information scenario, which is the focus of this thesis, highlighting some open problems of the field. Then, we formalize our research objectives. Finally, we outline the organization of the remaining chapters.

1

1.1. MULTI-AGENT NETWORKS

POWER grid management, energy market regulation, communication systems, cybersecurity: many contemporary technological applications involve large networks,[1] composed of physical infrastructures and of multiple *agents*, namely entities (processing units or human users) with decision-making capabilities. While these agents are autonomous, their decision processes are strictly interdependent and ultimately determine the functioning of the complex system. Ensuring a desired emerging behavior requires control actions and effective multi-agent coordination mechanisms, able to capture the increasing complexity of engineering, social and economics systems.

For instance, in *traffic networks*, reducing congestion and air pollutant emissions entails dealing with the large number of vehicles that share the road space-time. The traditional approach consists of tolling policies, implemented by a road authority to incentivize the independent drivers to choose system-optimal routes [97], [134]. Meanwhile, novel opportunities for sustainable mobility are offered by the proliferation of advanced technologies [39], [142]. Above all, autonomous cars and intelligent transportation systems (where vehicles can exchange information with close-by vehicles or with roadside modules) hold the promise to greatly reduce congestion and energy consumptions – subject to the deployment of adequate coordination strategies – for example by avoiding accidents and phantom traffic jams.

Multi-agent applications also arise in *cognitive radio* systems [1], [141], which are designed to fully exploit the available spectrum in wireless communication networks – a precious resource, in the "internet of things" era. In essence, the users (transceivers) must adaptively select their operating parameters by interacting with other users and with the environment, e.g., picking vacant channels to avoid interference. Design of robust spectrum sharing protocols is an arduous task, due to unreliability of wireless communication, users' mobility and time-varying service requests. In fact, conjugating responsiveness and spectral efficiency remains an open issue.

A third prominent coordination problem stems from the power system realm. Owing to the desire for cheap, reliable and green energy, national grids are progressively incorporating more microgrids – local power networks with storage and generation facilities, operated autonomously to efficiently provide electricity to the local users, often relying on renewable resources (solar panels, wind turbines). The resulting *smart grid* is foreseen as a complex cyber-physical system, where the microgrids communicate, trade energy and interact between them and with the main grid [129], [139].

The examples above highlight some crucial features of emerging network architectures. These applications involve a sizeable number of agents (e.g., drivers, transceivers, microgrid managers), each having only access to limited, local knowledge about the overall system. Promptly gathering global information might be prohibitive, because of bandwidth constraints and communication costs, or because the agents are reluctant to share private data. Besides, real-time centralized processing of large amounts of data is impractical, especially in dynamic, heterogeneous environments (like cognitive radio networks). Altogether, achieving scalable, robust, privacy-preserving decision-making demands the design of *distributed methods* – where each agent's decision is based only

[1] A network is a a group or system of interconnected people or things [131].

on the incomplete knowledge available locally (e.g., received by some other neighboring agent), and the computational burden is partitioned among the agents.

Another fundamental trait of complex network systems is the presence of *self-interested agents*, that pursue their own individual objectives (e.g., minimizing the travel time, maximizing the quality of service, minimizing the electricity bill), disregarding the intents of the other decision makers or the social welfare. While an agent has only authority over its own choices (e.g., route, channel, amount of energy traded with a grid), its objective often depends on the actions of the other agents – for instance due to shared resources (e.g., road, spectrum, power grid) or market dynamics. Accordingly, the agents may still coordinate or compromise, to advance their own goals and avoid detrimental outcomes (e.g., traffic jams) – although their distinct, possibly conflicting, interests prevent a fully cooperative behavior.

1.2. GAME THEORY FOR COORDINATION

NETWORKS of rational, self-interested entities, as those previously introduced, can be mathematically described by *generalized games*. A game is a collection of interdependent optimization problems, one for each agent (decision maker/player) in a system. Each agent has a set of available decisions (actions/strategies),[2] and an individual cost function depending both on its own decision and on the decisions of (some of) the other players. A game is called generalized if the agents' actions are further coupled via shared constraints (e.g., the total quantity of a resource used by all the agents is bounded by the resource availability).

Game theory was conceived in the field of mathematical economics [42], [100], as a modeling paradigm for markets of competitive, noncooperative firms (without external enforcement or coordination). Over the years, it was also recognized across diverse research fields and areas of engineering, as a crucial framework to *control* the interaction between self-interested agents. Indeed, game theory provides the tools not only to modulate the emerging behavior of a network via intervention strategies (as in incentive mechanism design [61], [132]), but also to shape the actual decision-making process of the agents, to guarantee the attainment of a desirable decision. The theme of this dissertation is the latter task. Specifically, the challenge is to design the local interactions between the agents to ensure the achievement of a stable, safe and efficient operating point, most often identified with a *generalized Nash equilibrium* (GNE) [55]. The dynamic, iterative processes employed by the agents to reach a GNE are named GNE seeking algorithms.

GNE seeking has been employed in a variety of topical application domains, including competition in markets [88], power systems [119], wireless communication [72], wired communication [110], multi-agent machine learning [45], blockchain [80]. Naturally, the networked structure of these problems calls for distributed solutions.

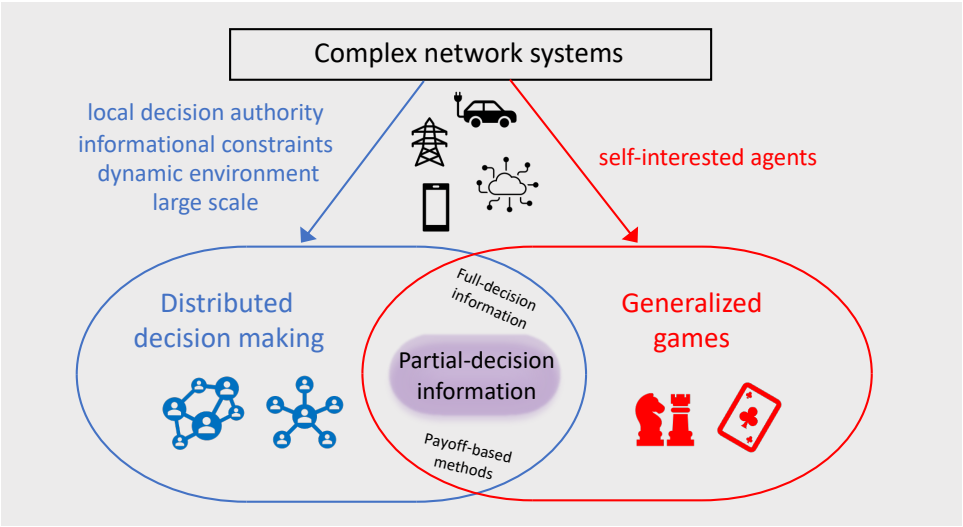

Figure 1.1: GNE seeking under partial-decision information conciliates distributed operation with GNE computation. In this setting, an agent cannot access all the actions that affect its objective (differently from the canonical full-decision information scenario [55]), nor measure its own cost function (in contrast to what assumed for zeroth-order learning, e.g., extremum seeking [84], [130]).

1.3. GNE SEEKING UNDER PARTIAL-DECISION INFORMATION

DISTRIBUTED solution of *GNE problem*s (GNEPs) is a thriving research topic; countless algorithms have been studied, that rest on different informational constraints (i.e., assumptions on the knowledge that players can access). The emphasis of this thesis is on network applications where there is no coordinator that gathers and broadcasts data over the system, and where the agents can only rely on fragmentary peer-to-peer interaction: cognitive radio [141], OSNR optimization in optical networks [122], demand response in the smart grid [147], formation control [93], to mention a few. Such center-free systems are especially empowered by the current deregulation and authority decentralization trends in markets, telecommunications and transportation industries.

Specifically, we focus on the *partial-decision information* model, where (i) an agent can only obtain new data by communicating with some neighboring agents; (ii) each agent holds an analytical expression of a private cost function, which possibly depends on the (unknown) actions of non-neighboring agents. To compensate for the lack of global knowledge, the agents keep an estimate of the unknown quantities, and engage in information sharing with their neighbors, combining the local decision processes with consensus dynamics. This setting has only been introduced very recently, in the seminal work [83]; yet, it has attracted since then considerable scientific interest [108], [109], [136], [146], due to its prospect engineering applications as well as fresh theoretical challenges. To be precise, the major technical difficulty is that the operator-theoretic properties of the game mapping (a crucial operator, characterizing the game) are not preserved,

[2] In this thesis, we consider *continuous-action* games, where the decision set of each agent is uncountably infinite (e.g., a convex set); we do not consider finite-action games, where the decision set is finite.

when moving from the decision space to the lifted space of the estimates.

In fact, in spite of the many proposals, the current state-of-the-art GNE seeking algorithms in the partial-decision information scenario suffer severe drawbacks (in terms of communication cost, computation burden and conditions required to ensure convergence), which significantly hinder their applicability. In particular, we individuate some key pitfalls of the existing methods as follows:

1. *Convergence rate:* a large number of iterations/communications is required for convergence (resulting in time/energy inefficiency).

2. *Scalability:* each agent is required to keep and exchange with neighbors an estimate of the actions of all other players (except for the class of aggregative games), which might be impractical if the agent population size is large; furthermore, the convergence rate deteriorates rapidly when the number of agents grows.

3. *Monotonicity:* converge is guaranteed only under quite restrictive monotonicity assumptions on the game mapping.

4. *Smoothness:* converge is typically proven only if the game mapping is Lipschitz continuous.

5. *Communication topology:* the communication network is required to be undirected and connected (except for gossip algorithms), while there is no method to address complex topologies (time-varying, directed, lossy or delayed networks).

6. *Dynamical agents:* known algorithms to address games with coupling constraints assume that the agents can freely switch among their strategies, which is limiting for many control problems, where the agents' actions are represented by the outputs of some dynamical systems.

Towards safe and efficient decision-making in center-free networks of self-interest agents, in this thesis, we intend to probe the above issues, as summarized in the next section.

1.4. RESEARCH GOALS

THE aim of this PhD dissertation is to develop a mathematical theory and computational algorithms to efficiently solve GNE problems in the partial-decision information scenario, in the presence of complex network interaction and dynamic coupling. We approach this objective by investigating the following questions:

Q1. How to achieve fast, scalable, communication-efficient, fully-distributed GNE seeking?

Q2. What methods can be employed to relax monotonicity and smoothness requirements in GNE seeking under partial-decision information?

Q3. How to design distributed controllers to solve GNE problems in the presence of dynamical agents?

Q4. How to analyze convergence to an equilibrium in games played over directed, time-varying communication networks?

We assess in the concluding Chapter 8 to what extent the results in this thesis contribute to answer each of the questions above.

1.5. THESIS ORGANIZATION

FIGURE 1.2 presents the outline of the thesis and the connections of the chapters. The contents of each chapter are summarized next.

The basic notation used throughout the thesis is introduced in **Appendix A**. We review definitions and results from graph theory in **Appendix B**, and we present some background material on operator theory and fixed-point iterations in **Appendix C**: these are the chief mathematical tools employed in the dissertation, which we integrate in the design of GNE seeking algorithms.

The rest of the thesis is composed of four parts.

PART I: FULLY-DISTRIBUTED PROXIMAL-POINT ALGORITHMS

Driven by the need for faster solution methods, in this part of the thesis, we develop GNE seeking algorithms based on proximal best-response dynamics. We compare these methods with state-of-the-art gradient-type iterations, in terms of speed and communication efficiency (Chapter 2) and of monotonicity and smoothness conditions required to ensure convergence (Chapter 3).

- **Chapter 2** (addressing Q1)
  In this chapter, we derive a fully-distributed[3] primal-dual proximal algorithm, to solve games with coupling constraints, by leveraging an operator-theoretic approach and devising a suitable preconditioning. We illustrate how, by allowing for larger step sizes, our scheme outperforms existing gradient-type methods. Additionally, we tailor the developed algorithm for the important class of aggregative games, and we analyze three accelerations strategies, that can further improve convergence speed.

  This chapter is partially based on the following publication:

  [22] M. Bianchi, G. Belgioioso, and S. Grammatico, "Fast generalized Nash equilibrium seeking under partial-decision information," *Automatica*, vol. 136, p. 110 080, 2022. DOI: 10.1016/j.automatica.2021.110080.

- **Chapter 3** (addressing Q2)
  We relate several monotonicity and smoothness assumptions, used in literature for the partial-decision information scenario. Then, we prove convergence of the proximal algorithm developed in Chapter 2 but under a weaker condition, namely a restricted monotonicity property that does not require the game mapping to be strongly monotone nor (Lipschitz) continuous.

[3] Namely, not requiring the presence of a central node; not to be confused with *distributed* algorithms devised for games in full-decision information, where the computation is partitioned among the agents, but generally requiring a coordinator to broadcast the actions of the agents over the network. In this thesis, with *fully-distributed* algorithms, we refer to algorithms for the partial-decision information scenario.

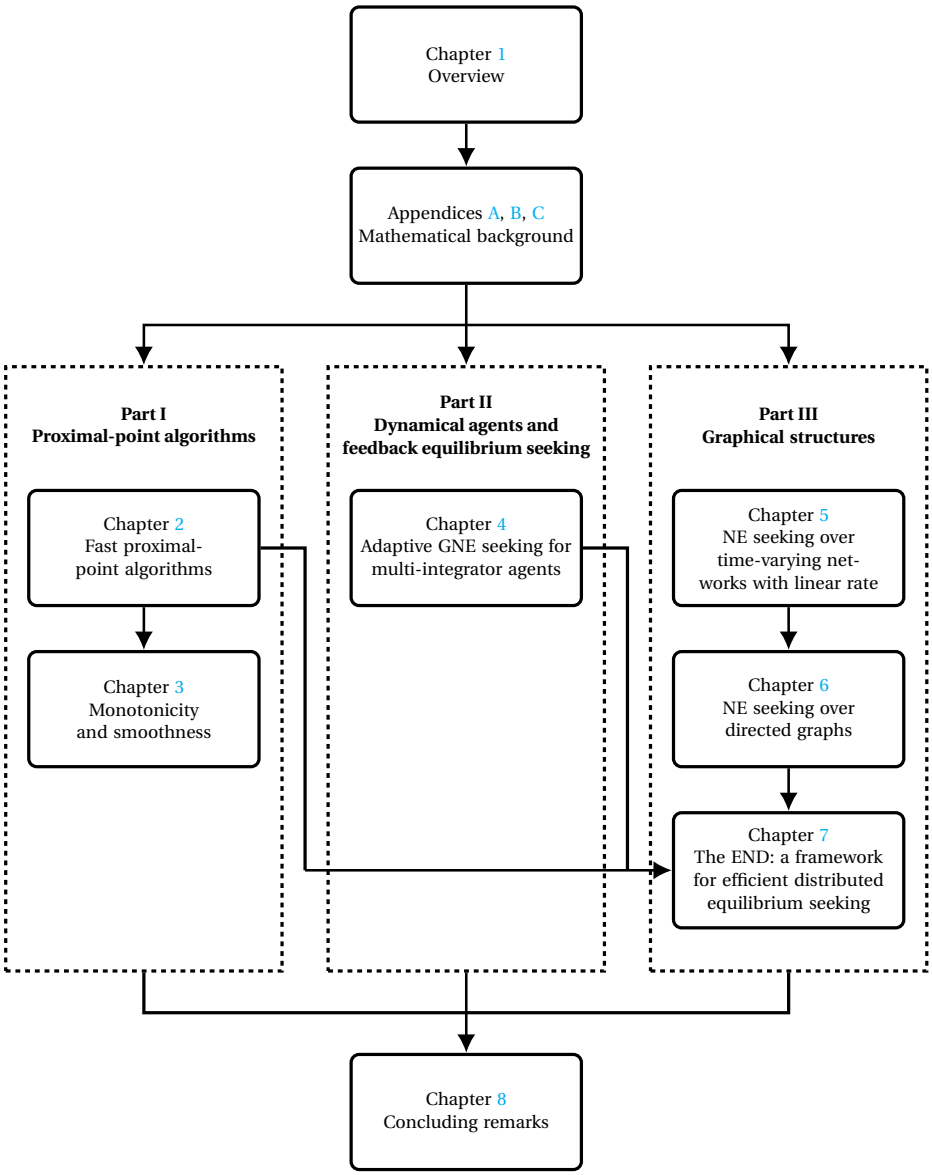

Figure 1.2: Structure of the thesis. Arrows indicate *read-before* relations.

This chapter is partially based on the following paper:

[26] M. Bianchi and S. Grammatico, "Nash equilibrium seeking under partial decision information: Monotonicity, smoothness and proximal-point algorithms," in *2022 61st IEEE Conference on Decision and Control (CDC)*, 2022, pp. 5080–5085. DOI: 10.1109/CDC51059.2022.9993145.

PART II: FEEDBACK EQUILIBRIUM SEEKING FOR DYNAMICAL AGENTS

This part of the thesis is devoted to the design of continuous-time distributed controllers to drive the outputs of some dynamical systems, associated with the agents in a game, to an (a priori unknown) GNE.

- **Chapter 4** (addressing Q3, Q2)
  Motivated by robotic applications and by feedback linearization of nonlinear systems, in this chapter we focus on generalized games played by multi-integrator agents. We propose a fully-distributed controller that guarantees asymptotic stability of a GNE, by resorting to a LaSalle argument and to the monotonicity properties of the game. Furthermore, we design uncoordinated adaptive weights for the consensus of the estimates, which allow for a totally decentralized tuning.

  This chapter is partially based on the following publication:

  [24] M. Bianchi and S. Grammatico, "Continuous-time fully distributed generalized Nash equilibrium seeking for multi-integrator agents," *Automatica*, vol. 129, p. 109 660, 2021. DOI: 10.1016/j.automatica.2021.109660.

PART III: GRAPHICAL STRUCTURES IN GAMES UNDER PARTIAL-DECISION INFORMATION

In this part of the thesis, we study games in the partial-decision information scenario under the lens of graph theory. Specifically, we exploit graph theoretic properties to develop fully-distributed *Nash equilibrium* (NE) seeking algorithms (for games without coupling constraints) supported by directed and time-varying graphs (Chapters 5 to 7) and to improve efficiency of GNE seeking methods, by leveraging the inherent partial coupling of the problem at hand (Chapter 7).

- **Chapter 5** (addressing Q1, Q4)
  We study a simple algorithm to seek a NE over time-varying doubly stochastic graphs. Linear convergence is proven by integrating contractivity and monotonicity properties over two complementary subspaces. Compared to existing approaches, this procedure ensures bounds on the step sizes that do not diminish when the network size grows, hence superior scalability. Moreover, we analyze a different algorithm, guaranteed to converge on time-varying balanced digraphs.

  This chapter is partially based on the following publication:

  [25] M. Bianchi and S. Grammatico, "Fully distributed Nash equilibrium seeking over time-varying communication networks with linear convergence rate," *IEEE Control Systems Letters*, vol. 5, pp. 499–504, 2021. DOI: 10.1109/LCSYS.2020.3002734.

- **Chapter 6** (addressing Q4)
  In this chapter, we solve for the first time NE problems over a directed row stochastic communication graph, by employing either a small-enough step or by means of vanishing step sizes. The algorithm we design requires the knowledge of the Perron-Frobenius eigenvector of the network, but we show that the procedure can be adapted to compute this eigenvector online.

  This chapter is partially based on the following publication:

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

- **Chapter 7** (addressing Q1, Q4)
  Solving multi-agent decision problems via fully-distributed algorithms requires each agent to store a copy of some variables (e.g., decision or dual variable, aggregation value), often leading to poor scalability and communication overhead. In this chapter, we develop a graph-theoretic design language that enables exploiting the inherent sparsity of a problem to improve efficiency. We demonstrate the framework on generalized games in partial-decision information, by developing methods where each agent only has to estimate some components of primal/dual variables (or of an aggregation function). We also propose applications in the related field of consensus optimization.

  This chapter is partially based on the following paper:

  [28] M. Bianchi and S. Grammatico, "The END: Estimation Network Design for efficient distributed equilibrium seeking," *IEEE Transactions on Automatic Control*, under review. [Online]. Available: https://arxiv.org/abs/2208.11377.

PART IV: CONCLUSION

Finally, **Chapter 8** draws some concluding remarks regarding the main contributions of the thesis and highlights directions for future research.

A complete list of articles published by the author during his doctoral studies, including those not featured in this dissertation, is provided at the end of the thesis.

I

FULLY-DISTRIBUTED PROXIMAL-POINT ALGORITHMS

2

FAST PROXIMAL-POINT ALGORITHMS

If I have seen further, it is by standing on the shoulders of giants.

Isaac Newton

In mathematics, you don't understand things. You just get used to them.

Jhon von Neuman

The few existing methods for GNE seeking under partial-decision information build on projected pseudo-gradient dynamics, and require either double-layer iterations or conservative conditions on the step sizes. To overcome both these flaws and improve efficiency, we design the first fully-distributed single-layer algorithms based on proximal best-response. Our schemes are fixed-step and allow for inexact updates, which is crucial for reducing the computational complexity. Under standard assumptions on the game primitives, we establish convergence to a variational equilibrium (with linear rate for games without coupling constraints) by recasting our algorithms as proximal-point methods, opportunely preconditioned to distribute the computation among the agents. Since our analysis hinges on a restricted monotonicity property, we also provide new general results that significantly extend the domain of applicability of proximal-point methods. Besides, our operator-theoretic approach favors the implementation of provably correct acceleration schemes that can further improve the convergence speed. Finally, the potential of our algorithms is demonstrated numerically, revealing much faster convergence with respect to projected pseudo-gradient methods and validating our theoretical findings.

Parts of this chapter have been published in [22].

2.1. INTRODUCTION

GENERALIZED games model the interaction between self-interested decision makers, or agents, that aim at optimizing their individual, yet inter-dependent, objective functions, subject to shared constraints. This competitive scenario has received increasing attention with the spreading of networked systems, due to the numerous engineering applications, including demand-side management in the smart grid [119], charging/discharging of electric vehicles [71], demand response in competitive markets [88], and radio communication [57]. From a game-theoretic perspective, the challenge is to assign the agents behavioral rules that eventually ensure the attainment of a satisfactory equilibrium.

A recent part of the literature focuses in fact on designing *distributed* algorithms to seek a GNE, a decision set from which no agent has interest to unilaterally deviate [13], [38], [55], [149], [151]. In these works, the computational effort is partitioned among the agents, but assuming that each of them has access to the decision of all the competitors (or to an aggregation value, in the case of aggregative games). Such an hypothesis, referred to as *full-decision information*, generally requires the presence of a central coordinator that communicates with all the agents, which is impractical in some cases [133], [60]. One example is the Nash–Cournot competition model described in [82], where the profit of each of a group of firms depends not only on its own production, but also on the total supply, a quantity not directly accessible by any of the firms. Instead, here we consider the so-called *partial-decision information* scenario, where each agent estimates the actions of all the competitors by relying only on the information exchanged with some neighbors over a communication network. Thus, the goal is to design *fully-distributed* (namely, center-free) algorithms, based exclusively on peer-to-peer communication.

The partial-decision information setup has only been introduced very recently. Most results consider non-generalized games (i.e., games without shared constraints) [82], [122], [136], [120]. Even fewer algorithms can cope with the presence of coupling constraints [109], [10], [64], despite this extension arises naturally in most resource allocation problems [55, §2], e.g., due to shared capacity limitations. All the cited formulations resort to (projected) gradient and consensus-type dynamics, and are single-layer (i.e., they require a fixed finite number of communications per iteration). The main drawback is that, due to the partial-decision information assumption, theoretical guarantees are obtained only for small (or vanishing) step sizes, which significantly affect the speed of convergence. The only alternative available in literature consists of double-layer algorithms, [86], [108], where the agents must communicate multiple (virtually infinite) times to reach consensus, before each update. An extensive communication requirement is however a performance bottleneck, as the communication time can overwhelm the time spent on local useful processing – in fact, this is a common problem in parallel computing [78]. Let alone the time lost in the transmission, sending large volumes of data on wireless networks results in a dramatically increased energetic cost.

Contributions: To improve speed and efficiency, we design the first fully-distributed single-layer GNE seeking algorithms based on proximal best-response. For the sake of generality and mathematical elegance, we take here an operator-theoretic approach [9], [149], and reformulate the GNE problem as that of finding a zero of a monotone operator. The advantage is that several fixed-point iterations are known to solve monotone

inclusions [8, §26], thus providing a unifying framework to design algorithms and study their convergence. For instance, the methods in [109], [10], [64], were developed based on the (preconditioned) *forward-backward* (FB) splitting [8, §26.5]. To enhance the convergence speed, we instead employ a *proximal-point algorithm* (PPA) [8, Th. 28.1], which typically can tolerate much larger step sizes. Nonetheless, the design of distributed GNE seeking PPAs was elusive until now, because a direct implementation results in double-layer algorithms [124], [148]. The novelties of this chapter are summarized as follows:

- We propose the first PPA to compute a zero of a *restricted* monotone operator, which significantly generalizes classical results for maximally monotone operators. Differently from other recent extensions [52], [99], we also allow for set-valued resolvents and inexact updates, and we do not assume pseudomonotonicity or hypomonotonicity. This is a fundamental result of independent interest, which we exploit to prove convergence of our algorithms (2.4.2);
- We introduce a novel primal-dual proximal best-response GNE seeking algorithm, which is the first non-gradient-based scheme for the partial-decision information setup. We derive our method as a PPA, where we design a novel preconditioning matrix to distribute the computation and obtain a single-layer iteration. Under strong monotonicity and Lipschitz continuity of the game mapping, we prove global convergence with fixed step sizes, by exploiting restricted monotonicity properties. Convergence is retained even if the proximal best-response is computed inexactly (with summable errors), which is crucial for practical implementation. Differently from [109, Alg. 1], the step sizes can be chosen independently of a certain restricted strong monotonicity constant. In turn, not only we allow for much larger steps, but parametric dependence is also improved: for instance, the bounds do not vanish when the number of agents grows, and the resulting convergence rate for non-generalized games is superior. Moreover our scheme requires only one communication per iteration, instead of two (§2.4.3, §2.5.1);
- We apply some acceleration schemes [77] to our *preconditioned PPA* (PPPA) and provide new theoretical convergence guarantees. We observe numerically that the iterations needed to converge can be halved (§2.5);
- We tailor our method to efficiently solve aggregative games, by letting each agent keep and exchange an estimate of the aggregative value only, instead of an estimate of all the other agents' actions (§2.6);
- Via numerical simulations, we show that our PPPAs significantly outperform the pseudo-gradient methods in [64], [109] (the only other known fully-distributed, single-layer, fixed-step GNE seeking schemes), not only in terms of number of iterations needed to converge (hence with a considerable reduction of the communication burden), but also in terms of total computational cost (despite each agent must locally solve a strongly convex optimization problem, rather than a projection, at each step) (§2.7).

To improve readability, some of the proofs are in the chapter appendix. We refer to Appendices A, B, C for the basic notation and mathematical background.

2.2. MATHEMATICAL SETUP

We consider a set of agents, $\mathcal{I} := \{1, \ldots, N\}$, where each agent $i \in \mathcal{I}$ shall choose its decision variable (i.e., strategy) x_i from its local decision set $\Omega_i \subseteq \mathbb{R}^{n_i}$. Let $x := \mathrm{col}((x_i)_{i\in\mathcal{I}}) \in \Omega$ denote the stacked vector of all the agents' decisions, $\Omega := \Omega_1 \times \cdots \times \Omega_N \subseteq \mathbb{R}^n$ the overall action space and $n := \sum_{i=1}^N n_i$. The goal of each agent $i \in \mathcal{I}$ is to minimize its objective function $J_i(x_i, x_{-i})$, which depends on both the local variable x_i and on the decision variables of the other agents $x_{-i} := \mathrm{col}((x_j)_{j\in\mathcal{I}\setminus\{i\}})$. Furthermore, the feasible decisions of each agent depends on the action of the other agents via coupling constraints, which we assume affine: most of the literature focuses on this case [108], [10], which in fact accounts for the vast majority of practical applications [55, §3.2]. Specifically, the overall feasible set is $\mathcal{X} := \Omega \cap \{x \in \mathbb{R}^n \mid Ax \leq b\}$, where $A := [A_1, \ldots, A_N]$ and $b := \sum_{i=1}^N b_i$, $A_i \in \mathbb{R}^{m\times n_i}$ and $b_i \in \mathbb{R}^m$ being local data. The game is then represented by the inter-dependent optimization problems:

$$\forall i \in \mathcal{I}: \min_{y_i \in \mathbb{R}^{n_i}} J_i(y_i, x_{-i}) \quad \text{s.t. } (y_i, x_{-i}) \in \mathcal{X}. \tag{2.1}$$

The technical problem we consider here is the computation of a GNE, namely a set of decisions that simultaneously solve all the optimization problems in (2.1).

Definition 2.1. A collective strategy $x^* = \mathrm{col}\left((x_i^*)_{i\in\mathcal{I}}\right)$ is a generalized Nash equilibrium if, for all $i \in \mathcal{I}$, $J_i\left(x_i^*, x_{-i}^*\right) \leq \inf\{J_i\left(y_i, x_{-i}^*\right) \mid (y_i, x_{-i}^*) \in \mathcal{X}\}$. □

Next, we postulate some common regularity and convexity assumptions for the constraint sets and cost functions, as in, e.g., [82, Asm. 1], [109, Asm. 1].

Standing Assumption 2.1. For each $i \in \mathcal{I}$, the set Ω_i is closed and convex; $\mathcal{X}$ is nonempty and satisfies Slater's constraint qualification[1]; J_i is continuous and $J_i(\cdot, x_{-i})$ is convex and continuously differentiable for every x_{-i}. □

As per standard practice [108], [149], among all the possible GNEs, we focus on the subclass of *variational GNE*s (v-GNEs) [55, Def. 3.11], which are more economically justifiable, as well as computationally tractable [85]. The v-GNEs are so called because they coincide with the solutions to the variational inequality $\mathrm{VI}(F, \mathcal{X})$, where F is the *pseudo-gradient* mapping of the game:

$$F(x) := \mathrm{col}\left((\nabla_{x_i} J_i(x_i, x_{-i}))_{i\in\mathcal{I}}\right). \tag{2.2}$$

Under Standing Assumption 2.1, x^* is a v-GNE of the game in (2.1) if and only if there exists a dual variable $\lambda^* \in \mathbb{R}^m$ such that the following *Karush–Kuhn–Tucker* (KKT) conditions are satisfied [55, Th. 4.8]:

$$\begin{aligned} \mathbf{0}_n &\in F\left(x^*\right) + A^\top \lambda^* + \mathrm{N}_\Omega\left(x^*\right) \\ \mathbf{0}_m &\in -\left(Ax^* - b\right) + \mathrm{N}_{\mathbb{R}^m_{\geq 0}}\left(\lambda^*\right). \end{aligned} \tag{2.3}$$

[1] Namely, there exists a point $\bar{x} \in \mathcal{X}$ in the relative interior of Ω, and such that all the non-affine coupling constraints are strictly satisfied at $\bar{x}$ (in this chapter, we only consider affine coupling constraints). This condition ensures that strong duality holds for the optimization problems in (2.1).

Standing Assumption 2.2. The pseudo-gradient mapping F in (2.2) is μ-strongly monotone and θ_0-Lipschitz continuous, for some $\mu, \theta_0 > 0$. □

The strong monotonicity of F is sufficient to ensure existence and uniqueness of a v-GNE [56, Th. 2.3.3]; it was always assumed for GNE seeking under partial-decision information with fixed step sizes [136, Asm. 2], [109, Asm. 3] (while it is sometimes replaced by strict monotonicity or cocoercivity, under vanishing steps and compactness of $\mathcal{X}$ [82, Asm. 2], [106, Asm. 3] [10, Asm. 5]).

2.3. FULLY-DISTRIBUTED EQUILIBRIUM SEEKING

In this section, we present our baseline algorithm to seek a v-GNE of the game in (2.1) in a fully-distributed way. Specifically, each agent i only knows its own cost function J_i and feasible set Ω_i, and the portion of the coupling constraints (A_i, b_i). Moreover, agent i does not have full knowledge of x_{-i}, and only relies on the information exchanged locally with some neighbors over an undirected communication network $\mathcal{G}(\mathcal{I}, \mathcal{E})$. The pairs $(i,j), (j,i)$ belong to the set of edges $\mathcal{E}$ if and only if agent i and j can mutually exchange information. We denote: $W = [w_{i,j}]_{i,j\in\mathcal{I}} \in \mathbb{R}^{N\times N}$ the symmetric weight matrix of $\mathcal{G}$, with $w_{i,j} > 0$ if $(i,j) \in \mathcal{E}$, $w_{i,j} = 0$ otherwise, and the convention $w_{ii} = 0$ for all $i \in \mathcal{I}$; $L := D - W$ the Laplacian matrix of $\mathcal{G}$, with degree matrix $D := \operatorname{diag}((d_i)_{i\in\mathcal{I}})$, and $d_i := \deg(i) = \sum_{j=1}^{N} w_{i,j}$ for all $i \in \mathcal{I}$; $\mathcal{N}_i = \{j \mid (i,j) \in \mathcal{E}\}$ the set of neighbors of agent i. Moreover, let $V \in \mathbb{R}^{E\times N}$ the weighted incidence matrix of $\mathcal{G}$, where $2E$ is the cardinality of $\mathcal{E}$ (see Appendix B). It holds that $L = V^\top V$; moreover, $\operatorname{null}(V) = \operatorname{null}(L) = \{\kappa \mathbf{1}_N, \kappa \in \mathbb{R}\}$ under the following connectedness assumption [70, Ch. 8].

Standing Assumption 2.3. The communication graph $\mathcal{G}(\mathcal{I}, \mathcal{E})$ is undirected and connected. The weight matrix W is symmetric. □

In the partial-decision information, to cope with the lack of knowledge, each agent keeps an estimate of all other agents' actions [146], [136], [109]. We denote $\boldsymbol{x}_i := \operatorname{col}((\boldsymbol{x}_{i,j})_{j\in\mathcal{I}}) \in \mathbb{R}^n$, where $\boldsymbol{x}_{i,i} := x_i$ and $\boldsymbol{x}_{i,j}$ is agent i's estimate of agent j's action, for all $j \neq i$; let also $\boldsymbol{x}_{j,-i} := \operatorname{col}((\boldsymbol{x}_{j,\ell})_{\ell\in\mathcal{I}\setminus\{i\}})$. Moreover, we let each agent keep an estimate $\lambda_i \in \mathbb{R}^m_{\geq 0}$ of the dual variable, and an auxiliary variable $z_i \in \mathbb{R}^m$.

Our proposed dynamics are summarized in Algorithm 2.1, where the global parameter $\alpha > 0$ and the positive step sizes τ_i, δ_i, $\nu_{(i,j)} = \nu_{(j,i)}$, for all $i \in \mathcal{I}$ and $(i,j) \in \mathcal{E}$, have to be chosen appropriately (see §2.4). Each agent i updates its action x_i similarly to a proximal best-response, but with two extra terms that are meant to penalize and correct the disagreement among the estimates and the coupling constraints violation. Most importantly, the agents evaluate their cost functions in their local estimates, not on the actual collective strategy. In steady state, the agents should agree on their estimates, i.e., $\boldsymbol{x}_i = \boldsymbol{x}_j$, $\lambda_i = \lambda_j$, for all $i, j \in \mathcal{I}$. This motivates the presence of consensus terms for both primal and dual variables. From a control-theoretic perspective, the updates of each z_i can be seen as integrator dynamics driven by the disagreement of the variables λ_j's. This integral action is meant to permit the distributed asymptotic satisfaction of the coupling constraints, despite the computation of each λ_i only involves the local block (A_i, b_i) – differently from typical centralized dual ascent iterations. We postpone a formal derivation of Algorithm 2.1 to §2.4.

Algorithm 2.1. Fully-distributed v-GNE seeking via PPPA

Initialization

- For all $i \in \mathcal{I}$, set $x_i^0 \in \Omega_i$, $\boldsymbol{x}_{i,-i}^0 \in \mathbb{R}^{n-n_i}$, $z_i^0 = \mathbf{0}_m$, $\lambda_i^0 \in \mathbb{R}^m_{\geq 0}$.

For all $k \in \mathbb{N}$:

- Communication: The agents exchange the variables $\{x_i^k, \boldsymbol{x}_{i,-i}^k, \lambda_i^k\}$ with their neighbors.
- Local variables update: each agent $i \in \mathcal{I}$ computes

$$\begin{aligned}
\boldsymbol{x}_{i,-i}^{k+1} &= \tfrac{1}{1+\tau_i d_i}\big(\boldsymbol{x}_{i,-i}^k + \tau_i \textstyle\sum_{j\in\mathcal{N}_i} w_{i,j}\boldsymbol{x}_{j,-i}^k\big)\\
x_i^{k+1} &= \operatorname*{argmin}_{y\in\Omega_i}\Big(J_i(y,\boldsymbol{x}_{i,-i}^{k+1}) + \tfrac{1}{2\alpha\tau_i}\big\|y - x_i^k\big\|^2 + \tfrac{d_i}{2\alpha}\big\|y - \tfrac{1}{d_i}\textstyle\sum_{j\in\mathcal{N}_i} w_{i,j}\boldsymbol{x}_{j,i}^k\big\|^2 + \tfrac{1}{\alpha}(A_i^\top\lambda_i^k)^\top y\Big)\\
z_i^{k+1} &= z_i^k + \textstyle\sum_{j\in\mathcal{N}_i} \nu_{(i,j)} w_{i,j}(\lambda_i^k - \lambda_j^k)\\
\lambda_i^{k+1} &= \operatorname{proj}_{\mathbb{R}^m_{\geq 0}}\big(\lambda_i^k + \delta_i\big(A_i(2x_i^{k+1} - x_i^k) - b_i - (2z_i^{k+1} - z_i^k)\big)\big).
\end{aligned}$$

Remark 2.1. The functions $J_i(\cdot, \boldsymbol{x}_{i,-i})$ are strongly convex, for all $\boldsymbol{x}_{i,-i}$, $i \in \mathcal{I}$, as a consequence of Standing Assumption 2.2. Hence, the argmin operator in Algorithm 2.1 is single-valued, and the algorithm is well defined. □

Remark 2.2. In Algorithm 2.1, each agent has to *locally* solve an optimization problem, at every iteration. Not only these subproblems are fully-decentralized (i.e., they do not require extra communication), but they are also of low dimension (n_i). This is a major departure from the procedure proposed in the PPAs [124, Alg. 2], [148, Alg. 2], where the agents have to collaboratively solve a subgame (of dimension n) before each update. □

2.4. CONVERGENCE ANALYSIS

2.4.1. DEFINITIONS AND PRELIMINARY RESULTS

We denote $\boldsymbol{x} := \operatorname{col}((\boldsymbol{x}_i)_{i\in\mathcal{I}}) \in \mathbb{R}^{Nn}$. Besides, let us define, as in [109, Eq. 13, 14], for all $i \in \mathcal{I}$,

$$\mathcal{R}_i := \begin{bmatrix}\mathbf{0}_{n_i\times n_{<i}} & I_{n_i} & \mathbf{0}_{n_i\times n_{>i}}\end{bmatrix} \in \mathbb{R}^{n_i\times n}, \tag{2.4a}$$

$$\mathcal{S}_i := \begin{bmatrix} I_{n_{<i}} & \mathbf{0}_{n_{<i}\times n_i} & \mathbf{0}_{n_{<i}\times n_{>i}} \\ \mathbf{0}_{n_{>i}\times n_{<i}} & \mathbf{0}_{n_{>i}\times n_i} & I_{n_{>i}}\end{bmatrix} \in \mathbb{R}^{n_{-i}\times n} \tag{2.4b}$$

where $n_{<i} := \sum_{j<i, j\in\mathcal{I}} n_j$, $n_{>i} := \sum_{j>i, j\in\mathcal{I}} n_j$ and $n_{-i} := n - n_i$. In simple terms, $\mathcal{R}_i$ selects the i-th n_i-dimensional component from an n-dimensional vector, while $\mathcal{S}_i$ removes it. Thus, $\mathcal{R}_i\boldsymbol{x}_i = \boldsymbol{x}_{i,i} = x_i$ and $\mathcal{S}_i\boldsymbol{x}_i = \boldsymbol{x}_{i,-i}$. Let $\mathcal{R} := \operatorname{diag}\big((\mathcal{R}_i)_{i\in\mathcal{I}}\big)$, $\mathcal{S} := \operatorname{diag}\big((\mathcal{S}_i)_{i\in\mathcal{I}}\big)$. It follows that $x = \mathcal{R}\boldsymbol{x}$ and $\operatorname{col}((\boldsymbol{x}_{i,-i})_{i\in\mathcal{I}}) = \mathcal{S}\boldsymbol{x} \in \mathbb{R}^{(N-1)n}$. Moreover, $\boldsymbol{x} = \mathcal{R}^\top x + \mathcal{S}^\top\mathcal{S}\boldsymbol{x}$. We

define the *extended pseudo-gradient* mapping $\boldsymbol{F}:\mathbb{R}^{Nn}\to\mathbb{R}^{n}$ as

$$\boldsymbol{F}(\boldsymbol{x}) := \operatorname{col}\left((\nabla_{x_i} J_i(x_i,\boldsymbol{x}_{i,-i}))_{i\in\mathcal{I}}\right), \tag{2.5}$$

and the operators

$$\boldsymbol{F}_{\mathrm{a}}(\boldsymbol{x}) := \alpha\mathcal{R}^{\top}\boldsymbol{F}(\boldsymbol{x}) + (\boldsymbol{D}_n - \boldsymbol{W}_n)\boldsymbol{x}, \tag{2.6}$$

$$\mathcal{A}(\boldsymbol{\omega}) := \underbrace{\begin{bmatrix}\boldsymbol{F}_{\mathrm{a}}(\boldsymbol{x})\\ \boldsymbol{0}_{Em}\\ \boldsymbol{b}\end{bmatrix} + \begin{bmatrix}\mathcal{R}^{\top}\boldsymbol{A}^{\top}\boldsymbol{\lambda}\\ -\boldsymbol{V}_m\boldsymbol{\lambda}\\ \boldsymbol{V}_m^{\top}\boldsymbol{v} - \boldsymbol{A}\mathcal{R}\boldsymbol{x}\end{bmatrix}}_{:=\mathcal{A}_1(\omega)} + \begin{bmatrix}\mathrm{N}_{\boldsymbol{\Omega}}(\boldsymbol{x})\\ \boldsymbol{0}_{Em}\\ \mathrm{N}_{\mathbb{R}^{Nm}_{\geq 0}}(\boldsymbol{\lambda})\end{bmatrix} \tag{2.7}$$

where $\alpha>0$ is a design constant, $\boldsymbol{\omega} := \operatorname{col}(\boldsymbol{x},\boldsymbol{v},\boldsymbol{\lambda})$, $\boldsymbol{v} := \operatorname{col}((v_\ell)_{\ell\in\{1,\dots,E\}})\in\mathbb{R}^{Em}$, $\boldsymbol{\lambda} := \operatorname{col}((\lambda_i)_{i\in\mathcal{I}})\in\mathbb{R}^{Nm}$, $\boldsymbol{A} := \operatorname{diag}((A_i)_{i\in\mathcal{I}})$, $\boldsymbol{W}_n := W\otimes I_n$, $\boldsymbol{D}_n := D\otimes I_n$, $\boldsymbol{V}_m := V\otimes I_m$, and $\boldsymbol{\Omega} := \{\boldsymbol{x}\in\mathbb{R}^{nN} \mid \mathcal{R}\boldsymbol{x}\in\Omega\}$.

The following lemma relates the unique v-GNE of the game in (2.1) to the zeros of the operator $\mathcal{A}$. The proof is analogous to [109, Th. 1] or Lemma 2.10 in Section 2.9.2, and hence it is omitted.

Lemma 2.1. Let $\mathcal{A}$ be as in (2.7). It holds that $\operatorname{zer}(\mathcal{A})\neq\varnothing$. Moreover, let $\boldsymbol{x}^*\in\mathbb{R}^{Nn}$, $\boldsymbol{\lambda}^*\in\mathbb{R}^{Nm}$; then, the following statements are equivalent:

(i) There exists $\boldsymbol{v}^*$ such that $\operatorname{col}(\boldsymbol{x}^*,\boldsymbol{v}^*,\boldsymbol{\lambda}^*)\in\operatorname{zer}(\mathcal{A})$.

(ii) $\boldsymbol{x}^*=\mathbf{1}_N\otimes x^*$ and $\boldsymbol{\lambda}^*=\mathbf{1}_N\otimes\lambda^*$, where the pair (x^*,λ^*) satisfies the KKT conditions in (2.3), hence x^* is the v-GNE of the game in (2.1). □

Effectively, Lemma 2.1 provides an extension of the KKT conditions in (2.3) and allows us to recast the GNE problem as that of computing a zero of the operator $\mathcal{A}$, for which a number of iterative algorithms are available [8, §26-28]. In fact, in §2.4.3, we show that Algorithm 2.1 can be recast as a PPA [8, Th. 23.41].

Nonetheless, technical difficulties arise in the analysis because of the partial-decision information setup. Specifically, in (2.5), each partial gradient $\nabla_{x_i}J_i\left(x_i,\boldsymbol{x}_{i,-i}\right)$ is evaluated on the local estimate $\boldsymbol{x}_{i,-i}$, and not on the actual value x_{-i}. Only when the estimates $\boldsymbol{x}$ are at consensus, i.e., $\boldsymbol{x}=\mathbf{1}_N\otimes x$ (namely, the estimate of each agents coincide with the actual value of x), we have that $\boldsymbol{F}(\boldsymbol{x})=F(x)$. As a result, the operator $\mathcal{R}^{\top}\boldsymbol{F}$ (and consequently the operator $\mathcal{A}$) is not monotone in general[2], not even under the strong monotonicity of the game mapping F in Standing Assumption 2.2. Instead, analogously to the approaches in [120], [109], [63], our analysis is based on a *restricted* monotonicity property.

Definition 2.2. An operator $\mathcal{F}:\mathbb{R}^q\rightrightarrows\mathbb{R}^q$ is restricted (μ-strongly) monotone in $\mathcal{H}_P$ if $\operatorname{zer}(\mathcal{F})\neq\varnothing$ and $\langle\omega-\omega^* \mid u\rangle_P\geq 0$ ($\geq\mu\|\omega-\omega^*\|_P^2$) for all $(\omega,u)\in\operatorname{gra}(\mathcal{F})$, $\omega^*\in\operatorname{zer}(\mathcal{F})$ (we omit the characterization "in $\mathcal{H}_P$" whenever $P=I$). □

[2] In Chapter 3, we show that $\mathcal{R}^{\top}\boldsymbol{F}$ is monotone only if the mappings $\nabla_{x_i}J_i(x)$'s do not depend on x_{-i} (in which case, there is no need for a partial-decision information assumption).

Definition 2.2 differs from that in [109, Lem. 3], as we only consider properties with respect to the zero set and we need to include set-valued operators. The definition comprises the nonemptiness of the zero set and it does not exclude an operator that is multi-valued on its zeros. The next lemmas show that restricted monotonicity of $\mathcal{A}$ can be guaranteed for any game satisfying Standing Assumptions 2.1–2.3, without additional hypotheses.

Lemma 2.2 (*[23, Lemma 3]*)**.** The mapping $\boldsymbol{F}$ in (2.5) is θ-Lipschitz continuous, for some $\theta \in [\mu, \theta_0]$. □

Lemma 2.3. Let $\alpha_{\max} := \frac{4\mu\lambda_2(L)}{(\theta_0+\theta)^2+4\mu\theta}$,

$$M := \alpha \begin{bmatrix} \frac{\mu}{N} & -\frac{\theta_0+\theta}{2\sqrt{N}} \\ -\frac{\theta_0+\theta}{2\sqrt{N}} & \frac{\lambda_2(L)}{\alpha} - \theta \end{bmatrix}, \quad \mu_{\boldsymbol{F}_a} := \lambda_{\min}(M). \tag{2.8}$$

If $\alpha \in (0, \alpha_{\max}]$, then $\mu_{\boldsymbol{F}_a} \geq 0$ and $\mathcal{A}$ in (2.7) is restricted monotone. □

Proof. The operator $\mathcal{A}$ in (2.7) is the sum of three operators. The third is monotone by properties of normal cones [8, Th. 20.25]; the second is a linear skew-symmetric operator, hence monotone [8, Ex. 20.35]. Let $\boldsymbol{\omega}^* = \operatorname{col}(\boldsymbol{x}^*, \boldsymbol{v}^*, \boldsymbol{\lambda}^*) \in \operatorname{zer}(\mathcal{A})$, where $\operatorname{zer}(\mathcal{A}) \neq \varnothing$ by Lemma 2.1. By Lemma 2.1, $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, with x^* the v-GNE of the game in (2.1); hence by [109, Lemma 3], for any $\alpha \in (0, \alpha_{\max}]$, it holds that $M \succcurlyeq 0$ and that, for all $\boldsymbol{x} \in \mathbb{R}^{Nn}$

$$\langle \boldsymbol{x} - \boldsymbol{x}^* \mid \boldsymbol{F}_a(\boldsymbol{x}) - \boldsymbol{F}_a(\boldsymbol{x}^*) \rangle \geq \mu_{\boldsymbol{F}_a} \|\boldsymbol{x} - \boldsymbol{x}^*\|^2. \tag{2.9}$$

Hence, for all $(\boldsymbol{\omega}, \boldsymbol{u}) \in \operatorname{gra}(\mathcal{A})$, with $\boldsymbol{\omega} = \operatorname{col}(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{\lambda})$, $\langle \boldsymbol{\omega} - \boldsymbol{\omega}^* \mid \boldsymbol{u} - \mathbf{0} \rangle \geq \mu_{\boldsymbol{F}_a} \|\boldsymbol{x} - \boldsymbol{x}^*\|^2 \geq 0$. ■

2.4.2. PPA FOR RESTRICTED MONOTONE OPERATORS

In the remainder of this section, we show that Algorithm 2.1 is an instance of the PPA, applied to seek a zero of the (suitably preconditioned) operator $\mathcal{A}$ in (2.7). Then, we show its convergence based on the restricted monotonicity result in Lemma 2.3.

Informally speaking, in proximal-point methods, a problem is decomposed into a sequence of regularized subproblems, which are possibly better conditioned and easier to solve. Let $\mathcal{B} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$ be maximally monotone [8, Def. 20.20] in a space $\mathcal{H}_P$, and $\mathrm{J}_{\mathcal{B}} = (\mathrm{Id} + \mathcal{B})^{-1}$ its resolvent. Then, $\operatorname{dom}(\mathrm{J}_{\mathcal{B}}) = \mathbb{R}^q$ and $\mathrm{J}_{\mathcal{B}}$ is single-valued; moreover, if $\operatorname{zer}(\mathcal{B}) \neq \varnothing$, then the sequence $(\omega^k)_{k\in\mathbb{N}}$ generated by the PPA,

$$(\forall\, k \in \mathbb{N}) \quad \omega^{k+1} = \mathrm{J}_{\mathcal{B}}(\omega^k), \quad \omega^0 \in \mathbb{R}^q, \tag{2.10}$$

converges to a point in $\operatorname{zer}(\mathcal{B}) = \operatorname{fix}(\mathrm{J}_{\mathcal{B}})$ [8, Th. 23.41]. Note that performing the update in (2.10) is equivalent to solving for ω^{k+1} the (regularized) inclusion

$$\mathbf{0} \in \mathcal{B}(\omega^{k+1}) + \omega^{k+1} - \omega^k. \tag{2.11}$$

Unfortunately, many operator-theoretic properties are not guaranteed if $\mathcal{B}$ is only restricted monotone. In fact, $\mathrm{J}_{\mathcal{B}}$ might not be defined everywhere or single-valued.

Example 2.1. Let $\mathcal{B}:\mathbb{R}\to\mathbb{R}$, with $\mathcal{B}(\omega)=9-2\omega$ if $\omega\in[3,4)$, $\mathcal{B}(\omega)=\omega$ otherwise. Then, $\operatorname{zer}(\mathcal{B})=\{0\}$ and $\mathcal{B}$ is restricted strongly monotone. However, $\mathrm{J}_{\mathcal{B}}(\omega)=\{\frac{\omega}{2},9-\omega\}$ if $\omega\in[5,6)$ and $\mathrm{J}_{\mathcal{B}}(\omega)=\varnothing$ if $\omega\in(6,8)$. □

Nonetheless, some important properties carry on to the restricted monotone case, as we prove next.

Lemma 2.4. Let $\mathcal{B}:\mathbb{R}^q\rightrightarrows\mathbb{R}^q$ be restricted monotone in $\mathcal{H}_P$. Then, $\mathrm{J}_{\mathcal{B}}$ is firmly quasinonexpansive in $\mathcal{H}_P$: for any $(\omega,u)\in\operatorname{gra}(\mathrm{J}_{\mathcal{B}})$, $\omega^*\in\operatorname{zer}(\mathcal{B})=\operatorname{fix}(\mathrm{J}_{\mathcal{B}})$, it holds that

$$\langle\omega-u\mid\omega-\omega^*\rangle_P-\|u-\omega\|_P^2=\langle\omega-u\mid u-\omega^*\rangle_P\geq 0. \tag{2.12}$$

Moreover, $\mathrm{J}_{\mathcal{B}}(\omega^*)=\{\omega^*\}$. □

Proof. By definition of resolvent, $\omega^*\in\mathrm{J}_{\mathcal{B}}(\omega^*)\ \Leftrightarrow\ \omega^*+\mathcal{B}\omega^*\ni\omega^*\ \Leftrightarrow\ \mathbf{0}\in\mathcal{B}(\omega^*)$; also, for any $(\omega,u)\in\operatorname{gra}(\mathrm{J}_{\mathcal{B}})$, $\omega-u\in\mathcal{B}(u)$. Hence, the inequality in (2.12) is the restricted monotonicity of $\mathcal{B}$; the elementary equality follows by expanding the terms. Finally, by taking $\omega=\omega^*$ in (2.12), we infer that $\mathrm{J}_{\mathcal{B}}$ is single-valued on $\operatorname{fix}(\mathrm{J}_{\mathcal{B}})$. ■

Next, by leveraging Lemma 2.4, we extend classical results for the PPA [41, Th. 5.6] to the case of a restricted monotone operator (possibly with multi-valued resolvent).

Theorem 2.1. Let $\mathcal{B}:\mathbb{R}^q\rightrightarrows\mathbb{R}^q$ be restricted monotone in $\mathcal{H}_P$, and $C:=\operatorname{zer}(\mathcal{B})\neq\varnothing$. Let $(\gamma^k)_{k\in\mathbb{N}}$ be a sequence in $[0,2]$, and $(e^k)_{k\in\mathbb{N}}$ a sequence in $\mathbb{R}^q$ such that $(\gamma^k\|e^k\|_P)_{k\in\mathbb{N}}\in\ell^1$ (where ℓ^1 is the set of absolutely summable sequences). Let $\omega^0\in\mathbb{R}^q$ and let $(\omega^k)_{k\in\mathbb{N}}$ be any sequence such that:

$$(\forall k\in\mathbb{N})\quad \omega^{k+1}=\omega^k+\gamma^k(u^k-\omega^k+e^k),\ u^k\in\mathrm{J}_{\mathcal{B}}(\omega^k). \tag{2.13}$$

Then, the following statements hold:

(i) $(\forall\omega^*\in C)(\forall k\in\mathbb{N})\ \ \|\omega^{k+1}-\omega^*\|_P\leq\|\omega^k-\omega^*\|_P+\gamma^k\|e^k\|_P$.

(ii) $\big(\gamma^k(2-\gamma^k)\|u^k-\omega^k\|_P^2\big)_{k\in\mathbb{N}}\in\ell^1$.

(iii) Assume that every cluster point of $(\omega^k)_{k\in\mathbb{N}}$ belongs to C. Then, $(\omega^k)_{k\in\mathbb{N}}$ converges to a point in C.

(iv) Assume that $\mathcal{B}$ is $\mu_{\mathcal{B}}$-strongly restricted monotone in $\mathcal{H}_P$. Then, $C=\{\omega^*\}$ and, for all $k\in\mathbb{N}$, $\|\omega^{k+1}-\omega^*\|_P\leq\rho^k\|\omega^k-\omega^*\|_P+\gamma^k\|e^k\|_P$, where $\rho^k=\max(1-\frac{\gamma^k\mu_{\mathcal{B}}}{1+\mu_{\mathcal{B}}},\gamma^k-1)$. □

Proof. See Section 2.9.1. ■

Remark 2.3. The condition $\operatorname{dom}(\mathrm{J}_{\mathcal{B}})=\mathbb{R}^q$ is sufficient (but not necessary) for the existence of a sequence $(\omega^k)_{k\in\mathbb{N}}$ that satisfies (2.13), which can be constructed choosing arbitrarily $u_k\in\mathrm{J}_{\mathcal{B}}(\omega^k)$, for all $k\in\mathbb{N}$. □

Example 2.2. Consider the $\mathrm{VI}(\Psi,S)$ (see Appendix C), where $S\subset\mathbb{R}^q$ is compact and convex, and $\Psi:\mathbb{R}^q\to\mathbb{R}^q$ is continuous and pseudomonotone in the sense of Karamardian (i.e., for all $\omega,\omega'\in\mathbb{R}^q$, the implication $\langle\Psi(\omega),\omega'-\omega\rangle\geq 0\Rightarrow\langle\Psi(\omega'),\omega'-\omega\rangle\geq 0$ holds). It

holds that $\mathrm{SOL}(\Psi, S) = \mathrm{zer}(\mathcal{B}) \neq \varnothing$, where $\mathcal{B} = \Psi + \mathrm{N}_S$ [56, Prop. 2.2.3] (where $\mathrm{SOL}(\Psi, S)$ denotes the solution set of $\mathrm{VI}(\Psi, S)$). Moreover $\mathcal{B}$ is restricted monotone. To show this, consider any $\omega^* \in \mathrm{zer}(\mathcal{B})$ and $(\omega, u) \in \mathrm{gra}(\mathcal{B})$, so $u = \Psi(\omega) + u'$, for some u' such that $(\omega, u') \in \mathrm{gra}(\mathrm{N}_S)$. Then, $\langle u \mid \omega - \omega^* \rangle = \langle \Psi(\omega) \mid \omega - \omega^* \rangle + \langle u' - \mathbf{0} \mid \omega - \omega^* \rangle \geq 0$, where we used that $\langle \Psi(\omega) \mid \omega - \omega^* \rangle \geq 0$, by pseudomonotonicity and because $\langle \Psi(\omega^*) \mid \omega - \omega^* \rangle \geq 0$ by definition of VI, and $\langle u' - \mathbf{0} \mid \omega - \omega^* \rangle \geq 0$ because $(\omega^*, \mathbf{0}) \in \mathrm{gra}(\mathrm{N}_S)$ and monotonicity of the normal cone.

We note that $\mathrm{dom}(\mathrm{J}_{\mathcal{B}}) = \mathbb{R}^q$ by [56, Prop. 2.2.3]. Let us consider any sequence $(\omega^k)_{k\in\mathbb{N}}$ such that, for all $k \in \mathbb{N}$, $\omega^{k+1} = u^k$, $u^k \in \mathrm{J}_{\mathcal{B}}(\omega^k)$, (or equivalently (2.11) or $\omega^{k+1} \in \mathrm{SOL}(\Psi + \mathrm{Id} - \omega^k, S)$). By Theorem 2.1 (with $\gamma^k = 1, e^k = \mathbf{0}$), $(\omega^k)_{k\in\mathbb{N}}$ is bounded, hence it admits at least one cluster point, say $\bar{\omega}$; by Theorem 2.1*(ii)* $\|u^k - \omega^k\| \to 0$. However, by definition of VI, for any $\omega \in S$, $\langle \Psi(u^k) + u^k - \omega^k \mid \omega - u^k \rangle \geq 0$. By passing to the limit (on a subsequence) and by continuity, we obtain $\langle \Psi(\bar{\omega}) \mid \omega - \bar{\omega} \rangle \geq 0$, which shows that $\bar{\omega} \in \mathrm{SOL}(\Psi, S)$. Therefore $(\omega^k)_{k\in\mathbb{N}}$ converges to a solution to $\mathrm{VI}(\Psi, S)$ by Theorem 2.1*(iii)*. This extends the results in [52, §4.2], where hypomonotonicity of Ψ is assumed and where a small-enough step size is chosen to ensure that $\mathrm{J}_{\mathcal{B}}$ is single-valued (besides, pseudomonotonicity of Ψ is sufficient, but not necessary, for the restricted monotonicity of $\mathcal{B}$, and Theorem 2.1 would also allow to take into account iterations with errors, cf. [52, §4.2]). □

2.4.3. DERIVATION AND CONVERGENCE

Next, we show how that Algorithm 2.1 is obtained by applying the iteration in (2.13) to the operator $\Phi^{-1}\mathcal{A}$, where

$$\Phi := \begin{bmatrix} \bar{\tau}^{-1} + \boldsymbol{W}_n & \mathbf{0} & -\mathcal{R}^\top \boldsymbol{A}^\top \\ \mathbf{0} & \bar{\nu}^{-1} & \boldsymbol{V}_m \\ -\boldsymbol{A}\mathcal{R} & \boldsymbol{V}_m^\top & \bar{\delta}^{-1} \end{bmatrix} \tag{2.14}$$

is called *preconditioning* matrix. The step sizes $\bar{\tau} := \mathrm{diag}((\tau_i I_n)_{i\in\mathcal{I}})$, $\bar{\nu} := \mathrm{diag}((\nu_{(i,j)} I_m)_{(i,j)\in\mathcal{E}})$, $\bar{\delta} := \mathrm{diag}((\delta_i I_m)_{i\in\mathcal{I}})$, have to be chosen such that $\Phi \succ 0$. In this case, it also holds that $\mathrm{zer}(\Phi^{-1}\mathcal{A}) = \mathrm{zer}(\mathcal{A})$. Sufficient conditions that ensure $\Phi \succ 0$ are given in the next lemma, which follows by the Gershgorin's circle theorem.

Lemma 2.5. The matrix Φ in (2.14) is positive definite if ${\nu_{(i,j)}}^{-1} > 2\sqrt{(w_{i,j})}$ for all $(i, j) \in \mathcal{E}$ and $\tau_i^{-1} > d_i + \|A_i^\top\|_\infty$, $\delta_i^{-1} > \|A_i\|_\infty + \sum_{j=1}^N \sqrt{(w_{i,j})}$ for all $i \in \mathcal{I}$. □

In the following, we always assume that the step sizes in Algorithm 2.1 are chosen such that $\Phi \succ 0$. Then, we are able to formulate the following result.

Lemma 2.6. Algorithm 2.1 is equivalent to the iteration

$$(\forall k \in \mathbb{N}) \quad \boldsymbol{\omega}^{k+1} \in \mathrm{J}_{\Phi^{-1}\mathcal{A}}(\boldsymbol{\omega}^k), \tag{2.15}$$

with $\mathcal{A}$ as in (2.7), Φ as in (2.14): for any initial condition $\boldsymbol{\omega}^0 = \mathrm{col}(\boldsymbol{x}^0, \boldsymbol{v}^0 = \mathbf{0}_{Em}, \boldsymbol{\lambda}^0)$, the sequence $(\boldsymbol{x}^k, \boldsymbol{V}_m^\top \boldsymbol{v}^k, \boldsymbol{\lambda}^k)_{k\in\mathbb{N}}$ generated by (2.15) coincides with the sequence $(\boldsymbol{x}^k, \boldsymbol{z}^k, \boldsymbol{\lambda}^k)_{k\in\mathbb{N}}$ generated by Algorithm 2.1 with initial conditions $(\boldsymbol{x}^0, \boldsymbol{z}^0 = \mathbf{0}_{Nm}, \boldsymbol{\lambda}^0)$. □

Proof. By definition of inverse operator, we have that

$$\boldsymbol{\omega}^{k+1} \in (\mathrm{Id} + \Phi^{-1}\mathcal{A})^{-1}(\boldsymbol{\omega}^k)$$

$$
\begin{aligned}
&\Leftrightarrow \mathbf{0} \in \Phi^{-1}\mathcal{A}(\boldsymbol{\omega}^{k+1}) - \boldsymbol{\omega}^k + \boldsymbol{\omega}^{k+1} \\
&\Leftrightarrow \mathbf{0} \in \Phi(\boldsymbol{\omega}^{k+1} - \boldsymbol{\omega}^k) + \mathcal{A}(\boldsymbol{\omega}^{k+1}) \qquad (2.16)\\
&\Leftrightarrow \begin{cases}
\mathbf{0} \in \bar{\tau}^{-1}(\boldsymbol{x}^{k+1} - \boldsymbol{x}^k) + \cancel{\boldsymbol{W}_n \boldsymbol{x}^{k+1}} - \boldsymbol{W}_n \boldsymbol{x}^k + \boldsymbol{D}_n \boldsymbol{x}^{k+1} \\
\qquad - \cancel{\mathcal{R}^\top \boldsymbol{A}^\top \boldsymbol{\lambda}^{k+1}} + \mathcal{R}^\top \boldsymbol{A}^\top \boldsymbol{\lambda}^k + \alpha \mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}^{k+1}) \\
\qquad - \cancel{\boldsymbol{W}_n \boldsymbol{x}^{k+1}} + \cancel{\mathcal{R}^\top \boldsymbol{A}^\top \boldsymbol{\lambda}^{k+1}} + \mathrm{N}_{\boldsymbol{\Omega}}(\boldsymbol{x}^{k+1}) \\
\mathbf{0} \in \bar{\nu}^{-1}(\boldsymbol{v}^{k+1} - \boldsymbol{v}^k) + \cancel{\boldsymbol{V}_m \boldsymbol{\lambda}^{k+1}} - \boldsymbol{V}_m \boldsymbol{\lambda}^k - \cancel{\boldsymbol{V}_m \boldsymbol{\lambda}^{k+1}} \\
\mathbf{0} \in \bar{\delta}^{-1}(\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k) + \mathrm{N}_{\mathbb{R}^{mN}_{\geq 0}}(\boldsymbol{\lambda}^{k+1}) + \boldsymbol{b} \\
\qquad - \boldsymbol{A}\mathcal{R}(2\boldsymbol{x}^{k+1} - \boldsymbol{x}^k) + \boldsymbol{V}_m^\top (2\boldsymbol{v}^{k+1} - \boldsymbol{v}^k)
\end{cases} \qquad (2.17)
\end{aligned}
$$

In turn, the first inclusion in (2.17) can be split in two by left-multiplying both sides with $\mathcal{R}$ and $\mathcal{S}$. By $\mathcal{S}\mathrm{N}_{\boldsymbol{\Omega}} = \mathbf{0}_{(N-1)n}$, $\mathcal{R}\mathcal{R}^\top = I_n$ and $\mathcal{S}\mathcal{R}^\top = \mathbf{0}_{(N-1)n \times n}$, we get

$$
\begin{aligned}
&\begin{cases}
\mathbf{0} \in \mathcal{S}((I + \bar{\tau}\boldsymbol{D}_n)\boldsymbol{x}^{k+1} - \boldsymbol{x}^k - \bar{\tau}\boldsymbol{W}_n \boldsymbol{x}^k) \\
\mathbf{0} \in \mathcal{R}((I + \bar{\tau}\boldsymbol{D}_n)\boldsymbol{x}^{k+1} - \boldsymbol{x}^k - \bar{\tau}\boldsymbol{W}_n \boldsymbol{x}^k) \\
\qquad + \mathrm{N}_\Omega(x^{k+1}) + \alpha\bar{\tau}\boldsymbol{F}((x^{k+1}, \mathcal{S}\boldsymbol{x}^{k+1})) + \bar{\tau}\boldsymbol{A}^\top \boldsymbol{\lambda}^k
\end{cases} \\
&\underset{\forall i \in \mathcal{I}}{\Leftrightarrow} \begin{cases}
\boldsymbol{x}_{i,-i}^{k+1} = \frac{1}{1+\tau_i d_i}(\boldsymbol{x}_{i,-i}^k + \tau_i \sum_{j=1}^N w_{i,j} \boldsymbol{x}_{j,-i}^k) \\
\mathbf{0}_{n_i} \in \partial_{x_i^{k+1}} \big(J_i(x_i^{k+1}, \boldsymbol{x}_{i,-i}^{k+1}) + \frac{1}{2\alpha\tau_i} \big\| x_i^{k+1} - x_i^k \big\|^2 \\
\qquad + \frac{1}{2\alpha d_i} \big\| d_i x_i^{k+1} - \sum_{j=1}^N w_{i,j} \boldsymbol{x}_{j,i}^k \big\|^2 \\
\qquad + \iota_{\Omega_i}(x_i^{k+1}) + \frac{1}{\alpha}(A_i^\top \lambda_i^k)^\top x_i^{k+1} \big).
\end{cases}
\end{aligned}
$$

Therefore, since the zeros of the subdifferential of a (strongly) convex function coincide with the minima (unique minimum) [8, Th. 16.3], (2.17) can be rewritten as

$$
\begin{aligned}
&\forall\, i \in \mathcal{I} : \begin{cases}
\boldsymbol{x}_{i,-i}^{k+1} = \frac{1}{1+\tau_i d_i}(\boldsymbol{x}_{i,-i}^k + \tau_i \sum_{j=1}^N w_{i,j} \boldsymbol{x}_{j,-i}^k) \\
x_i^{k+1} = \underset{y \in \Omega_i}{\operatorname{argmin}} \big(J_i(y, \boldsymbol{x}_{i,-i}^{k+1}) + \frac{1}{2\alpha\tau_i} \big\| y - x_i^k \big\|^2 + \frac{1}{2\alpha d_i} \big\| d_i y - \sum_{j=1}^N w_{i,j} \boldsymbol{x}_{j,i}^k \big\|^2 \\
\qquad\qquad + \frac{1}{\alpha}(A_i^\top \lambda_i^k)^\top y \big)
\end{cases} \\
&\boldsymbol{v}^{k+1} = \boldsymbol{v}^k + \bar{\nu}\boldsymbol{V}_m \boldsymbol{\lambda}^k \\
&\boldsymbol{\lambda}^{k+1} = \mathrm{proj}_{\mathbb{R}^{mN}_{\geq 0}} \big(\boldsymbol{\lambda}^k + \bar{\delta} \big(\boldsymbol{A}\mathcal{R}(2\boldsymbol{x}^{k+1} - \boldsymbol{x}^k) - \boldsymbol{b} - \boldsymbol{V}_m^\top (2\boldsymbol{v}^{k+1} - \boldsymbol{v}^k) \big) \big).
\end{aligned} \qquad (2.18)
$$

The conclusion follows by defining $\boldsymbol{z}^k := \boldsymbol{V}_m^\top \boldsymbol{v}^k$, where $\boldsymbol{z}^k = \mathrm{col}((z_i)_{i \in \mathcal{I}}) \in \mathbb{R}^{Nm}$ and $z_i^k \in \mathbb{R}^m$ are local variables kept by each agent, provided that $\boldsymbol{z}^0 = \boldsymbol{V}_m^\top \boldsymbol{v}^0$. The latter is ensured by $\boldsymbol{z}^0 = \mathbf{0}_{Nm}$, as in Algorithm 2.1. ■

Remark 2.4. The preconditioning matrix Φ is designed to make the system in (2.17) block triangular, i.e., to remove the term $\boldsymbol{W}_n \boldsymbol{x}^{k+1}$ and $\mathcal{R}^\top \boldsymbol{A}^\top \boldsymbol{\lambda}^{k+1}$ from the first inclusion, and the terms $\boldsymbol{V}_m \boldsymbol{\lambda}^{k+1}$ from the second one: in this way, $\boldsymbol{x}_i^{k+1}$ and $\boldsymbol{z}^{k+1}$ do not depend on $\boldsymbol{x}_j^{k+1}$, for $i \neq j$, or $\boldsymbol{\lambda}^{k+1}$. This ensures that the resulting iteration can be computed by

the agents in a fully-distributed fashion (differently from the non-preconditioned resolvent $\mathrm{J}_{\mathcal{A}}$). Furthermore, the change of variable $\boldsymbol{z} = \boldsymbol{V}_m^\top \boldsymbol{\nu}$ reduces the number of auxiliary variables and decouples the dual update in (2.18) from the graph structure. □

Remark 2.5. By Lemma 2.6, Remark 2.1 and by $\mathrm{J}_{\Phi^{-1}\mathcal{A}}$ in (2.18), we conclude that $\mathrm{dom}(\mathrm{J}_{\Phi^{-1}\mathcal{A}}) = \mathbb{R}^{Nn+Em+Nm}$ and that $\mathrm{J}_{\Phi^{-1}\mathcal{A}}$ is single-valued. □

In order to apply Theorem 2.1 to the iteration in (2.15), we still need the following lemma.

Lemma 2.7. Let $\alpha \in (0, \alpha_{\max}]$, $\alpha_{\max}$ as in Lemma 2.3. Then $\Phi^{-1}\mathcal{A}$ is restricted monotone in $\mathcal{H}_\Phi$. □.

Proof. Let $(\boldsymbol{\omega}, \boldsymbol{u}) \in \mathrm{gra}(\Phi^{-1}\mathcal{A})$, $\boldsymbol{\omega}^* \in \mathrm{zer}(\Phi^{-1}\mathcal{A})$. Then, $(\boldsymbol{\omega}, \Phi\boldsymbol{u}) \in \mathrm{gra}(\mathcal{A})$ and $\boldsymbol{\omega}^* \in \mathrm{zer}(\mathcal{A})$. By Lemma 2.3 we conclude that $\langle \boldsymbol{u} \mid \boldsymbol{\omega} - \boldsymbol{\omega}^* \rangle_\Phi = \langle \Phi\boldsymbol{u} \mid \boldsymbol{\omega} - \boldsymbol{\omega}^* \rangle \geq 0$. ■

Theorem 2.2. Let $\alpha \in (0, \alpha_{\max}]$, with $\alpha_{\max}$ as in Lemma 2.3, and let the step sizes $\bar{\tau}, \bar{\nu}, \bar{\delta}$ be as in Lemma 2.5. Then, the sequence $(\boldsymbol{x}^k, \boldsymbol{z}^k, \boldsymbol{\lambda}^k)_{k\in\mathbb{N}}$ generated by Algorithm 2.1 converges to some equilibrium $(\boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{\lambda}^*)$, where $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$ and x^* is the v-GNE of the game in (2.1). □

Proof. By Lemma 2.6, we can equivalently study the convergence of the iteration in (2.15). In turn, (2.15) can be rewritten as (2.13) with $\gamma^k = 1$, $e^k = \mathbf{0}$, for all $k \in \mathbb{N}$. For later reference, let us define $\boldsymbol{u}^k = \mathrm{J}_{\Phi^{-1}\mathcal{A}}(\boldsymbol{\omega}^k)$ (here $\boldsymbol{u}^k = \boldsymbol{\omega}^{k+1}$). $\Phi^{-1}\mathcal{A}$ is restricted monotone in $\mathcal{H}_\Phi$ by Lemma 2.7. By Theorem 2.1*(i)*, the sequence $(\boldsymbol{\omega}^k)_{k\in\mathbb{N}}$ is bounded, hence it admits at least one cluster point, say $\bar{\boldsymbol{\omega}}$. By (2.16) and (2.7), it holds, for any $\boldsymbol{\omega} \in \boldsymbol{\Omega} \times \mathbb{R}^{Em} \times \mathbb{R}^{Nm}_{\geq 0}$, that $\langle \mathcal{A}_1(\boldsymbol{u}^k) + \Phi(\boldsymbol{u}^k - \boldsymbol{\omega}^k) \mid \boldsymbol{\omega} - \boldsymbol{u}^k \rangle \geq 0$, with $\mathcal{A}_1$ as in (2.7). By Theorem 2.1*(ii)*, $\boldsymbol{u}^k - \boldsymbol{\omega}^k \to \mathbf{0}$. Therefore, by continuity of $\mathcal{A}_1$, taking the limit on a diverging subsequence $(l_k)_{k\in\mathbb{N}}$ such that $(\boldsymbol{\omega}^{l_k})_{k\in\mathbb{N}} \to \bar{\boldsymbol{\omega}}$, we have that for all $\boldsymbol{\omega} \in \boldsymbol{\Omega} \times \mathbb{R}^{Em} \times \mathbb{R}^{Nm}_{\geq 0})$, $\langle \mathcal{A}_1(\bar{\boldsymbol{\omega}}) \mid \boldsymbol{\omega} - \bar{\boldsymbol{\omega}} \rangle \geq 0$, which shows that $\bar{\boldsymbol{\omega}} \in \mathrm{zer}(\mathcal{A}) = \mathrm{fix}(\mathrm{J}_{\Phi^{-1}\mathcal{A}})$. Hence $(\boldsymbol{\omega}^k)_{k\in\mathbb{N}}$ converges to an equilibrium of (2.15) by Theorem 2.1*(iii)*. The conclusion follows by Lemma 2.1. ■

Remark 2.6. While the choice of step sizes in Lemma 2.5 is decentralized, computing the bound $\alpha_{\max}$ for the common parameter α in Algorithm 2.1 requires some global information on the graph $\mathcal{G}$ (i.e., the algebraic connectivity) and on the game mapping (the strong monotonicity and Lipschitz constants). □

Remark 2.7. If $\boldsymbol{x}^0 \in \Omega^N$, then $\boldsymbol{x}^k \in \Omega^N$ for all $k \in \mathbb{N}$ (by convexity and the updates in Algorithm 2.1), and Assumption 2.2 can be relaxed to hold only on Ω. □

Remark 2.8 (*Inexact updates*)**.** The local optimization problems in Algorithm 2.1 are strongly convex, hence they can be efficiently solved by several iterative algorithms (with linear rate). While computing the exact solutions $\bar{x}_i^k$ would require an infinite number of iterations, the convergence in Theorem 2.2 still holds if x_i is updated with an approximation $\hat{x}_i^k$ of $\bar{x}_i^k$, provided that the errors $e_i^k := \bar{x}_i^k - \hat{x}_i^k$ are norm summable, i.e., $(\|e_i^k\|)_{k\in\mathbb{N}} \in \ell^1$, for all $i \in \mathcal{I}$ (the same proof applies, since the condition on e^k in Theorem 2.1 would be satisfied, by equivalence of norms). For example, assume that $\hat{x}_i^k$ is

computed via a finite number $j_i^k \geq 1$ of steps of the projected gradient method, warm-started at x_i^k, with (small enough) fixed step. Then, each agent can independently ensure that $\|e_i^k\| \leq \varepsilon_i^k$, for some $(\varepsilon_i^k)_{k\in\mathbb{N}} \in \ell^1$, by simply choosing

$$j_i^k \geq \log\left(\varepsilon_i^k(1-\rho_i)/\|x_i^k - \widehat{x}_i^{k,1}\|\right)/\log(\rho_i), \tag{2.19}$$

where $\widehat{x}_i^{k,1}$ is the approximation obtained after one gradient step and $\rho_i \in (0,1)$ is the contractivity parameter of the gradient descent[3]. We finally remark that $\bar{x}_i^k$ must be estimated with increasing accuracy. In practice, however, when x_i^k is converging, $\|x_i^{k+1} - x_i^k\| \to 0$. Hence x_i^k is a good initial guess for $\bar{x}_i^k$, and the computation of x_i^{k+1} often requires few gradient steps, see also §2.7. □

2.5. ACCELERATIONS

LEMMA 2.6 shows that Algorithm 2.1 can be recast (modulo the change of variables $\boldsymbol{z} = \boldsymbol{V}_m^\top \boldsymbol{v}$) as

$$\boldsymbol{\omega}^{k+1} = T(\boldsymbol{\omega}^k), \tag{2.20}$$

where $T := \mathrm{J}_{\Phi^{-1}\mathcal{A}}$. This compact operator representation allows for some modifications of Algorithm 2.1, that can increase its convergence speed. In particular, we consider three popular accelerations schemes [77], which have been extensively studied for the case of firmly nonexpansive operators [8, Def. 4.1], and also found application in games under full-decision information [14], [124]. Here we provide convergence guarantees for the partial-decision information setup, where T is only firmly quasinonexpansive. Our fully-distributed accelerated algorithms are illustrated in Algorithm 2.2. In the following, we assume that $\alpha \in (0, \alpha_{\max}]$, $\alpha_{\max}$ as in Lemma 2.3, and that the step sizes $\bar{\tau}, \bar{\nu}, \bar{\delta}$ are chosen as in Lemma 2.5.

Proposition 2.1 (*Overrelaxation*)**.** Let $\gamma \in [1,2)$. Then, for any $\boldsymbol{\omega}^0$, the sequence $(\boldsymbol{\omega}^k)_{k\in\mathbb{N}}$ generated by

$$(\forall k \in \mathbb{N}), \quad \boldsymbol{\omega}^{k+1} = \boldsymbol{\omega}^k + \gamma(T(\boldsymbol{\omega}^k) - \boldsymbol{\omega}^k), \tag{2.21}$$

converges to an equilibrium $(\boldsymbol{x}^*, \boldsymbol{v}^*, \boldsymbol{\lambda}^*) \in \mathrm{zer}(\mathcal{A})$, where $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$ and x^* is the v-GNE of the game in (2.1). □

Proof. The iteration in (2.21) is in the form (2.13), with $\gamma^k = \gamma$, $e^k = \mathbf{0}$, for all $k \in \mathbb{N}$. Then, the conclusion follows analogously to Theorem 2.2. ■

Proposition 2.2 (*Inertia*)**.** Let $\zeta \in [0, \frac{1}{3})$. Then, for any $\boldsymbol{\omega}^{-1} := \boldsymbol{\omega}^0$, the sequence $(\boldsymbol{\omega}^k)_{k\in\mathbb{N}}$ generated by

$$(\forall k \in \mathbb{N}), \quad \boldsymbol{\omega}^{k+1} = T(\boldsymbol{\omega}^k + \zeta(\boldsymbol{\omega}^k - \boldsymbol{\omega}^{k-1})), \tag{2.22}$$

converges to an equilibrium $(\boldsymbol{x}^*, \boldsymbol{v}^*, \boldsymbol{\lambda}^*) \in \mathrm{zer}(\mathcal{A})$, where $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$ and x^* is the v-GNE of the game in (2.1). □

[3] ρ_i can be taken independent of k: since $\nabla J_i(\cdot, \boldsymbol{x}_{i,-i})$ is μ_i strongly monotone and θ_i Lipschitz, for some $\mu_i \geq \mu$, $\theta_i \leq \theta$ and for all $\boldsymbol{x}_{i,-i}$, the factor $\rho_i = \frac{\theta_i - \mu_i}{\theta_i + \mu_i + 1/(\alpha\tau_i) + d_i/\alpha}$ is ensured by the step $2/(\theta_i + \mu_i + 1/(\alpha\tau_i) + d_i/\alpha)$.

Algorithm 2.2. Fully-distributed v-GNE seeking via accelerated PPPA

Initialization:

- Choose acceleration:

Overrelaxation:	set $\gamma > 0, \zeta = 0, \eta = 0$;
Inertia:	set $\gamma = 0, \zeta > 0, \eta = 0$;
Alternated inertia:	set $\gamma = 0, \zeta = 0, \eta > 0$;

- For all $i \in \mathcal{I}$, set $x_i^{-1} = x_i^0 \in \Omega_i$, $\boldsymbol{x}_{i,-i}^{-1} = \boldsymbol{x}_{i,-i}^0 \in \mathbb{R}^{n-n_i}$, $z_i^{-1} = z_i^0 = \mathbf{0}_m$, $\lambda_i^{-1} = \lambda_i^0 \in \mathbb{R}^m_{\geq 0}$.

For all $k > 0$:

- (Alternated) inertial step: set $\tilde{\eta}^k = 0$ if k is even, $\tilde{\eta}^k = \eta$ otherwise; each agent $i \in \mathcal{I}$ computes

$$\begin{aligned}
\tilde{\boldsymbol{x}}_{i,-i}^k &= \boldsymbol{x}_{i,-i}^k + (\zeta + \tilde{\eta}^k)(\boldsymbol{x}_{i,-i}^k - \boldsymbol{x}_{i,-i}^{k-1}) \\
\tilde{x}_i^k &= x_i^k + (\zeta + \tilde{\eta}^k)(x_i^k - x_i^{k-1}) \\
\tilde{z}_i^k &= z_i^k + (\zeta + \tilde{\eta}^k)(z_i^k - z_i^{k-1}) \\
\tilde{\lambda}_i^k &= \lambda_i^k + (\zeta + \tilde{\eta}^k)(\lambda_i^k - \lambda_i^{k-1})
\end{aligned}$$

- Communication: The agents exchange $\{\tilde{x}_i^k, \tilde{\boldsymbol{x}}_{i,-i}^k, \tilde{\lambda}_i^k\}$ with their neighbors.
- Resolvent computation: each agent $i \in \mathcal{I}$ computes

$$\begin{aligned}
\check{\boldsymbol{x}}_{i,-i}^{k+1} &= \tfrac{1}{1+\tau_i d_i}(\tilde{\boldsymbol{x}}_{i,-i}^k + \tau_i \textstyle\sum_{j=1}^N w_{i,j}\tilde{\boldsymbol{x}}_{j,-i}^k) \\
\check{x}_i^{k+1} &= \operatorname*{argmin}_{y \in \Omega_i}\big(J_i(y, \check{\boldsymbol{x}}_{i,-i}^{k+1}) + \tfrac{1}{2\alpha\tau_i}\big\|y - \tilde{x}_i^k\big\|^2 + \tfrac{1}{2\alpha d_i}\big\|d_i y - \textstyle\sum_{j=1}^N w_{i,j}\tilde{\boldsymbol{x}}_{j,i}^k\big\|^2 + \tfrac{1}{\alpha}(A_i^\top \tilde{\lambda}_i^k)^\top y\big) \\
\check{z}_i^{k+1} &= \tilde{z}_i^k + \textstyle\sum_{j=1}^N \nu_{(i,j)} w_{i,j}(\tilde{\lambda}_i^k - \tilde{\lambda}_j^k) \\
\check{\lambda}_i^{k+1} &= \operatorname{proj}_{\mathbb{R}^m_{\geq 0}}\big(\tilde{\lambda}_i^k + \delta_i\big(A_i(2\check{x}_i^{k+1} - \tilde{x}_i^k) - b_i - (2\check{z}_i^{k+1} - \tilde{z}_i^k)\big)\big).
\end{aligned}$$

- Relaxation step: each agent $i \in \mathcal{I}$ computes

$$\begin{aligned}
\boldsymbol{x}_{i,-i}^{k+1} &= \gamma\check{\boldsymbol{x}}_{i,-i}^{k+1} + (1-\gamma)\boldsymbol{x}_{i,-i}^k \\
x_i^{k+1} &= \gamma\check{x}_i^{k+1} + (1-\gamma)x_i^k \\
z_i{}^{k+1} &= \gamma\check{z}_i^{k+1} + (1-\gamma)z_i^k \\
\lambda_i^{k+1} &= \gamma\check{\lambda}_i^{k+1} + (1-\gamma)\lambda_i^k
\end{aligned}$$

Proof (sketch). By following all the steps in the proof of [32, Th. 5] (which can be done by recalling that an operator T is firmly (quasi)nonexpansive if and only if the operator $2T - \mathrm{Id}$ is (quasi)nonexpansive [8, Prop. 4.2, 4.4]), it can be shown that, if $\zeta \in [0, \frac{1}{3})$, then $(\boldsymbol{\omega}^k)_{k\in\mathbb{N}}$ is bounded and $\boldsymbol{\omega}^{k+1} - \boldsymbol{\omega}^k \to 0$. Then, the proof follows analogously to Theorem 2.2. ■

Proposition 2.3 (*Alternated inertia*)**.** Let $\eta \in [0,1]$. Then, for any $\boldsymbol{\omega}^0$, the sequence $(\boldsymbol{\omega}^k)_{k\in\mathbb{N}}$ generated by

$$\begin{cases} \boldsymbol{\omega}^{k+1} = T(\boldsymbol{\omega}^k) & \text{if } k \text{ is even,} \\ \boldsymbol{\omega}^{k+1} = T(\boldsymbol{\omega}^k + \eta(\boldsymbol{\omega}^k - \boldsymbol{\omega}^{k-1})) & \text{if } k \text{ is odd,} \end{cases} \tag{2.23}$$

converges to an equilibrium $(\boldsymbol{x}^*, \boldsymbol{v}^*, \boldsymbol{\lambda}^*) \in \operatorname{zer}(\mathcal{A})$, where $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$ and x^* is the v-GNE of the game in (2.1). □

Proof. For all $k \in \mathbb{N}$, $\boldsymbol{\omega}^{2k+2} = T(T(\boldsymbol{\omega}^{2k}) + \eta(T(\boldsymbol{\omega}^{2k}) - \boldsymbol{\omega}^{2k}))$, which is the same two-steps update obtained in (2.13) with $\gamma^{2k} = 1+\eta$, $\gamma^{2k+1} = 1$ (and $\mathcal{B} := \Phi^{-1}\mathcal{A}$, $e^k = \mathbf{0}$). Therefore the convergence of the sequence $(\boldsymbol{\omega}^{2k})_{k\in\mathbb{N}}$ to an equilibrium $(\boldsymbol{x}^*, \boldsymbol{v}^*, \boldsymbol{\lambda}^*) \in \operatorname{zer}(\mathcal{A})$ follows analogously to Theorem 2.2 (with a minor modification for the case $\eta = 1$). The convergence of the sequence $(\boldsymbol{\omega}^{2k+1})_{k\in\mathbb{N}}$ then follows by Theorem 2.1(*i*). ■

We note that, by Theorem 2.1, the convergence results in Propositions 2.1 and 2.3 hold also in the case of summable errors on the updates, as in Remark 2.8. Analogously to our analysis, provably convergent acceleration schemes could also be obtained for the FB algorithm in [109]: however, an advantage of our PPA is that the bounds on the inertial/relaxation parameters are fixed and independent on (unknown) problem parameters.

2.5.1. ON THE CONVERGENCE RATE

We conclude this section with a discussion on the convergence rate of Algorithms 2.1 and 2.2. First, even under Standing Assumption 2.2, the KKT operator on the right-hand side of (2.3) is generally not strongly monotone. Similarly, the operator $\mathcal{A}$ in (2.7) is not strongly monotone and Algorithm 2.1 can have multiple fixed points. Therefore, one should not expect linear convergence. By Lemma 2.6 and the proof of Theorem 2.1, we can derive the following ergodic rate for the fixed-point residual in Algorithm 2.1:

$$\tfrac{1}{k}\textstyle\sum_{i=0}^{k} \|\boldsymbol{\omega}^{k+1} - \boldsymbol{\omega}^k\|^2 = O(1/k)$$

This rate also holds for the iterations in (2.21), (2.22), (2.23); for the case of general operator splittings (and differently from optimization algorithms), tighter rates for accelerated schemes are only known for particular cases, and most works focus on mere convergence [77], [32]. Yet, the practice shows that relaxation and inertia often result in improved speed, see [14] or §2.7.

The same residual rate $O(1/k)$ can also be shown for the pseudo-gradient method in [109, Alg. 1]. However, a major difference from Lemma 2.5 is that the upper bounds for the step sizes in [109, Th. 2] are proportional to the constant μ_{F_a} in (2.8), which is typically very small (up to scaling of the whole operator $\boldsymbol{F}_a$), [25] (see also §2.7.1), and,

Algorithm 2.3. Fully-distributed NE seeking via PPPA

$$\check{\boldsymbol{x}}_{i,-i}^{k+1} = \tfrac{1}{1+\tau_i d_i}(\boldsymbol{x}_{i,-i}^k + \tau_i \textstyle\sum_{j=1}^N w_{i,j}\boldsymbol{x}_{j,-i}^k)$$
$$\check{x}_i^{k+1} = \operatorname*{argmin}_{y\in\Omega_i}\big(J_i(y,\check{\boldsymbol{x}}_{i,-i}^{k+1}) + \tfrac{1}{2\alpha\tau_i}\|y-x_i^k\|^2 + \tfrac{1}{2\alpha d_i}\|d_i y - \textstyle\sum_{j=1}^N w_{i,j}\boldsymbol{x}_{j,i}^k\|^2\big)$$
$$\boldsymbol{x}_i^{k+1} = \boldsymbol{x}_i^k + \gamma(\check{\boldsymbol{x}}_i^{k+1} - \boldsymbol{x}_i^k)$$

	FB [109, Alg. 1]	PPPA
step sizes	$O\left(\frac{\mu_{\boldsymbol{F}_\mathrm{a}}}{\theta_{\boldsymbol{F}_\mathrm{a}}{}^2+\mu_{\boldsymbol{F}_\mathrm{a}}}\right)$	$O(1)$
linear rate ρ (no coupling constraints)	$(1-\kappa_{\boldsymbol{F}_\mathrm{a}}{}^2)^{\frac{1}{2}}$	$1-\kappa_{\boldsymbol{F}_\mathrm{a}}$

Table 2.1: Comparison between our PPPA and projected pseudo-gradient methods.

most importantly, it vanishes as the number of agents increases (fixed the other parameters). In contrast, our algorithms allows for much larger steps, which can be chosen independently of the number of agents. This is a structural advantage of the PPA, whose convergence does not depend on the cocoercivity constant of the operators involved. Indeed, step sizes must be taken into account if convergence is evaluated in terms of residuals.

We finally note that linear convergence can be achieved via PPPA for games without coupling constraints. For instance, Algorithm 2.3 corresponds to the overrelaxed method in Algorithm 2.2, and can be derived, as in Lemma 2.6, by taking $\mathcal{B} = \Phi_{\mathrm{NE}}^{-1}\mathcal{A}_{\mathrm{NE}}(\boldsymbol{x})$ in (2.13), where $\mathcal{A}_{\mathrm{NE}}(\boldsymbol{x}) := \boldsymbol{F}_\mathrm{a}(\boldsymbol{x}) + \mathrm{N}_{\boldsymbol{\Omega}}(\boldsymbol{x})$ and $\Phi_{\mathrm{NE}} := \bar{\tau}^{-1} + \boldsymbol{W}_n$ are obtained by removing the dual variables from $\mathcal{A}$, Φ. By (2.9), as in Lemma 2.7, it can be shown that $\mathcal{A}_{\mathrm{NE}}$ is restricted $\frac{\mu_{\boldsymbol{F}_\mathrm{a}}}{\|\Phi_{\mathrm{NE}}\|}$-strongly monotone in $\mathcal{H}_{\Phi_{\mathrm{NE}}}$. Thus, recursively applying Theorem 2.1*(iv)*, we can infer the following result, which appeared in [21] only limited to $\gamma = 1$.

Theorem 2.3. Let $\tau_i^{-1} > d_i$ for all $i \in \mathcal{I}$, let $\gamma \in (0,2)$, and let $\alpha \in (0, \alpha_{\max}]$, with $\alpha_{\max}$ as in Lemma 2.3. Then, the sequence $(\boldsymbol{x}^k)_{k\in\mathbb{N}}$ generated by Algorithm 2.3 converges to $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, where x^* is the unique Nash equilibrium of the game in (2.1), with linear rate:

$$(\forall k \in \mathbb{N}) \quad \|\boldsymbol{x}^k - \boldsymbol{x}^*\|_{\Phi_{\mathrm{NE}}} \le (\rho_\gamma)^k \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_{\Phi_{\mathrm{NE}}},$$

where $\rho_\gamma := \max(1 - \frac{\gamma\mu_{\boldsymbol{F}_\mathrm{a}}}{\|\Phi_{\mathrm{NE}}\|+\mu_{\boldsymbol{F}_\mathrm{a}}}, \gamma - 1)$, $\mu_{\boldsymbol{F}_\mathrm{a}}$ as in (2.8). □

The best theoretical rate $\rho_{\bar{\gamma}} = 1 - 2\mu_{\boldsymbol{F}_\mathrm{a}}/(\|\Phi_{\mathrm{NE}}\| + 2\mu_{\boldsymbol{F}_\mathrm{a}})$ is obtained for $\bar{\gamma} = 1 + \|\Phi_{\mathrm{NE}}\|/(\|\Phi_{\mathrm{NE}}\| + 2\mu_{\boldsymbol{F}_\mathrm{a}})$. We observed in [21], also for numerical results. For instance, in the absence of coupling constraints, the FB algorithm in [109, Alg. 1] reduces to [136, Alg. 1], whose optimal linear rate $O((1-\kappa_{\boldsymbol{F}_\mathrm{a}}{}^2)^{\frac{k}{2}})$ depends *quadratically* on the quantity $\kappa_{\boldsymbol{F}_\mathrm{a}} := \mu_{\boldsymbol{F}_\mathrm{a}}/\theta_{\boldsymbol{F}_\mathrm{a}} < 1$ [136, Th. 7], where $\theta_{\boldsymbol{F}_\mathrm{a}} := 2\max((d_i)_{i\in\mathcal{I}}) + \alpha\theta$. Instead, $\rho_{\bar{\gamma}} \le 1 - \kappa_{\boldsymbol{F}_\mathrm{a}}$, for large enough τ_i's (since $\|\Phi_{\mathrm{NE}}\| + 2\mu_{\boldsymbol{F}_\mathrm{a}} \le \max((d_i + \tau_i^{-1})_{i\in\mathcal{I}}) + 2\alpha\theta$), as shown in Table 2.1.

2.6. AGGREGATIVE GAMES

Algorithm 2.4. Fully-distributed v-GNE seeking in aggregative games via PPPA

Initialization:

- For all $i \in \mathcal{I}$, set $x_i^0 \in \Omega_i$, $s_i^0 = \mathbf{0}_{\bar{n}}$, $z_i^0 = \mathbf{0}_m$, $\lambda_i^0 \in \mathbb{R}^m_{\geq 0}$.

For all $k > 0$:

- Communication: The agents exchange the variables $\{\sigma_i^k = x_i^k + s_i^k, \lambda_i^k\}$ with their neighbors.
- Local variables update: each agent $i \in \mathcal{I}$ computes

$$
\begin{aligned}
s_i^{k+1} &= s_i^k - \beta \textstyle\sum_{j=1}^N w_{i,j}(\sigma_i^k - \sigma_j^k) \\
x_i^{k+1} &\leftarrow y \ \text{ s.t. } \ \mathbf{0}_{\bar{n}} \in \alpha \tilde{\boldsymbol{F}}_i(y, y + s_i^{k+1}) + \tfrac{1}{\tau_i}(y - x_i^k) + A_i^\top \lambda_i^k + \textstyle\sum_{j=1}^N w_{i,j}(\sigma_i^k - \sigma_j^k) + \mathrm{N}_{\Omega_i}(y) \\
z_i^{k+1} &= z_i^k + \textstyle\sum_{j=1}^N \nu_{(i,j)} w_{i,j}(\lambda_i^k - \lambda_j^k) \\
\lambda_i^{k+1} &= \mathrm{proj}_{\mathbb{R}^m_{\geq 0}}\big(\lambda_i^k + \delta_i\big(A_i(2x_i^{k+1} - x_i^k) - b_i - (2z_i^{k+1} - z_i^k)\big)\big).
\end{aligned}
\tag{2.24}
$$

IN this section we focus on the particularly relevant class of (average) aggregative games, which arises in a variety of engineering applications, e.g., network congestion control and demand-side management [71]. In aggregative games, $n_i = \bar{n} > 0$ for all $i \in \mathcal{I}$ (hence $n = N\bar{n}$) and the cost function of each agent depends only on its local decision and on the value of the average strategy $\mathrm{avg}(x) := \frac{1}{N}\sum_{i\in\mathcal{I}} x_i$. Therefore, for each $i \in \mathcal{I}$, there is a function $f_i : \mathbb{R}^{\bar{n}} \times \mathbb{R}^{\bar{n}} \to \mathbb{R}$ such that the original cost function J_i in (2.1) can be written as

$$
J_i(x_i, x_{-i}) =: f_i(x_i, \mathrm{avg}(x)). \tag{2.25}
$$

Since an aggregative game is only a particular instance of the game in (2.1), all the considerations on the existence and uniqueness of a v-GNE and the equivalence with the KKT conditions in (2.3) are still valid.

Moreover, Algorithms 2.1 could still be used to compute a v-GNE. This would require each agent to keep (and exchange) an estimate of all other agents' action, i.e., a vector of $(N-1)\bar{n}$ components. In practice, however, the cost of each agent is only a function of the aggregative value $\mathrm{avg}(x)$, whose dimension $\bar{n}$ is independent of the number N of agents. To reduce communication and computation burden, in this section we introduce a PPPA specifically tailored to seek a v-GNE in aggregative games, that is scalable with the number of agents. The proposed iteration is shown in Algorithm 2.4, where the parameters α, β, and τ_i, δ_i for all $i \in \mathcal{I}$, $\nu_{(i,j)}$ for all $(i,j) \in \mathcal{E}$ have to be chosen appropriately, and we denote

$$
\tilde{\boldsymbol{F}}_i(x_i, \xi_i) := \nabla_{x_i} f_i(x_i, \xi_i) + \tfrac{1}{N}\nabla_{\xi_i} f_i(x_i, \xi_i). \tag{2.26}
$$

We note that $\tilde{\boldsymbol{F}}_i(x_i, \mathrm{avg}(x)) = \nabla_{x_i} J_i(x_i, x_{-i}) = \nabla_{x_i} f_i(x_i, \mathrm{avg}(x))$.

Because of the partial-decision information assumption, no agent has access to the actual value of the average strategy. Instead, we equip each agent with an auxiliary error variable $s_i \in \mathbb{R}^{\bar{n}}$, which is an estimate of the quantity $\mathrm{avg}(x) - x_i$. Each agent aims at reconstructing the true aggregate value, based on the information received from its neighbors. In particular, it should hold that $s^k \to \mathbf{1}_N \otimes \mathrm{avg}(x^k) - x^k$ asymptotically, where $s := \mathrm{col}((s_i)_{i\in\mathcal{I}})$. For brevity of notation, we also denote

$$\sigma_i := x_i + s_i, \quad \sigma := \mathrm{col}((\sigma_i)_{i\in\mathcal{I}}). \tag{2.27}$$

Remark 2.9. By the updates in Algorithm 2.4, we can infer an important invariance property, namely that $\mathrm{avg}(s^k) = \mathbf{0}_{\bar{n}}$, or equivalently $\mathrm{avg}(x^k) = \mathrm{avg}(\sigma^k)$, for any $k \in \mathbb{N}$, provided that the algorithm is initialized appropriately, i.e., $s_i^0 = \mathbf{0}_{\bar{n}}$, for all $i \in \mathcal{I}$. In fact, the update of σ, as it follows from Algorithm 2.4, is

$$\sigma^{k+1} = \sigma^k - \beta \boldsymbol{L}_{\bar{n}} \sigma^k + (x^{k+1} - x^k), \tag{2.28}$$

where $\boldsymbol{L}_{\bar{n}} := L \otimes I_{\bar{n}}$. This update is a dynamic tracking for the time-varying quantity $\mathrm{avg}(x)$, similar to those considered for aggregative games in [82], [10], [64]. Differently from [64], here we introduce the error variables s_i, which allow us to directly recast the iteration in (2.28) in an operator-theoretic framework. □

Similarly to §2.4, we study the convergence of Algorithm 2.4 by relating it to the iteration in (2.13). First, let us define the *extended pseudo-gradient* mapping

$$\tilde{\boldsymbol{F}}(x, \xi) := \mathrm{col}((\tilde{\boldsymbol{F}}_i(x_i, \xi_i))_{i\in\mathcal{I}}), \tag{2.29}$$

with $\xi := \mathrm{col}((\xi_i)_{i\in\mathcal{I}}) \in \mathbb{R}^n$, and the operators $\tilde{\boldsymbol{F}}_{\mathrm{a}}(x, s) := \mathrm{col}(\alpha \tilde{\boldsymbol{F}}(x, \sigma) + \boldsymbol{L}_{\bar{n}}\sigma, \boldsymbol{L}_{\bar{n}}\sigma)$,

$$\tilde{\mathcal{A}}(\boldsymbol{\omega}) := \begin{bmatrix} \alpha \tilde{\boldsymbol{F}}(x, \sigma) + \boldsymbol{L}_{\bar{n}}\sigma \\ \boldsymbol{L}_{\bar{n}}\sigma \\ \mathbf{0}_{Em} \\ \boldsymbol{b} \end{bmatrix} + \begin{bmatrix} \boldsymbol{A}^\top \boldsymbol{\lambda} \\ \mathbf{0}_n \\ -\boldsymbol{V}_m \boldsymbol{\lambda} \\ \boldsymbol{V}_m^\top \boldsymbol{\nu} - \boldsymbol{A}x \end{bmatrix} + \begin{bmatrix} \mathrm{N}_\Omega(x) \\ \mathbf{0}_n \\ \mathbf{0}_{Em} \\ \mathrm{N}_{\mathbb{R}^{Nm}_{\geq 0}}(\boldsymbol{\lambda}) \end{bmatrix}, \tag{2.30}$$

where $\boldsymbol{\omega} := \mathrm{col}(x, s, \boldsymbol{\nu}, \boldsymbol{\lambda}) \in \mathbb{R}^{2n+Em+Nm}$, and we recall that $\sigma = x + s$ is just a shorthand notation.

Lemma 2.8. The mapping $\tilde{\boldsymbol{F}}$ in (2.29) is $\tilde{\theta}$-Lipschitz continuous, for some $\tilde{\theta} > 0$. □

Proof. It follows from Lemma 2.2, by noticing that $\tilde{\boldsymbol{F}}(x, \xi) = \boldsymbol{F}((x, (I_N \otimes \mathbf{1}_{N-1} \otimes I_{\bar{n}})(\frac{N}{N-1}\xi - \frac{1}{N-1}x)))$. ■

Finally, we will assume that the step sizes $\bar{\tau} := \mathrm{diag}((\tau_i I_{\bar{n}})_{i\in\mathcal{I}})$, $\bar{\beta} := \beta I_{Nn}$, $\bar{\nu} := \mathrm{diag}((\nu_{(i,j)})_{(i,j)\in\mathcal{E}})$, $\bar{\delta} := \mathrm{diag}((\delta_i I_m)_{i\in\mathcal{I}})$ are chosen such that $\tilde{\Phi} \succ 0$, where

$$\tilde{\Phi} := \begin{bmatrix} \bar{\tau}^{-1} - \boldsymbol{L}_{\bar{n}} & -\boldsymbol{L}_{\bar{n}} & \mathbf{0} & -\boldsymbol{A}^\top \\ -\boldsymbol{L}_{\bar{n}} & \bar{\beta}^{-1} - \boldsymbol{L}_{\bar{n}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \bar{\nu}^{-1} & \boldsymbol{V}_m \\ -\boldsymbol{A} & \mathbf{0} & \boldsymbol{V}_m^\top & \bar{\delta}^{-1} \end{bmatrix}. \tag{2.31}$$

Lemma 2.9. The matrix $\tilde{\Phi}$ in (2.31) is positive definite if $\beta^{-1} > 4\max((d_i)_{i\in\mathcal{I}})$, ${\nu_{(i,j)}}^{-1} > 2\sqrt{(w_{i,j})}$ for all $(i,j)\in\mathcal{E}$, and $\tau_i^{-1} > 4d_i + \|A_i^\top\|_\infty$, $\delta_i^{-1} > \|A_i\|_\infty + \sum_{j=1}^N \sqrt{w_{i,j}}$ for all $i\in\mathcal{I}$. □

Theorem 2.4. Let $d_{\min} := \min((d_i)_{i\in\mathcal{I}})$ and

$$\tilde{\alpha}_{\max} := \min\left(\tfrac{4\mu\lambda_2(L)}{\tilde{\theta}^2}, \tfrac{2\sqrt{2}(d_{\min})}{\tilde{\theta}}\right). \tag{2.32}$$

Let $\alpha\in(0,\tilde{\alpha}_{\max}]$ and let the step sizes $\bar{\tau},\bar{\beta},\bar{\nu},\bar{\delta}$ be as in Lemma 2.9. Then, for all $k\in\mathbb{N}$, the inclusion in (2.24) has a unique solution. Moreover, the sequence $(x^k,s^k,\boldsymbol{z}^k,\boldsymbol{\lambda}^k)_{k\in\mathbb{N}}$ generated by Algorithm 2.4 converges to an equilibrium $(x^*,\mathbf{1}\otimes\operatorname{avg}(x^*)-x^*,\boldsymbol{z}^*,\boldsymbol{\lambda}^*)$, where x^* is the v-GNE of the game in (2.1). □

Proof. Similarly to Lemma 2.6, we first show that Algorithm 2.4 can be recast as a PPPA, applied to find a zero of the operator $\tilde{\Phi}^{-1}\tilde{A}$. Then, we restrict our analysis to the invariant subspace

$$\Sigma := \{(x,s,\boldsymbol{\nu},\boldsymbol{\lambda})\in\mathbb{R}^{2n+Em+Nm} \mid \operatorname{avg}(s)=\mathbf{0}_{\bar{n}}\}. \tag{2.33}$$

A detailed proof is in Section 2.9.2. ■

Remark 2.10. The update in (2.24) is implicitly defined by a strongly monotone inclusion, or, equivalently, variational inequality (see Section 2.9.2). We emphasize that there are several iterative methods to find the unique solution (with linear rate) [8, §26] and that, as in Remark 2.8, convergence is guaranteed even if the solution is approximated at each step (with summable errors). □

Remark 2.11. If, for some $i\in\mathcal{I}$, there exists a function φ_i such that $\nabla_y\varphi_i(y,s_i^{k+1}) = \tilde{\boldsymbol{F}}_i(y,y+s_i^{k+1})$, then the update of x_i^k in Algorithm 2.4 can be simplified as

$$x_i^{k+1} = \operatorname*{argmin}_{y\in\Omega_i}\big(\varphi_i(y,s_i^{k+1}) + \tfrac{1}{2\alpha\tau_i}\|y-x_i^k\|^2 + \tfrac{1}{\alpha}(A_i^\top\lambda_i^k)^\top y + \tfrac{1}{\alpha}\big(\textstyle\sum_{j=1}^N w_{ij}(\sigma_i^k-\sigma_j^k)\big)^\top y\big),$$

as in Lemma 2.6. For scalar games (i.e., $\bar{n}=1$) this condition holds for all $i\in\mathcal{I}$. Another noteworthy example is that of a cost $f_i(x_i,\operatorname{avg}(x)) = \bar{f}_i(x_i) + (Q_i\operatorname{avg}(x))^\top x_i$, for some function $\bar{f}_i$ and symmetric matrix Q_i, which models applications as the Nash–Cournot game described in [82] and the resource allocation problem considered in [9]. In this case, $\varphi_i(x_i,s_i) = \bar{f}_i(x_i) + (Q_i(s_i+x_i))^\top x_i - \frac{N-1}{2N}x_i^\top Q_i x_i$. □

Remark 2.12. Inertial/relaxed versions of Algorithm 2.4 can be studied as in §2.5; further, linear convergence can be established for aggregative games without coupling constraints, based on the restricted strong monotonicity of $\tilde{\boldsymbol{F}}_{\mathrm{a}}$ (see the proof of Lemma 2.11 in Section 2.9.2), as in Theorem 2.3. □

2.7. NUMERICAL SIMULATIONS

2.7.1. NASH–COURNOT GAME

We consider a Nash–Cournot game [109, §6], where N firms produce a commodity that is sold to m markets. Each firm $i\in\mathcal{I}=\{1,\dots,N\}$ participates in $n_i\le m$ of the markets,

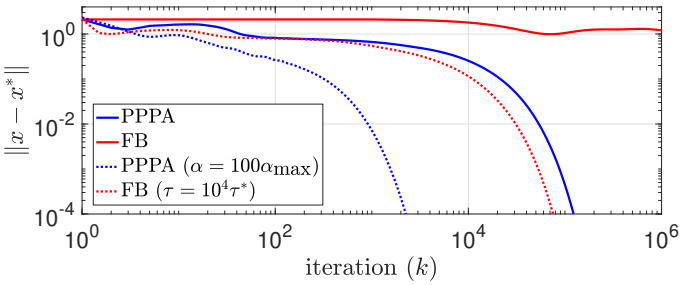

Figure 2.1: Distance from the v-GNE, for our PPPA (Algorithm 2.1) and the FB algorithm in [109, Alg. 1] for different parameters (the solid line for the theoretical step sizes).

and decides on the quantities $x_i \in \mathbb{R}^{n_i}$ of commodity to be delivered to these n_i markets. The quantity of product that each firm can deliver is bounded by the local constraints $\mathbf{0}_{n_i} \leq x_i \leq X_i$. Moreover, each market $k = 1, \ldots, m$ has a maximal capacity r_k. This results in the shared affine constraint $Ax \leq r$, with $r = \mathrm{col}((r_k)_{k=1,\ldots,m})$ and $A = [A_1 \ldots A_N]$, where $A_i \in \mathbb{R}^{m \times n_i}$ is the matrix that expresses which markets firm i participates in. Specifically, $[A_i]_{k,j} = 1$ if $[x_i]_j$ is the amount of product sent to the k-th market by agent i, $[A_i]_{k,j} = 0$ otherwise, for all $j = 1, \ldots, n_i$, $k = 1, \ldots, m$. Hence, $Ax = \sum_{i=1}^{N} A_i x_i \in \mathbb{R}^m$ is the vector of the quantities of total product delivered to the markets. Each firm i aims at maximizing its profit, i.e., minimizing the cost function $J_i(x_i, x_{-i}) = 10^{-3} * (c_i(x_i) - p(Ax)^\top A_i x_i)$. Here, $c_i(x_i) = x_i^\top Q_i x_i + q_i^\top x_i$ is firm i's production cost, with $Q_i \in \mathbb{R}^{n_i \times n_i}$, $Q_i \succ 0$, $q_i \in \mathbb{R}^{n_i}$. Instead, $p : \mathbb{R}^m \to \mathbb{R}^m$ associate to each market a price that depends on the amount of product delivered to that market. Specifically, the price for the market k, for $k = 1, \ldots, m$, is $[p(x)]_k = \bar{P}_k - \chi_k [Ax]_k$, where $\bar{P}_k, \chi_k > 0$.

We set $N = 20$, $m = 7$. The market structure (i.e., which firms are allowed to participate in which of the m markets) is defined as in [109, Fig. 1]; thus $x = \mathrm{col}((x_i)_{i \in \mathcal{I}}) \in \mathbb{R}^n$ and $n = 32$. The firms cannot access the production of all the competitors, but they are allowed to communicate with their neighbors on a randomly generated connected graph. We select randomly with uniform distribution r_k in $[1, 2]$, Q_i diagonal with diagonal elements in $[1, 8]$, q_i in $[1, 2]$, $\bar{P}_k$ in $[10, 20]$, χ_k in $[1, 3]$, X_i in $[5, 10]$, for all $i \in \mathcal{I}$, $k = 1, \ldots, m$.

The resulting setup satisfies all our theoretical assumptions [109, §VI]. We set $\alpha = \alpha_{\max} \approx 0.7$ as in Lemma 2.3 and we choose the step sizes as in Lemma 2.5 to satisfy all the conditions of Theorem 2.2.

We compare the performance of Algorithm 2.1 versus that of the pseudo-gradient method in [109, Alg. 1], which is to the best of our knowledge the only other available single-layer fixed-step scheme to solve GNE problems under partial-decision information. In [109, Alg. 1], we choose the parameter c that maximize the step sizes τ, ν, σ, provided that the conditions in [109, Th. 2] are satisfied. This results in very small step sizes, e.g., $\tau^* \approx 10^{-5}$.

The results are illustrated in Figure 2.1, where the two Algorithms are initialized with the same random initial conditions. [109, Alg. 1] is extremely slow, due to the small step sizes; and our PPPA method shows a much faster convergence. According to our numer-

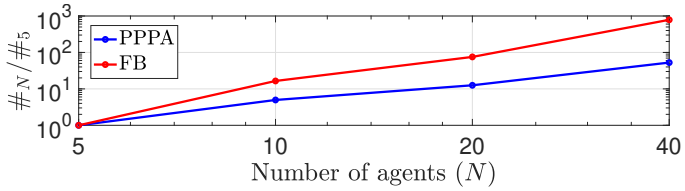

Figure 2.2: Variation of the number of iterations $\#_N$ needed to reach a precision of $\|x^k - x^*\| \le 10^{-2}$ for different values of the number of agents N (in logarithmic scale): our PPPA (Algorithm 2.1) versus the FB algorithm in [109, Alg. 1]

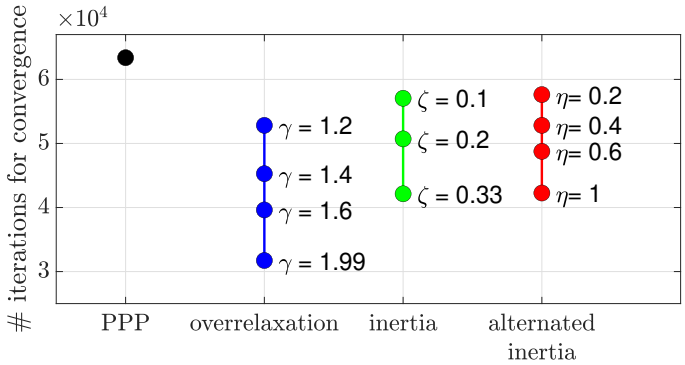

Figure 2.3: Number of iterations needed to reach a precision of $\|x^k - x^*\| \le 10^{-2}$, with different acceleration schemes and parameters.

ical experience, the bounds on the parameters are conservative, and in effect we observe faster convergence for larger step sizes. For [109, Alg. 1], the fastest convergence is attained by setting the step sizes 10^4 times bigger than the theoretical bounds; for larger steps, convergence is lost.

We repeat the simulation for different numbers of agents (and random market structures). Differently from Algorithm 1, the upper bounds for the step sizes in [109, Alg. 1] decrease when N grows (see §2.5.1), resulting in a greater performance degradation, as shown in Figure 2.2 (with theoretical parameters for our PPPA, and steps 10^3 times larger than their upper bounds for [109, Alg. 1]).

Finally, we apply the acceleration schemes discussed in Section 2.5 to Algorithm 2.1, with parameters that theoretically ensure convergence. The impact is remarkable, up to halving the number of iterations needed for convergence, as shown in Figure 2.3.

2.7.2. CHARGING OF PLUG-IN ELECTRIC VEHICLES

We consider the charging scheduling problem for a group of plug-in electric vehicles, modeled by an aggregative game [71]. Each user $i \in \mathcal{I} = \{1, \dots, N\}$ plans the charging of its vehicle for an horizon of 24 hours, discretized into $\bar{n}$ intervals; the goal is to choose the energy injections $x_i \in \mathbb{R}^{\bar{n}}$ of each time interval to minimize its cost $J_i(x_i, \mathrm{avg}(x)) = g_i(x_i) + p(\mathrm{avg}(x))^\top x_i$, where $g_i(x_i) = x_i^\top Q_i x_i + c_i^\top x^i$ is the battery degradation cost, and $p(\xi) = a(\xi + d) + b\mathbf{1}_{\bar{n}}$ is the cost of energy, with b a baseline price, a the inverse of the price

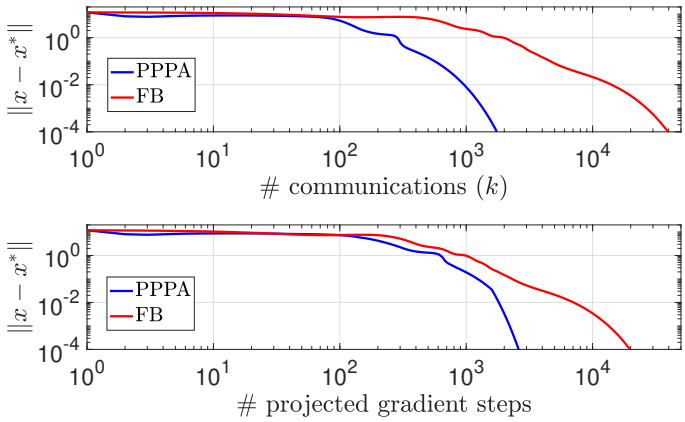

Figure 2.4: Distance of the primal variable from the v-GNE. Our PPPA (Algorithm 2.4) outperforms the FB algorithm in [64, Alg. 1], in terms of both communication rounds and performed projected gradient steps.

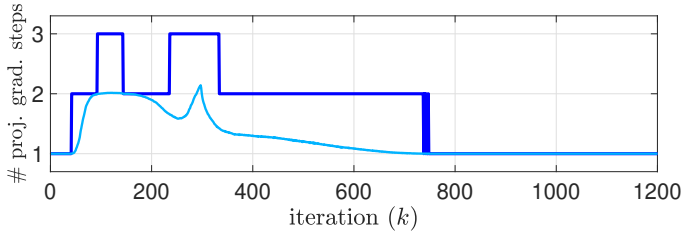

Figure 2.5: Maximum (blue) and average (light blue) number of projected gradient steps performed by the agents at each iteration in Algorithm 2.4, with guaranteed accuracy of $\varepsilon^k = 1/k^2$.

elasticity and $d \in \mathbb{R}^{\bar{n}}$ the inelastic demand (not related to vehicle charging) along the horizon. We assume a maximum injection per interval and a desired final charge level for each user, resulting in the local constraints $\Omega_i = \{y \in [0_{\bar{n}}, \bar{x}_i] \mid \mathbf{1}_{\bar{n}}^\top y = \gamma_i\}$. Moreover, we consider the transmission line constraints $\mathbf{0}_{\bar{n}} \leq \sum_{i\in\mathcal{I}} x_i \leq \bar{c}N$.

We set $N = 1000$, $\bar{n} = 12$. For all $i \in \mathcal{I}$, we select with uniform distribution c_i in $[0.55, 0.95]$, $Q_i \succ 0$ with diagonal and off-diagonal elements in $[0.2, 0.8]$ and $[0, 0.05]$, respectively, γ_i in $[0.6, 1]$; $[\bar{x}_i]_j = 0.25$ with probability 20%, $[\bar{x}_i]_j = 0$ otherwise. We set $[\bar{c}]_j$ as 0.04 if $j \in \{1, 2, 3, 11, 12\}$, as 0.01 otherwise (corresponding to more restrictive limitations in the daytime); $a = 0.38$, $b = 0.6$ and d as in [71]. We check numerically that Standing Assumptions 2.1, 2.2 hold, and let the agents communicate over a randomly generated connected graph. We implement Algorithm 2.4, by performing only a finite number of gradient steps per iteration; each agent uses the stopping criterion in (2.19) to ensure an accuracy of $\varepsilon^k = 1/k^2$. Figure 2.4 compares the performance of Algorithm 2.4 and [64, Alg. 1] (which requires two rounds of communication per iteration), with step sizes set to their theoretical upper bounds. Notably, our PPPA significantly outperforms [64, Alg. 1], even in terms of total projected gradient steps required (for Algorithm 2.4, we consider the maximum per iteration). Interestingly, Figure 2.5 shows that the maximum

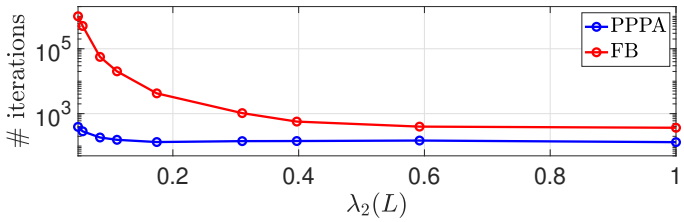

Figure 2.6: Number of iterations to reach a precision of $\|x - x^*\| \leq 10^{-2}$ for different values of the algebraic connectivity, where $\lambda_2(L) = 1$ indicates a complete graph (all the graphs are doubly stochastic): our PPPA (Algorithm 2.4) versus the FB in [64, Alg. 1].

number of performed gradient steps at each iteration is 3 and decreases as the iteration converges, despite the increasing accuracy required in the local optimizations (see also Remark 2.8).

Differently from our PPPA, the upper bounds for the step sizes in [64] are proportional to the quantity $\mu_{\tilde{A}}$ in [64, Lem. 4], hence they depend on $\lambda_2(L)$, θ_0, μ, θ (but not on N, cf. § 2.7.1, 2.5.1); in turn, we expect these parameters to affect to a larger extent the convergence speed for the FB method. In Figure 2.6 we compare the two algorithms, with $N = 10$, for different values of the communication graph connectivity: in the considered range, the number of iterations to converge varies by a factor 2 for Algorithm 2.4, by a factor 10^3 for [64, Alg. 1].

2.8. CONCLUSION

INEXACT preconditioned proximal-point methods are extremely efficient to design fully-distributed single-layer generalized Nash equilibrium seeking algorithms. The advantage is that convergence can be guaranteed for much larger step sizes compared to pseudo-gradient-based algorithms. In fact, in our numerical experience, our algorithms proved much faster than the existing methods, resulting in a considerable reduction of communication and computation requirements. Besides, our operator-theoretic approach facilitates the design of acceleration schemes, also in the partial-decision information setup. As future work, it would be highly valuable to relax our monotonicity and connectivity assumptions, namely to allow for merely monotone game mappings and jointly connected networks, and to address the case of nonlinear coupling constraints.

2.9. APPENDIX

2.9.1. PROOF OF THEOREM 2.1

For all $k \in \mathbb{N}$, let $z^k := \omega^k + \gamma^k(u^k - \omega^k)$, so that $\omega^{k+1} = z^k + \gamma^k e^k$. Consider any $\omega^* \in C$. We have, for all $k \in \mathbb{N}$,

$$\begin{aligned} &\|z^k - \omega^*\|_P^2 \\ &= \|\omega^k - \omega^*\|_P^2 - 2\gamma^k \langle \omega^k - u^k \mid \omega^k - \omega^* \rangle_P + (\gamma^k)^2 \|u^k - \omega^k\|_P^2 \\ &\le \|\omega^k - \omega^*\|_P^2 - \gamma^k(2-\gamma^k)\|u^k - \omega^k\|_P^2, \end{aligned} \tag{2.34}$$

where the inequality follows by Lemma 2.4.
(i) By (2.34), $\|z^k - \omega^*\|_P \le \|\omega^k - \omega^*\|_P$, and the conclusion follows by the Cauchy–Schwartz inequality.
(ii) By $(\gamma^k \|e^k\|_P)_{k\in\mathbb{N}} \in \ell^1$ and point *(i)*, $(\omega^k)_{k\in\mathbb{N}}$ is bounded. Let $c := \sup_{k\in\mathbb{N}} \|\omega^k - \omega^*\|_P < \infty$ and $\epsilon^k := 2c(\gamma^k \|e^k\|_P) + (\gamma^k \|e^k\|_P)^2$, for all $k \in \mathbb{N}$. Clearly, $(\epsilon^k)_{k\in\mathbb{N}} \in \ell^1$. Moreover, for all $k \in \mathbb{N}$ we have

$$\begin{aligned} &\|w^{k+1} - w^*\|_P^2 \\ &\le (\|z^k - w^*\|_P + \gamma^k \|e^k\|_P)^2 \\ &\le \|w^k - w^*\|_P^2 - \gamma^k(2-\gamma^k)\|u^k - w^k\|_P^2 + \epsilon^k, \end{aligned} \tag{2.35}$$

and the thesis follows by recursion.
(iii) By (2.35), [41, Prop. 3.2(i)] and [41, Th. 3.8].
(iv) By definition of resolvent, $\omega^k - u^k \in \mathcal{B}(u^k)$; hence

$$\langle u^k - \omega^* \mid \omega^k - u^k \rangle_P \ge \mu_{\mathcal{B}} \|u^k - \omega^*\|_P^2. \tag{2.36}$$

By the Cauchy–Schwartz inequality, $\|\omega^k - u^k\|_P \ge \mu_{\mathcal{B}} \|u^k - \omega^*\|_P$. Thus, (2.36) yields

$$\begin{aligned} &\|\omega^k - \omega^*\|_P^2 \\ &= \|u^k - \omega^*\|_P^2 + 2\langle u^k - \omega^* \mid \omega^k - u^k \rangle_P + \|\omega^k - u^k\|_P^2 \\ &\ge (1+\mu_{\mathcal{B}})^2 \|u^k - \omega^*\|_P^2. \end{aligned} \tag{2.37}$$

If $\gamma^k \le 1$, by the Cauchy–Schwartz inequality and (2.37), we have $\|z^k - \omega^*\|_P \le (1 - \gamma^k)\|\omega^k - \omega^*\|_P + \gamma^k \|u^k - \omega^*\|_P \le (1 - \frac{\gamma^k \mu_{\mathcal{B}}}{1+\mu_{\mathcal{B}}})\|\omega^k - \omega^*\|_P$. For $\gamma^k > 1$, we can write

$$\begin{aligned} &\|z^k - \omega^*\|_P^2 \\ &= (1-\gamma^k)^2 \|\omega^k - \omega^*\|_P^2 + \gamma^k(2-\gamma^k)\|u^k - \omega^*\|_P^2 \\ &\quad + 2\gamma^k(1-\gamma^k)\langle u^k - \omega^* \mid \omega^k - u^k \rangle_P \\ &\le (1-\gamma^k)^2 \|\omega^k - \omega^*\|_P^2 \\ &\quad + \gamma^k(2(1+\mu_{\mathcal{B}}) - \gamma^k(1+2\mu_{\mathcal{B}}))\|u^k - \omega^*\|_P^2 \qquad (2.38) \\ &\le (\max(1 - \tfrac{\gamma^k \mu_{\mathcal{B}}}{1+\mu_{\mathcal{B}}}, \gamma^k - 1))^2 \|\omega^k - \omega^*\|_P^2, \qquad (2.39) \end{aligned}$$

where the first equality follows by rearranging the terms in (2.34); in the first inequality we used (2.36); the last inequality follows by taking into account that the second term in (2.38) is nonpositive if $\gamma^k \in (1, 1+\frac{1}{1+2\mu_{\mathcal{B}}}]$ and can be upper bounded via (2.37) if $\gamma^k \in [1+\frac{1}{1+2\mu_{\mathcal{B}}}, 2)$. Finally, assume that $\omega^k \in C$, and choose $u^k = \omega^k$. Then (2.39) implies $\omega^k = \omega^*$, hence C must be a singleton. ■

2.9.2. PROOF OF THEOREM 2.4

Analogously to Lemma 2.6, it can be shown that Algorithm 2.4 is equivalent to the iteration

$$\boldsymbol{\omega}^{k+1} \in \mathrm{J}_{\tilde{\Phi}^{-1}\tilde{\mathcal{A}}}(\boldsymbol{\omega}^k), \quad \boldsymbol{\omega}^0 = \bar{\boldsymbol{\omega}}^0, \tag{2.40}$$

where $\bar{\boldsymbol{\omega}}^0 = (\boldsymbol{x}^0, \mathbf{0}_n, \mathbf{0}_{Em}, \boldsymbol{\lambda}^0)$, for some $\boldsymbol{x}^0 \in \Omega$, $\boldsymbol{\lambda}^0 \in \mathbb{R}^{Nm}_{\geq 0}$, modulo the transformation $\boldsymbol{z}^k = \boldsymbol{V}_m^\top \boldsymbol{v}^k$.

First, we show that the iteration in (2.40) is uniquely defined. For all $i \in \mathcal{I}$, let $\mathcal{F}_i(y, \vartheta^k) := \alpha \tilde{\boldsymbol{F}}_i(y, y + s_i^{k+1}) + \frac{1}{\tau_i}(y - x_i^k) + A_i^\top \lambda_i^k + \sum_{j=1}^N w_{i,j}(\sigma_i^k - \sigma_j^k) + \mathrm{N}_{\Omega_i}(y)$, where $\vartheta^k = (x^k, s^{k+1}, s^k, \boldsymbol{\lambda}^k)$. We note that $\tilde{\boldsymbol{F}}_i$ is $\tilde{\theta}$-Lipschitz, because $\tilde{\boldsymbol{F}}$ is $\tilde{\theta}$-Lipschitz by Lemma 2.8. Then, by monotonicity of the normal cone, we have $\langle y - y' \mid \mathcal{F}_i(y, \vartheta^k) - \mathcal{F}_i(y', \vartheta^k)\rangle \geq (\tau_i^{-1} - \alpha\sqrt{2}\tilde{\theta})\|y - y'\|^2$, for any $y, y' \in \mathbb{R}^n$, for any ϑ^k. By the assumption on α, $\mathcal{F}_i$ is strongly monotone in y for any ϑ^k, hence the inclusion in (2.24) has a unique solution, for any ϑ^k [8, Cor. 23.37]. Therefore, it also holds that $\mathrm{dom}(\mathrm{J}_{\tilde{\Phi}^{-1}\tilde{\mathcal{A}}}) = \mathbb{R}^{2n+Em+Nn}$ and that $\mathrm{J}_{\tilde{\Phi}^{-1}\tilde{\mathcal{A}}}$ is single-valued.

We turn our attention to the set Σ in (2.33). As in Remark 2.9, for any $\varsigma \in \Sigma$, $J_{\tilde{\Phi}^{-1}\tilde{\mathcal{A}}}(\varsigma) \in \Sigma$; hence Σ is invariant for (2.40). Moreover, $\boldsymbol{\omega}^0 \in \Sigma$. Hence, in (2.40), it is enough to consider the operator $J_{\tilde{\Phi}^{-1}\tilde{\mathcal{A}}}|_\Sigma \colon \Sigma \to \Sigma$, where $\mathcal{B}|_\Sigma$ is the restriction of the operator $\mathcal{B}$ to Σ, i.e., $\mathcal{B}|_\Sigma(\boldsymbol{\omega}) = \mathcal{B}(\boldsymbol{\omega})$ if $\boldsymbol{\omega} \in \Sigma$, $\mathcal{B}|_\Sigma(\boldsymbol{\omega}) = \varnothing$ otherwise. By invariance and (2.16), it also follows that $J_{\tilde{\Phi}^{-1}\tilde{\mathcal{A}}}|_\Sigma = J_{\tilde{\Phi}^{-1}\tilde{\mathcal{A}}|_\Sigma}|_\Sigma$. Thus, the iteration in (2.40) is rewritten as

$$\boldsymbol{\omega}^{k+1} = \mathrm{J}_{\tilde{\Phi}^{-1}\tilde{\mathcal{A}}|_\Sigma}(\boldsymbol{\omega}^k), \quad \boldsymbol{\omega}^0 = \bar{\boldsymbol{\omega}}^0. \tag{2.41}$$

We show the convergence of (2.41) by studying the properties of $\tilde{\mathcal{A}}|_\Sigma$. We start by characterizing the zero set.

Lemma 2.10. The following statements holds:

(i) If $\mathrm{col}(x^*, s^*, \boldsymbol{v}^*, \boldsymbol{\lambda}^*) \in \mathrm{zer}(\tilde{\mathcal{A}}|_\Sigma)$, then $s^* = \mathbf{1}_N \otimes \mathrm{avg}(x^*) - x^*$ and x^* is the v-GNE of the game in (2.1).

(ii) $\mathrm{zer}(\tilde{\mathcal{A}}|_\Sigma) \neq \varnothing$. □

Proof. Let $\boldsymbol{V}_q := V \otimes I_q$, $\boldsymbol{L}_q := L \otimes I_q = \boldsymbol{V}_q^\top \boldsymbol{V}_q$, for any $q > 0$; hence, under Standing Assumption 2.3, we have

$$\mathrm{null}\left(\boldsymbol{L}_q\right) = \mathrm{null}\left(\boldsymbol{V}_q\right) \quad = \mathrm{range}(\mathbf{1}_N \otimes I_q) \tag{2.42}$$

$$\mathrm{range}(\boldsymbol{V}_q^\top) \supseteq \mathrm{range}\left(\boldsymbol{L}_q\right) = \mathrm{null}(\mathbf{1}_N^\top \otimes I_q). \tag{2.43}$$

(i) Let us consider any $\boldsymbol{\omega}^* = \mathrm{col}(x^*, s^*, \boldsymbol{\nu}^*, \boldsymbol{\lambda}^*) \in \mathrm{zer}(\tilde{\mathcal{A}}|_\Sigma)$, and let $\sigma^* = x^* + s^*$; then we have

$$\mathbf{0}_{\tilde{n}} \in \alpha \tilde{\boldsymbol{F}}(x^*, \sigma^*) + \boldsymbol{L}_{\tilde{n}} \sigma^* + \mathrm{N}_\Omega(x^*) + \boldsymbol{A}^\top \boldsymbol{\lambda}^* \tag{2.44a}$$

$$\mathbf{0}_{\tilde{n}} = \boldsymbol{L}_{\tilde{n}} \sigma^* \tag{2.44b}$$

$$\mathbf{0}_{Em} = -\boldsymbol{V}_m \boldsymbol{\lambda}^* \tag{2.44c}$$

$$\mathbf{0}_{Nm} \in \boldsymbol{b} + \mathrm{N}_{\mathbb{R}^{Nm}_{\geq 0}}(\boldsymbol{\lambda}^*) - \boldsymbol{A} x^* + \boldsymbol{V}_m^\top \boldsymbol{\nu}^* \tag{2.44d}$$

By (2.44c) and by (2.42), we have $\boldsymbol{\lambda}^* = \mathbf{1}_N \otimes \lambda^*$, for some $\lambda^* \in \mathbb{R}^m$; by (2.44b) and since $\boldsymbol{\omega}^* \in \Sigma$, it must hold $\sigma^* = x^* + s^* = 1_N \otimes \mathrm{avg}(x^*)$. It is then enough to prove that the pair (x^*, λ^*) satisfies the KKT conditions in (2.3). By (2.44a), by recalling that $\boldsymbol{A}^\top (\mathbf{1}_N \otimes \lambda^*) = A^\top \lambda^*$ and $\tilde{\boldsymbol{F}}(x^*, \mathbf{1}_N \otimes x^*) = F(x^*)$, we retrieve the first KKT condition in (2.3). We obtain the second KKT condition by left-multiplying both sides of (2.44d) with $(\mathbf{1}_N^\top \otimes I_m)$ and using that $(\mathbf{1}_N^\top \otimes I_m)\boldsymbol{b} = b$, $(\mathbf{1}_N^\top \otimes I_m)\boldsymbol{L}_m = 0$ by (2.42) and symmetry of L, $(\mathbf{1}_N^\top \otimes I_m)\boldsymbol{A} = A$ and $(\mathbf{1}_N^\top \otimes I_m) N_{\mathbb{R}^{Nm}_{\geq 0}}(\mathbf{1}_N \otimes \lambda^*) = N \mathrm{N}_{\mathbb{R}^m_{\geq 0}}(\lambda^*) = \mathrm{N}_{\mathbb{R}^m_{\geq 0}}(\lambda^*)$. *(ii)* Let us consider any pair (x^*, λ^*) satisfying the KKT conditions in (2.3) (one such pair exists by Assumption 2.2). We next show that there exists $\boldsymbol{\nu}^* \in \mathbb{R}^{Em}$ such that $\boldsymbol{\omega}^* = \mathrm{col}(x^*, \mathbf{1}_N \otimes \mathrm{avg}(x^*) - x^*, \boldsymbol{z}^*, \mathbf{1}_N \otimes \lambda^*) \in \mathrm{zer}(\tilde{\mathcal{A}}|_\Sigma)$. Clearly, $\boldsymbol{\omega}^* \in \Sigma$. Besides, $\boldsymbol{\omega}^*$ satisfies the conditions (2.44a)-(2.44c), as in point *(i)*. By (2.3), there exists $u^* \in \mathrm{N}_{\mathbb{R}^m_{\geq 0}}(\lambda^*)$ such that $Ax^* - b - u^* = \mathbf{0}_n$. Also, $\mathrm{N}_{\mathbb{R}^{Nm}_{\geq 0}}(1_N \otimes \lambda^*) = \prod_{i \in \mathcal{I}} \mathrm{N}_{\mathbb{R}^m_{\geq 0}}(\lambda^*)$, and it follows by properties of cones that $\mathrm{col}\left(u_1^*, \ldots, u_N^*\right) \in \mathrm{N}_{\mathbb{R}^{Nm}_{\geq 0}}(\mathbf{1}_N \otimes \lambda^*)$, with $u_1^* = \cdots = u_N^* = \frac{1}{N} u^*$. Hence $(\mathbf{1}_N^\top \otimes I_m)\left(-\boldsymbol{A} x^* + \boldsymbol{b} + \mathrm{col}\left(u_1^*, \ldots, u_N^*\right)\right) = b - Ax^* + u^* = \mathbf{0}_m$, or $-\boldsymbol{A} x^* + \boldsymbol{b} + \mathrm{col}\left(u_1^*, \ldots, u_N^*\right) \in \mathrm{null}(\mathbf{1}_N^\top \otimes I_m) \subseteq \mathrm{range}(\boldsymbol{V}_m^\top)$, by (2.43). Therefore there exists $\boldsymbol{\nu}^*$ such that also the condition (2.44d) is satisfied, for which $\boldsymbol{\omega}^* \in \mathrm{zer}(\tilde{\mathcal{A}})$. ■

Next, similar to Lemma 2.3, we show restricted monotonicity of the operator $\tilde{\mathcal{A}}|_\Sigma$.

Lemma 2.11. Let $\alpha \in (0, \tilde{\alpha}_{\max}]$, with $\tilde{\alpha}_{\max}$ as in (2.32). Then $\tilde{\mathcal{A}}|_\Sigma$ is restricted monotone.

Proof. The operator $\tilde{\mathcal{A}}|_\Sigma$ is the sum of three components, as in (2.30). The third is monotone by properties of the normal cones [8, Th. 20.25], the second because it is a linear skew-symmetric operator [8, Ex. 20.35] (and restriction does not cause loss of monotonicity, by definition). For the first term, let $(\boldsymbol{\omega}, \boldsymbol{u}) \in \mathrm{gra}(\tilde{\mathcal{A}}|_\Sigma)$, $\boldsymbol{\omega} := \mathrm{col}(x, s, \boldsymbol{\nu}, \boldsymbol{\lambda})$, $\boldsymbol{\omega}^* = \mathrm{col}(x^*, s^*, \boldsymbol{\nu}^*, \boldsymbol{\lambda}^*) \in \mathrm{zer}(\tilde{\mathcal{A}}|_\Sigma)$, $\sigma = x + s$, $\sigma^* = s^* + x^*$. By Lemma 2.10, $s^* = \mathbf{1}_N \otimes \mathrm{avg}(x^*) - x^*$. Then, by [64, Lemma 4], there is a $\tilde{\mu} > 0$ such that $\langle \mathrm{col}(x - x^*, s - s^*) \mid \tilde{\boldsymbol{F}}_{\mathrm{a}}(x, s) - \tilde{\boldsymbol{F}}_{\mathrm{a}}(x^*, s^*) \rangle = \langle x - x^* \mid \alpha \tilde{\boldsymbol{F}}(x, \sigma) - \alpha \tilde{\boldsymbol{F}}(x^*, \sigma^*) \rangle + \langle \sigma - \sigma^* \mid \boldsymbol{L}_{\tilde{n}}(\sigma - \sigma^*) \rangle \geq \tilde{\mu} \| \mathrm{col}(x - x^*, \sigma - \sigma^*) \|^2 \geq \mu_{\tilde{\boldsymbol{F}}_{\mathrm{a}}} \| \mathrm{col}(x - x^*, s - s^*) \|^2$, where $\mu_{\tilde{\boldsymbol{F}}_{\mathrm{a}}} := (3 - \sqrt{5})\tilde{\mu}/2$ and the last inequality follows by definition of σ and bounds on quadratic forms. ■

Finally, the preconditioning matrix $\tilde{\Phi}$ is positive definite by Lemma 2.9. As in Lemma 2.7, by Lemma 2.11, it holds that $\tilde{\Phi}^{-1} \tilde{\mathcal{A}}|_\Sigma$ is restricted monotone in $\mathcal{H}_{\tilde{\Phi}}$. In view of (2.41), the conclusion follows analogously to Theorem 2.2. ■

3

PROXIMAL-POINT ALGORITHMS: MONOTONICITY AND SMOOTHNESS

I have had my results for a long time: but I do not yet know how I am to arrive at them.

Carl Friedrich Gauss

I used to attack because it was the only thing I knew. Now I attack because I know it works.

Garry Kasparov

We consider *Nash equilibrium problem*s (NEPs) in the partial-decision information scenario, where each agent can only exchange information with some neighbors, while its cost function possibly depends on the strategies of all agents. We characterize the relation between several monotonicity and smoothness assumptions postulated in the literature. Furthermore, we prove convergence of a preconditioned proximal-point algorithm, under a restricted monotonicity property that allows for a non-Lipschitz, non-continuous game mapping.

Parts of this chapter have been published in [26]

3.1. INTRODUCTION

NASH equilibrium seeking under partial-decision information has recently attracted considerable research interest, due to its prospect engineering applications as well as theoretical challenges. This scenario arises when, in the absence of a central coordinator, the agents in a network can only rely on the information received from some neighbors, for instance in ad-hoc-networks and sensor positioning problems [10], [51]. The technical goal is the distributed computation of an NE; the main complication is that the cost function of each agent may depend on the decision variables of other non-neighboring agents. To cope with the lack of knowledge, each agent estimates and tries to reconstruct the strategies of all the competitors [63], [136] (or an aggregation value [64], [82]) via peer-to-peer communication.

In fact, most existing methods resort to pseudogradient and consensus-type dynamics [47], [146]. Some works studied linearly convergent algorithms for games without coupling constraints [25], [136]. Other authors focused on generalized games, for example resorting to an operator-theoretic approach and forward-backward dual methods [64], [109]. All these schemes mainly suffer the following three drawbacks.

The first is that gradient-based methods typically require restrictive *monotonicity* assumptions for convergence. For instance, all the cited works postulate strong monotonicity of the game mapping. Weaker conditions are sometimes sufficient if allowing for vanishing stepsizes: strict monotonicity in the work [82], cocoercivity for generalized games in [10]. Remarkably, mere monotonicity was recently assumed in [87], via an additional diminishing Tikhonov regularization. Nonetheless, vanishing stepsizes are undesirable as they negatively affect the convergence speed. Most recently, the authors of [66] proposed a continuos-time gradient-based method for (hypo)-monotone games under a novel inverse Lipschitz assumption. The second drawback is that the agents' cost functions must be *differentiable* with *Lipschitz* gradient [109], [146]; in turn this ensures that the pseudogradient mapping of the game is Lipschitz. As the game mapping is a global operator, implementing, in a distributed setup, the common alternatives employed in nonsmooth optimization (linesearch or adaptive steps) seems far from trivial. The third drawback is that, due to partial-decision information, the stepsizes must be chosen very small, in turn increasing the number of iterations for convergence. Importantly, this also translates in prohibitive communication cost, as the agents need to exchange data at each step.

A possible solution to remedy all three limitations is the proximal-point method [8, Th. 23.41]. Although a direct implementation in games results in double layer schemes (where the agents have to communicate virtually infinite time between iterations [124], [148]), in our recent work [21], [22] we have shown that an efficient method can be obtained via *preconditioning* – for the case of games with strongly monotone and Lipschitz mapping. The result is that, at the price of some additional *local* complexity, the number of iterations and communications for convergence to a NE can be substantially reduced.

In this chapter we further leverage the properties of PPAs to deal with the other two issues: monotonicity and smoothness. Our contributions are summarized as follows:

- We compare a significant group of monotonicity and smoothness assumptions employed in the partial-decision information literature; we characterize the relations between the conditions, and exemplify their restrictiveness (§3.3);

- We prove convergence of our fully-distributed NE seeking PPP algorithm, under the restricted monotonicity of an augmented operator. Our condition is remarkably weaker than that recently proposed in [75, Th. 2] (for a Douglas–Rachford algorithm). In particular, we do not assume strong monotonicity, nor continuity of the game mapping –which requires a different limiting argument compared to [22, Th. 2]. Interestingly, nonsmoothness only affects the *local* optimization problems of the agents (§3.4).

To improve readability, the proofs are in the chapter appendix. We refer to Appendices A, B, C for the basic notation and mathematical background.

3.1.1. PRELIMINARIES: RESTRICTED MONOTONICITY

Definition 3.1 (*Restricted monotonicity*)**.** An operator $\mathcal{F}:\mathbb{R}^q \rightrightarrows \mathbb{R}^q$ is restricted (strictly, μ-strongly) monotone in $\mathcal{H}_P$ with respect to a set $\Sigma \neq \varnothing$ if $\langle x-x^\star, u-u^\star\rangle_P \geq 0$ (>0, $\geq \mu\|x-x^\star\|_P^2$) for all $(x,u)\in \operatorname{gra}(\mathcal{F})$, $(x^\star,u^\star)\in\operatorname{gra}(\mathcal{F})$ with $x^\star\in\Sigma$. We omit the characterization "in $\mathcal{H}_P$" whenever $P=I$. □

This definition slightly generalizes that in [22, Def. 1], which only considers the zero set; note that $\mathcal{F}$ is allowed to be set-valued on $x^\star\in\Sigma$.

3.2. MATHEMATICAL SETUP

3.2.1. THE GAME

Let $\mathcal{I}:=\{1,\dots,N\}$ be a set of agents, where each agent $i\in\mathcal{I}$ chooses its strategy (i.e., decision variable) x_i from its local decision set $\Omega_i\subseteq\mathbb{R}^{n_i}$. We denote by $x:=\operatorname{col}((x_i)_{i\in\mathcal{I}})\in\Omega$ the stacked vector of all the agents' strategies, with $\Omega:=\Omega_1\times\cdots\times\Omega\subseteq\mathbb{R}^n$ the overall decision space and $n:=\sum_{i\in\mathcal{I}} n_i$. Agent $i\in\mathcal{I}$ aims to minimize an objective function $J_i(x_i,x_{-i})$, depending both on the local variable x_i and on the strategies of the other agents $x_{-i}:=\operatorname{col}((x_j)_{j\in\mathcal{I}\setminus\{i\}})$. The game consists of N inter-dependent optimization problems

$$\forall i\in\mathcal{I}:\quad \operatorname*{argmin}_{y_i\in\Omega_i} J_i(y_i,x_{-i}). \tag{3.1}$$

The mathematical problem we consider is the distributed computation of a NE, a set of strategies simultaneously solving all the problems in (3.1).

Definition 3.2. A Nash equilibrium is a set of strategies $x^\star=\operatorname{col}\left((x_i^\star)_{i\in\mathcal{I}}\right)$ such that, for all $i\in\mathcal{I}$, $x_i^\star\in\operatorname{argmin}_{y_i\in\Omega_i} f_i(y_i,x_{-i}^\star)$. □

We restrict our attention to convex games. The following are standard regularity conditions [82, Asm. 1], [109, Asm. 1].

Assumption 3.1 (*Convexity*)**.** For each $i\in\mathcal{I}$, the set Ω_i is nonempty, closed and convex; the fuction J_i is continuous and the function $J_i(\cdot,x_{-i})$ is convex for any x_{-i}. □

Furthermore, we assume existence of a solution.

Assumption 3.2 (*Existence*)**.** The game in (3.1) admits at least one Nash equilibrium. □

Sufficient conditions for existence of a NE (e.g., compactness of Ω) can be found for instance in [57].

3.2.2. THE COMMUNICATION NETWORK

The agents can exchange information with some neighbors over an undirected communication network $\mathcal{G}(\mathcal{I},\mathcal{E})$. The pairs $(i,j),(j,i)$ belong to the set of edges $\mathcal{E}$ if and only if agent i and j can mutually exchange information. We denote: $W \in \mathbb{R}^{N\times N}$ the weight matrix of $\mathcal{G}$, with $w_{i,j} := [W]_{i,j}$ and $w_{i,j} > 0$ if $(i,j) \in \mathcal{E}$, $w_{i,j} = 0$ otherwise; $\mathcal{N}_i = \{j \mid (i,j) \in \mathcal{E}\}$ the set of neighbors of agent i.

Assumption 3.3 (*Connectivity*)**.** The communication graph $\mathcal{G}(\mathcal{I},\mathcal{E})$ is undirected and connected. The weight matrix W satisfies the following conditions:

(i) *Symmetry:* $W = W^\top$;

(ii) *Self loops:* $w_{i,i} > 0$ for all $i \in \mathcal{I}$;

(iii) *Double stochasticity:* $W\mathbf{1}_N = \mathbf{1}_N, \mathbf{1}^\top W = \mathbf{1}^\top$. □

The requirements (ii)-(iii) in Assumption 3.3 are intended to ease the notation; for instance, they can be satisfied by assigning Metropolis weights [25, §2].

3.2.3. THE PARTIAL-DECISION INFORMATION SCENARIO

We consider the so-called partial-decision information model, where agent $i \in \mathcal{I}$ can only access its own feasible set Ω_i and an analytic expression of its private cost J_i, but cannot access the strategies of all the competitors x_{-i}. Therefore, each agent i is unable to evaluate the actual value of $J_i(x_i, x_{-i})$. Instead, each agent keeps an estimate of all other agents' actions [63], [82], [47], and aims at reconstructing the actual values, only based information exchanged locally with neighbors over the communication graph $\mathcal{G}$. We denote $\boldsymbol{x}_i = \mathrm{col}((\boldsymbol{x}_{i,j})_{j\in\mathcal{I}}) \in \mathbb{R}^n$, where $\boldsymbol{x}_{i,i} := x_i$ and $\boldsymbol{x}_{i,j}$ is agent i's estimate of agent j's strategy, for all $j \neq i$; $\boldsymbol{x}_{j,-i} = \mathrm{col}((\boldsymbol{x}_{j,l})_{l\in\mathcal{I}\setminus\{i\}})$; $\boldsymbol{x} = \mathrm{col}((\boldsymbol{x}_i)_{i\in\mathcal{I}}) \in \mathbb{R}^{Nn}$ the overall estimate vector; $\boldsymbol{x}_{-i} = \mathrm{col}((\boldsymbol{x}_j)_{j\in\mathcal{I}\setminus\{i\}})$. Let

$$\mathcal{R}_i := \begin{bmatrix} \mathbf{0}_{n_i\times n_{<i}} & I_{n_i} & \mathbf{0}_{n_i\times n_{>i}} \end{bmatrix}, \tag{3.2}$$

where $n_{<i} := \sum_{j<i, j\in\mathcal{I}} n_j$, $n_{>i} := \sum_{j>i, j\in\mathcal{I}} n_j$. In simple terms, $\mathcal{R}_i$ selects the i-th n_i-dimensional component from an n-dimensional vector, i.e., $\mathcal{R}_i\boldsymbol{x}_i = \boldsymbol{x}_{i,i} = x_i$. Let also $\mathcal{R} := \mathrm{diag}\big((\mathcal{R}_i)_{i\in\mathcal{I}}\big)$, so that $x = \mathcal{R}\boldsymbol{x}$.

3.2.4. GAME MAPPING, EXTENDED MAPPING, AUGMENTED OPERATORS

Under Assumption 3.1, a strategy $x^\star$ is a NE of the game in (3.1) if and only if

$$\mathbf{0}_n \in F\left(x^\star\right) + \mathrm{N}_\Omega\left(x^\star\right), \tag{3.3}$$

where $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is the *game mapping*

$$F(x) := \mathrm{col}\left((\partial_{x_i} J_i(x_i, x_{-i}))_{i\in\mathcal{I}}\right), \tag{3.4}$$

(in fact, (3.3) are the first order optimality conditions of each convex problems in (3.1)). Typically, distributed NE seeking methods require some monotonicity assumption on F.

Since we deal with the partial-decision information scenario, it is also useful to introduce the *extended game mapping*

$$\boldsymbol{F}(\boldsymbol{x}) := \operatorname{col}\left((\partial_{x_i} J_i(x_i, \boldsymbol{x}_{i,-i}))_{i\in\mathcal{I}}\right) \tag{3.5}$$

where the subdifferentials are computed on the *estimates*, and the *augmented* operators

$$\mathcal{F}_\alpha(\boldsymbol{x}) := \alpha\mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}) + (I_{Nn} - \boldsymbol{W})\boldsymbol{x} \tag{3.6}$$

$$\mathcal{A}_\alpha(\boldsymbol{x}) := \mathcal{F}_\alpha(\boldsymbol{x}) + \mathrm{N}_{\boldsymbol{\Omega}}(\boldsymbol{x}), \tag{3.7}$$

where $\alpha > 0$ is a design parameter, $\boldsymbol{W} := W \otimes I_n$, $\boldsymbol{\Omega} := \{\boldsymbol{x} \in \mathbb{R}^{Nn} \mid \mathcal{R}\boldsymbol{x} \in \Omega\}$. The following well-known result (e.g., [136, Prop. 1]) provides an extension of the inclusion (3.3) to the estimate space.

Lemma 3.1. The following statements are equivalent:

i) $\boldsymbol{x}^\star = \mathbf{1}_N \otimes x^\star$, with $x^\star \in \Omega$ a NE of the game (3.1);

ii) $\mathbf{0}_{Nn} \in \mathcal{A}_\alpha(\boldsymbol{x}^\star)$. □

In particular, note that Assumption 3.2 implies that $\operatorname{zer}(\mathcal{A}_\alpha) \neq \varnothing$.

3.3. TOWARDS A TAXONOMY OF ASSUMPTIONS

In recent years, distributed NE seeking under partial-decision information has been studied under a variety of conditions on the operators $F, \mathcal{R}^\top\boldsymbol{F}, \mathcal{F}_\alpha, \mathcal{A}_\alpha$. Some of the assumptions postulated have not been exemplified, nor it is evident how restrictive they are –in theory and in practice. Towards a solution of this issue, we start by considering the following, representative, conditions.

C1. The operator $\mathcal{R}^\top\boldsymbol{F}$ is maximally monotone.

C2. The operator $\mathcal{R}^\top\boldsymbol{F}$ is restricted monotone with respect to $\operatorname{zer}(\mathcal{A}_\alpha)$.

C3. There exists $\alpha > 0$ such that the operator $\mathcal{F}_\alpha$ is maximally monotone.

C4. There exists $\alpha > 0$ such that the operator $\mathcal{F}_\alpha$ is restricted monotone with respect to $\operatorname{zer}(\mathcal{A}_\alpha)$.

C5. The operator F is μ-restricted strongly monotone with respect to the set of NEs and θ-Lipschitz, for some $\mu > 0$, $\theta > 0$.

C6. The operator F is μ-strongly monotone and θ-Lipschitz, for some $\mu > 0$, $\theta > 0$.

C7. The operator F is ν-hypomonotone, θ-Lipschitz, and R-inverse Lipschitz, for some $\nu \geq 0$, $\theta > 0$, $R > 0$, $R\nu < 1$.

C8. The operator F is strictly monotone and θ-Lipschitz, for some $\theta > 0$.

C9. The operator F is $\frac{1}{\theta}$ cocoercive for some $\theta > 0$.

C10. The operator F is monotone and θ-Lipschitz, for some $\theta > 0$.

	ref.	extra asm.	step sizes
C1	[63], [66]		Continuous time
C3	[74], [75]		Fixed
C5	[136]		Fixed
C6	[22], [47], [109]		Fixed
C7	[66]	$\Omega = \mathbb{R}^n$	Continuous time
C8	[82]	Ω compact	Vanishing
C9	[10]	Ω compact	Vanishing
C10	[87]	Ω compact	Vanishing

Table 3.1: Technical assumptions in the literature.

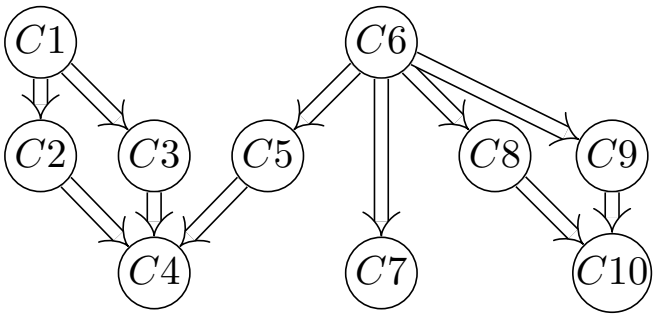

Figure 3.1: Relations between technical assumptions in monotone games under partial-decision information.

Although C6 is the most common technical assumption, all these conditions have been formulated in the literature (see Table 3.1), except for C2 (which is a natural relaxations of C1) and C4 (which we will use to show convergence of our algorithm). The following result characterizes the relation between them.

Proposition 3.1. The implications in Figure 3.1 hold true. □

It can be also shown by counter examples that no other implication exists between the conditions in C1-C10.

3.3.1. CONDITIONS ON THE EXTENDED PSEUDOGRADIENT

We next prove, under the commonly used assumption that F is single valued, that C1 is very restrictive.

Proposition 3.2 (*C1 is trivial*)**.** Assume that F is single valued and continuous. Then, condition C1 holds if and only if $\nabla J_i(\cdot, x_{-i})$ is independent of x_{-i}, for all $i \in \mathcal{I}$. □

As the actions x_{-i} are not affecting the optimization problem of agent i (beside possibly for a separable component), there appears to be no reason for agent i to keep estimates (hence, for a partial-decision information setup).

Example 3.1. The game defined by $N = 2$, $n = 2$, $\Omega = \mathbb{R}^n$, $J_1(x) = (x_1 - 1)^2(x_2{}^2 + 1)$, $J_2(x) = x_2{}^2(x_1{}^2 + 1)$ has a unique NE in $[1\ 0]^\top$ and satisfies C2, but not C1. □

Although $\nabla_{x_i} J_i$ depends on x_{-i} in Example 3.1, the next lemma shows that C2 is also not of particular interest.

Proposition 3.3 (*C2 is trivial*)**.** Assume that F is single valued and continuous. Then, condition C2 holds if and only if $\nabla_{x_i} J_i(x_i^\star, x_{-i})$ is independent of x_{-i}, for all $i \in \mathcal{I}$, for any $x^\star = (x_i^\star, x_{-i}^\star)$ NE of the game (3.1). □

In particular, Proposition 3.3 implies that $0 \leq \nabla_{x_i} J_i(x_i^\star, x_{-i}^\star), x_i - x_i^\star\rangle = \langle \nabla_{x_i} J_i(x_i^\star, x_{-i}), x_i - x_i^\star\rangle$ where the inequality is the first order optimality condition (as $x_i^\star$ solves (3.1)). This means that $x_i^\star$ is optimal for agent i regardless of x_{-i}; in other terms, C2 implies that the Nash equilibria are uniquely composed by dominant strategies (as in Example 3.1). This is also a trivial case, as the agents do not need to communicate to compute a NE. Although the condition in Proposition 3.3 might be violated if F is not continuous, this can only happen at discontinuity points, which is quite a pathological case.

3.3.2. CONDITIONS ON THE GAME PRIMITIVES

Conditions C5 to C10 are directly postulated on the game mapping F and are the most well-investigated (e.g., they are easy to check if F is a linear operator [8], [56], [66]). Conditions C5 to C8 imply uniqueness of the equilibrium; methods with linear convergence were proposed under C5, C6 [25], [136], but not C7, C8. Although C5 is weaker than C6 in theory, it is difficult to check without knowledge of the solutions; we have included it because it causes very limited complications in the convergence analysis with respect to C6. C6 actually implies that there is $\alpha > 0$ such that $\mathcal{F}_\alpha$ is Lipschitz and restricted strongly-monotone with respect to the whole consensus subspace $\boldsymbol{E} := \{\boldsymbol{y} \in \mathbb{R}^{Nn} \mid \boldsymbol{y} = \mathbf{1}_N \otimes y,\ y \in \mathbb{R}^n\} \supseteq \operatorname{zer}(\mathcal{A}_\alpha)$ [109, Lem. 3], a much more restrictive condition that C4. C10 and C9 allow for multiple NEs; yet – as for C8 – the related methods require not only compact feasible sets (possibly reasonable in practice) but also vanishing steps, which affect the convergence speed.

3.3.3. CONDITIONS ON THE AUGMENTED OPERATOR

C3 and C4 are more abstract and often replaced by more easily checked sufficient conditions. For example, restricted monotonicity of $\mathcal{F}_\alpha$ with respect to the consensus space $\boldsymbol{E}$ can be always checked without knowledge of the solutions, and implies C4.

Despite this complication, C3 and C4 are of great interest, especially for nonsmooth games, as exemplified next. The following examples also show that C3 is significantly more restrictive than C4.

Example 3.2. Consider the game defined by $N = 2$, $n = 2$, $\Omega = \mathbb{R}^n$, $F(x) = \bar{F}(x) + \hat{F}(x)$, with $\bar{F}(x) = \operatorname{col}(x_1{}^3, 0)$ and $\hat{F}(x) = \left[\begin{smallmatrix} 2 & 1 \\ 1 & 2 \end{smallmatrix}\right] x + \left[\begin{smallmatrix} 5 \\ 4 \end{smallmatrix}\right]$. As $\bar{F}$ is monotone and $\hat{F}$ is strongly monotone, the game admits a unique NE. Conditions C5–C10 are violated, as they require Lipschitz continuity of F; C2 also fails (as the best response of agent 2 is $-0.5x_1 - 2$ and by Proposition 3.3). However, C4 holds: to show this, consider the components of the extended game mapping $\bar{\boldsymbol{F}}$ and $\hat{\boldsymbol{F}}$ corresponding to $\bar{F}$ and $\hat{F}$; $\mathcal{R}^\top \bar{\boldsymbol{F}}$ is monotone, while

Algorithm 3.1. Fully-distributed PPP algorithm

$$\begin{aligned}\tilde{\boldsymbol{x}}_i^k &= \tfrac{1}{2}(\boldsymbol{x}_i^k + \textstyle\sum_{j=1}^N w_{i,j}\boldsymbol{x}_j^k)\\ \boldsymbol{x}_{i,-i}^{k+1} &= \tilde{\boldsymbol{x}}_{i,-i}^k\\ x_i^{k+1} &= \operatorname*{argmin}_{y\in\Omega_i}\left(J_i(y,\tilde{\boldsymbol{x}}_{i,-i}^k) + \tfrac{1}{\alpha}\|y-\tilde{\boldsymbol{x}}_{i,i}^k\|^2\right)\end{aligned}$$

$\alpha\mathcal{R}^\top\hat{\boldsymbol{F}}+(I-\boldsymbol{W})$ can be made restricted monotone with respect to the whole consensus subspace by choosing $\alpha>0$ small enough [109, Lem. 3]. We can check numerically that C3 also holds for some W (in particular, because $\alpha\mathcal{R}^\top\hat{\boldsymbol{F}}+(I-\boldsymbol{W})$ can be made monotone, although there is no analytical result available to check this a priori). □

Example 3.3 (*Non-monotone game*)**.** Consider Example 3.2 but with $\tilde{F}(x)=\mathrm{col}({x_1}^3({x_2}^4+1),0)$ and $\hat{F}(x)=\left[\begin{smallmatrix}2&1\\1&2\end{smallmatrix}\right]x$. The game admits a NE $x^\star=\mathbf{0}$. As F is restricted strongly monotone with respect to $x^\star$, the equilibrium must be unique. As for Example 3.2, it is easy to prove that C4 holds, because $\mathcal{R}^\top\tilde{\boldsymbol{F}}$ is restricted monotone with respect to $\mathbf{0}$ (by Proposition 3.3). Yet, F is not monotone: therefore C3 cannot hold (nor can C1, C2, C5-C10). □

Example 3.4 (*Set-valued pseudo-gradient*)**.** Consider the game defined by $N=2$, $n=2$, $\Omega=\mathbb{R}^n$, $J_1(x)={x_1}^2+|x_1||x_2|$, $J_2(x)={x_2}^2+x_2x_1$, where $|\cdot|$ is the absolute value. The game admits a unique NE in $\mathbf{0}$; moreover, F is set valued, as f_1 is not differentiable in the local variable. It can be checked that C4 holds. Yet, F is not monotone, thus C3 fails. □

3.4. THE PPP ALGORITHM

IN this section we consider the fully-distributed proximal-point NE seeking method shown in Algorithm 3.1. The iteration coincides with that studied in [21], although the terms have been rearranged. The algorithm includes a consensus phase, where the agents exchange and mix their variable vectors. The local actions are then updated according to a proximal-best response with stepsize $\alpha>0$ – importantly, the cost function of each agent i evaluated in the estimates $\boldsymbol{x}_{i,-i}$, and not on the real competitor's actions x_{-i}. Note that the algorithm is always well (uniquely) defined, as the update of x_i is the argmin of a strongly convex function (by convexity of $J_i(\cdot,x_{-i})$ in Assumption 3.1).

Algorithm 3.1 can be formulated as a proximal-point method applied to the operator $\mathcal{A}_\alpha$. However, the computation of $(\mathrm{Id}+\mathcal{A}_\alpha)^{-1}$ cannot be performed in a distributed way (more precisely, it would require the collaborative solution of a regularized game at each iteration, resulting in a scheme with nested layers of communication, see [124]). We have shown in [21], [22] that this complication can be tackled by preconditioning the operator $\mathcal{A}_\alpha$ with a positive definite matrix

$$\Phi := I_{Nn}+\boldsymbol{W}. \tag{3.8}$$

Lemma 3.2 ([21, Lem. 2])**.** Algorithm 3.1 can be written as

$$\boldsymbol{x}^{k+1} = (\mathrm{Id} + \Phi^{-1}\mathcal{A}_\alpha)^{-1}(\boldsymbol{x}^k). \tag{3.9}$$

□

This operator-theoretic interpretation is very powerful, as it seamlessly allows to study convergence of analogous proximal-best response schemes even in the presence of inexact updates (i.e., the argmin is only approximated at each iteration), coupling constraint, acceleration terms [22]. It also immediately shows that the fixed points of Algorithm 3.1 coincide with $\operatorname{zer}(\mathcal{A}_\alpha) = \operatorname{zer}(\Phi^{-1}\mathcal{A}_\alpha)$ (i.e., they are estimates at consensus at a Nash equilibrium).

The following theorem is the main result of this chapter. It extends the convergence results in [22, Th. 3], formulated under C6, to the case of restricted monotone – possibly nonsmoooth – games (i.e., C4).

Theorem 3.1. Let Assumption 3.1–3.3 hold, and assume that C4 holds for some $\alpha > 0$. Then, the sequence $(\boldsymbol{x}^k)$ generated by Algorithm 3.1 converges to a point $\boldsymbol{x}^\star = \mathbf{1}_N \otimes x^\star$, where $x^\star$ is a Nash equilibrium of the game in (3.1). □

Remark 3.1. In [21] we have proven (linear) convergence of Algorithm 3.1 assuming C6. Under the weaker C4, Theorem 3.1 leverages the general results for the proximal-point algorithm of restricted (merely) monotone games [22]. With respect to [22] and to the Douglas-Rachford algorithm in [75], we use a different limiting argument in our proof, which does not require F to be Lipschitz continuous (or even continuous). The core idea is to show that the operator $\mathrm{J}_{\Phi^{-1}\mathcal{A}_\alpha}$ is continuous, even if $\mathcal{A}_\alpha$ might not (nor is maximally monotone). For instance, Theorem 3.1 can be applied to the games in Examples 3.2 to 3.4, while [22, Th. 2], [66, Th. 2] cannot. Examples 3.3 and 3.4 also show a significant gap between C4 and the stronger condition C3, employed in [75, Th. 3]. □

We conclude this section by sketching some technical extensions of our results. To start, our arguments in Theorem 3.1 can be readily adapted to the algorithms – for generalized games – studied in [22], to show convergence under C4. Moreover, our convergence result would hold assuming the definition of restricted monotonicity proposed in [22, Def. 1], slightly less restrictive than our Definition 3.1. We also note that we assumed monotonicity properties of F (and similarly for the other game operators) to hold over all $\mathbb{R}^n$; however, the conditions can be relaxed to hold only over the feasible set, if the estimates $\boldsymbol{x}$'s are initialized in Ω^N (since the update in Algorithm 3.1 guarantees invariance for this set). The costs in (3.1) can be modified to include a more general (discontinuous) proper, convex, closed function $g_i(x_i)$ (besides the indicator function ι_{Ω_i}), without technical complications. Much more intriguing is the case of discontinuity in the part of the cost coupled with the other agents (i.e., violating Assumption 3.1): although our convergence arguments do not hold in this case, it would be interesting to verify whether C3 could be satisfied to apply standard PPA results.

3.5. CONCLUSION AND OUTLOOK

BESIDES their efficiency, proximal-point algorithms have the advantage of only requiring mild monotonicity and smoothness conditions. We have compared and ana-

lyzed several assumptions in NE seeking under partial-decision information, and proved the convergence of a fully-distributed PPP method under one of the weakest.

Future work should investigate linear rates in absence of (restricted) strong monotonicity. One promising option is to leverage inverse Lipschitz properties, which can ensure contractivity of certain resolvents. Proving convergence in merely monotone regime, under fixed step sizes, is also a challenging open problem.

3.6. APPENDIX

3.6.1. PROOF OF PROPOSITION 3.1

$C1 \Rightarrow C2$, $C3 \Rightarrow C4$, $C6 \Rightarrow C5$, $C6 \Rightarrow C8$, $C8 \Rightarrow C10$: By definition.

$C1 \Rightarrow C3$: As $(I - W)$ is a positive semidefinite matrix, the operator $I - \boldsymbol{W}$ is maximally monotone. Hence, for any $\alpha \geq 0$, $\mathcal{F}_\alpha = \alpha \mathcal{R}^\top \boldsymbol{F} + (I - \boldsymbol{W})$ is the sum of two maximally monotone operators; moreover, $\mathrm{dom}(I - \boldsymbol{W}) = \mathbb{R}^{Nn}$, so the conclusion follows by [8, Cor. 25.5].

$C2 \Rightarrow C4$: $\mathcal{F}_\alpha$ is the sum of a restricted monotone operator and a monotone operator, hence restricted monotone.

$C5 \Rightarrow C6$: See, for instance, [22, Lem. 3].

$C6 \Rightarrow C7$: It follows by definition and [66, Prop. 3].

$C6 \Rightarrow C9$: See, e.g., [66, Prop. 5].

$C9 \Rightarrow C10$: It follows by definition of cocoercivity and the Cauchy–Schwartz inequality. ■

3.6.2. PROOF OF PROPOSITION 3.2

"⇐": For the sake of contradiction, assume that, for some $i \in \mathcal{I}$, there exist $l \in \{1, 2, \ldots, n_i\}$, $x_i \in \mathbb{R}^{n_i}$ and a pair of vectors x_{-i} and x'_{-i} such that $[\nabla_{x_i} J_i(x_i, x_{-i})]_l < [\nabla_{x_i} J_i(x_i, x'_{-i})]_l$. By continuity, there exists $\epsilon > 0$ such that $[\nabla_{x_i} J_i(x_i + \epsilon \mathbf{e}_l, x_{-i})]_l < [\nabla_{x_i} J_i(x_i, x'_{-i})]_l$, where $\mathbf{e}_l \in \mathbb{R}^n_i$ is the l-th vector of the canonical basis. The monotonicity in C1, applied to a pair of estimate vectors $(\boldsymbol{x}_i, \boldsymbol{x}_{-i})$, $(\boldsymbol{x}'_i, \boldsymbol{x}_{-i})$, for any $\boldsymbol{x}_{-i}$ and $\boldsymbol{x}_i = (x_i + \epsilon \mathbf{e}_l, x_{-i})$, $\boldsymbol{x}'_i = (x_i, x'_{-i})$, gives

$$\begin{aligned} 0 &\leq \langle \nabla_{x_i} J_i(x_i + \epsilon \mathbf{e}_l, x_{-i}) - \nabla_{x_i} J_i(x_i, x'_{-i}), \epsilon \mathbf{e}_l \rangle \\ &= \epsilon [\nabla_{x_i} J_i(x_i + \epsilon \mathbf{e}_l, x_{-i}) - \nabla_{x_i} J_i(x_i, x'_{-i})]_l \qquad < 0 \end{aligned}$$

which is a contradiction. Because x_{-i}, x'_{-i} are arbitrary, we conclude that, for all $i \in \mathcal{I}$, for all x_i, and for all x_{-i}, x'_{-i}, $\nabla_{x_i} J_i(x_i, x_{-i}) = \nabla_{x_i} J_i(x_i, x'_{-i})$.

"⇒": By assumption, for any $i \in \mathcal{I}$, x_i, x'_i, x_{-i}, x'_{-i},

$$\begin{aligned} &\langle \nabla_{x_i} J_i(x_i, x_{-i}) - \nabla_{x_i} J_i(x'_i, x'_{-i}), x_i - x'_i \rangle \\ &= \langle \nabla_{x_i} J_i(x_i, x'_{-i}) - \nabla_{x_i} J_i(x'_i, x'_{-i}), x_i - x_i \rangle \geq 0, \end{aligned}$$

where the inequality is convexity of J_i in the first argument (Assumption 3.1). Stacking the inequalities for $i \in \mathcal{I}$ retrieves the monotonicity of $\mathcal{R}^\top \boldsymbol{F}$.

■

3.6.3. PROOF OF PROPOSITION 3.3

"⇐": For contradiction, assume that there exist $i \in \mathcal{I}$, $l \in \{1,2,\dots,n_i\}$, an NE $x^\star$ and x_{-i} such that $[\nabla_{x_i} J_i(x_i^\star, x_{-i})]_l < [\nabla_{x_i} J_i(x_i^\star, x_{-i}^\star)]_l$. By continuity, there exists $\epsilon > 0$ such that $[\nabla_{x_i} J_i(x_i^\star + \epsilon \mathbf{e}_l, x_{-i})]_l < [\nabla_{x_i} J_i(x_i^\star, x_{-i}^\star)]_l$. Restricted monotonicity in C2, applied to a pair of estimate vectors $(\boldsymbol{x}_i, \boldsymbol{x}_{-i})$, $(x^\star, \boldsymbol{x}_{-i})$, for any $\boldsymbol{x}_{-i}$ and $\boldsymbol{x}_i = (x_i^\star + \epsilon e_l, x_{-i})$, gives

$$\begin{aligned} 0 &\le \langle \nabla_{x_i} J_i(x_i^\star + \epsilon \mathbf{e}_l, x_{-i}) - \nabla_{x_i} J_i(x_i^\star, x_{-i}^\star), \epsilon \mathbf{e}_l \rangle \\ &= \epsilon [\nabla_{x_i} J_i(x_i^\star + \epsilon \mathbf{e}_l, x_{-i}) - \nabla_{x_i} J_i(x_i^\star, x_{-i}^\star)]_l \qquad < 0 \end{aligned}$$

which is a contradiction. Analogously it can be shown that $[\nabla_{x_i} J_i(x_i^\star, x_{-i})]_l > [\nabla_{x_i} J_i(x_i^\star, x_{-i}^\star)]_l$ leads to a contradiction. Hence $\nabla_{x_i} J_i(x_i^\star, x_{-i}) = \nabla_{x_i} J_i(x_i^\star, x_{-i}^\star)$.

"⇒": For any $i \in \mathcal{I}$, x_i, x_{-i}, NE $x^\star$, by assumption and convexity, $\langle J_i(x_i, x_{-i}) - \nabla_{x_i} J_i(x_i^\star, x_{-i}^\star), x_i - x_i^\star \rangle = \langle \nabla_{x_i} J_i(x_i, x_{-i}) - \nabla_{x_i} J_i(x_i^\star, x_{-i}), x_i - x_i^\star \rangle \ge 0$. ■

3.6.4. PROOF OF THEOREM 3.1

We start by an auxiliary result.

Lemma 3.3. Let $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R} : (x, y) \mapsto f(x, y)$ be a continuous function, and assume that $f(\cdot, y)$ is μ-strongly convex for any $y \in \mathbb{R}^m$, $\mu > 0$. Let $X \subseteq \mathbb{R}^n$ be a convex closed set. Then the (single valued, full domain) mapping $y \mapsto g(y) = \operatorname{argmin}_{x \in X} f(x, y)$ is continuous. □

Proof. We show that, for any given sequence $(y_k)_{k \in \mathbb{N}}$ with $y^k \to y^\star$ (converging, hence bounded), $x^k := g(y^k) \to g(y^\star) =: x^\star$, which is the definition of continuity.

First, we show that $(x^k)_{k \in \mathbb{N}}$ is bounded. Let Y be a compact set containing $(y^k)_{k \in \mathbb{N}}$. Let $x_0 \in X$ and

$$l_0 := \max_{y \in Y} f(x_0, y), \qquad l_1 := \min_{x \in \partial B(x_0, 1), y \in Y} f(x, y)$$

where $\partial B(x_0, 1) = \{x \in \mathbb{R}^n \mid \|x - x_0\| = 1\}$ is the boundary of the unit ball centered at x_0; the min and max are achieved because the domains are compact. Let $d \in \mathbb{R}^n$ be any unitary vector, i.e., $\|d\| = 1$; $x_1 := x_0 + d \in \partial B(x_0, 1)$; $x_2 = x_0 + Md$, for some scalar such that

$$M > 1, \qquad M > 2\tfrac{l_0 - l_1}{\mu} + 1. \tag{3.10}$$

Then, $x_1 = \frac{M-1}{M} x_0 + \frac{1}{M} x_2$. By definition of strong convexity, we have, for all $y \in Y$

$$\begin{aligned} l_1 &\le f(x_1, y) \\ &\le \tfrac{M-1}{M} f(x_0, y) + \tfrac{1}{M} f(x_2, y) - \tfrac{1}{2} \mu \tfrac{M-1}{M} \tfrac{1}{M} \|x_0 - x_2\|^2 \\ &= \tfrac{M-1}{M} f(x_0, y) + \tfrac{1}{M} f(x_2, y) - \tfrac{1}{2} \mu (M-1). \end{aligned}$$

Assume for contradiction that there exists $y \in Y$ such that $f(x_2, y) \le f(x_0, y)$. Then, since $f(x_0, y) \le l_0$, the previous inequality implies $l_1 - l_0 \le -\frac{1}{2}\mu(M-1)$, which contradicts (3.10). Since d is arbitrary, we conclude that, for any $y \in Y$, for all x such that $\|x_0 - x\| > M$, $f(x_0, y) < f(x, y)$. In turn, for all $y \in Y$, $\|g(y)\| < \|x_0\| + M$, i.e., g is uniformly bounded over Y; thus $(x_k)_{k \in \mathbb{N}}$ is bounded.

Hence $(x_k)_{k\in\mathbb{N}}$ admits an accumulation point, say x'. Let $\bar{K} = (\bar{k}_1, \bar{k}_2, \ldots) \subseteq \mathbb{N}$ be a diverging subsequence such that $x^{\bar{k}_n} \to x'$. Since $f(x^{\bar{k}_n}, y^{\bar{k}_n}) \leq f(x, y^{\bar{k}_n})$ for all $x \in X$, by continuity of f, we have $f(x', y^\star) \leq f(x, y^\star)$ for all $x \in X$. Since the minimizer must be unique by strong convexity, we have $x' = x^\star$. In particular, this shows that $x^\star$ is the unique accumulation point of x^k: therefore, $x^k \to x^\star$. ■

The proof of Theorem 3.1 is based on the following result.

Lemma 3.4. The operator $\mathrm{J}_{\Phi^{-1}\mathcal{A}_\alpha}$ is continuous. □

3

Proof. For each $i \in \mathcal{I}$, the mapping $\tilde{\boldsymbol{x}}_i \mapsto \operatorname{argmin}_{y\in\Omega_i}(J_i(y, \tilde{\boldsymbol{x}}_{i,-i}) + \frac{1}{\alpha}\|y - \tilde{\boldsymbol{x}}_{i,i}\|^2)$ is continuous by Lemma 3.3. The result follows by Lemma 3.2 and the explicit form of $\mathrm{J}_{\Phi^{-1}\mathcal{A}_\alpha}$ in Algorithm 3.1. ■

We are now in a position to apply the results on proximal-point algorithm for restricted monotone operators in [22]. First, note that the operator $\mathcal{A}_\alpha$ is restricted monotone with respect to $\operatorname{zer}(\mathcal{A}_\alpha)$ (because $\mathcal{F}_\alpha$ is so by assumption, and by monotonicity of the normal cone [8, Th. 20.25]), i.e., for all $(\boldsymbol{x}, \boldsymbol{u}), (x^\star, \boldsymbol{u}^\star) \in \operatorname{gra}(\mathcal{A}_\alpha)$, with $x^\star \in \operatorname{zer}(\mathcal{A}_\alpha)$

$$0 \leq \langle \boldsymbol{u} - \boldsymbol{u}^\star, \boldsymbol{x} - \boldsymbol{x}^\star \rangle = \langle \Phi^{-1}\boldsymbol{u} - \Phi^{-1}\boldsymbol{u}^\star, \boldsymbol{x} - \boldsymbol{x}^\star \rangle_\Phi,$$

which shows that $\Phi^{-1}\mathcal{A}_\alpha$ is restricted monotone with respect to $\operatorname{zer}(\mathcal{A}_\alpha)$ in $\mathcal{H}_\Phi$. Therefore, by Lemma 3.2 and by applying [22, Th. 1(i)], we infer that the sequence $(\boldsymbol{x}^k)$ is bounded, hence it admits at least one cluster point, say $\bar{\boldsymbol{x}}$. By [22, Th. 1(ii)], $\mathrm{J}_{\Phi^{-1}\mathcal{A}_\alpha}(\boldsymbol{x}^k) - \boldsymbol{x}^k \to 0$; therefore, by continuity in Lemma 3.4, it must be $\bar{\boldsymbol{x}} \in \operatorname{fix}(\mathrm{J}_{\Phi^{-1}\mathcal{A}_\alpha}) = \operatorname{zer}(\mathcal{A}_\alpha)$. The conclusion follows by [22, Th. 1(iii)]. ■

II

FEEDBACK EQUILIBRIUM SEEKING FOR DYNAMICAL AGENTS

4

ADAPTIVE GNE SEEKING FOR MULTI-INTEGRATOR AGENTS

Feedback linearizes!
[answering the question "If you had to summarize the essence of feedback in one claim, what would it be?"]

Stephen Boyd

When a measure becomes a target, it ceases to be a good measure.

Goodhart's law

We consider strongly monotone games with convex separable coupling constraints, played by dynamical agents, in a partial-decision information scenario. We start by designing continuous-time fully-distributed feedback controllers, based on consensus and primal-dual gradient dynamics, to seek a generalized Nash equilibrium in networks of single-integrator agents. Our first solution adopts a fixed gain, whose choice requires the knowledge of some global parameters of the game. To relax this requirement, we conceive a controller that can be tuned in a completely decentralized fashion, thanks to the use of uncoordinated integral adaptive weights. We further introduce algorithms specifically devised for generalized aggregative games. Finally, we adapt all our control schemes to deal with heterogeneous multi-integrator agents and, in turn, with nonlinear feedback-linearizable dynamical systems. For all the proposed dynamics, we show convergence to a variational equilibrium, by leveraging monotonicity properties and stability theory for projected dynamical systems.

Parts of this chapter have been published in [24].

4.1. INTRODUCTION

GENERALIZED games arise in several engineering applications, including demand-side management in the smart grid [119], charging scheduling of electric vehicles [71] and communication networks [57]. These scenarios involve multiple autonomous decision makers, or agents; each agent aims at minimizing its individual cost function – which depends on its own action as well as on the actions of other agents – subject to shared constraints. Specifically, in many distributed control problems, the action of an agent consists of the output of a dynamical system. For instance, in coverage maximization [51] and connectivity problems [130], the agents are vehicles with some inherent dynamics, designed to optimize inter-dependent objectives related to their positions; in electricity markets, the actions are represented by the power produced by some generators [48]; in optical networks, the costs are a function of the output powers of some dynamical channels [116].

In this context, the goal is to drive the physical processes to a desirable steady state, usually identified with a GNE, using only the local information available to each agent. One possibility is to exploit time-scale separation between the computation of a GNE and setpoint tracking; yet, this solution is typically economically inefficient and not robust [153]. Alternatively, part of the recent literature focuses on the design of distributed feedback controllers, to automatically steer a dynamical network to some (not known a priori) convenient operating point, while also ensuring closed-loop stability [43], [48]. This chapter fits in the latter framework.

In particular, we investigate GNE seeking for multi-integrator agents, motivated by robotics and mobile sensors applications [60], [130], where multi-integrator dynamics are commonly used to model elementary vehicles. The study of this class of systems allows us to address GNE problems for a variety of dynamical agents, linear or nonlinear, via feedback linearization (e.g., Euler–Lagrangian systems as in [49]).

Literature review: A variety of algorithms has been proposed to seek a GNE in a distributed way [31], [149], [151] with a focus on aggregative games [9], [46], [105]. These works refer to (aggregative) games played in a full-decision information setting, where each agent can access the action of all the competitors (aggregate value), for example in presence of a central coordinator that broadcasts the data to the network. Nevertheless, this is impractical in many applications, where the agents only rely on local information.

Instead, in this thesis we consider the so-called *partial-decision information* scenario, where each agent holds an analytic expression for its cost but is unable to evaluate the actual value, since it cannot access the strategies of all the competitors. To remedy the lack of knowledge, the agents agree on sharing some information with some trusted neighbors over a communication graph. Based on the data exchanged, each agent can estimate and asymptotically reconstruct the actions of all the other agents. This setup has been investigated for games without coupling constraints, resorting to gradient and consensus dynamics, both in discrete-time [82], [137], and continuous-time [63], [146]. Fewer works deal with generalized games [50], [108], [109]. Moreover, all the results mentioned above consider static or single-integrator agents only. Distributively driving a network of more complex physical systems to a NE is still a relatively unexplored problem. With regard to aggregative games, a proportional integral feedback algorithm was developed in [48] to seek a NE in networks of passive second-order systems; in [49]

and [154], continuous-time gradient-based controllers were introduced for some classes of nonlinear dynamic. [130] addressed generally coupled cost games played by linear agents, via an extremum seeking approach; NE problems in systems of multi-integrator agents were studied by [116]. Yet, none of these works considers generalized games. Despite the scarcity of results, the presence of shared constraints is a significant extension, which arises naturally when the agents compete for common resources [55, §2]. However, dealing with coupling constraints in a distributed fashion is extremely challenging. All the results available resort to primal-dual reformulations [50], [109], where the main technical complications are the loss of monotonicity properties of the original problem and the non-uniqueness of dual solutions.

Contributions: Motivated by the above, we develop fully-distributed continuous-time controllers to seek a GNE in networks of multi-integrator agents. We focus on games with separable coupling constraints, played under partial-decision information. Our novel contributions are summarized as follows:

- *Nonlinear coupling constraints:* We introduce primal-dual projected-gradient controllers to drive single-integrator agents to a GNE, with convergence guarantees under strong monotonicity and Lipschitz continuity of the game mapping. In contrast with the existing fully-distributed methods, we allow for arbitrary convex separable (not necessarily affine) coupling constraints. Besides, our schemes are the only continuous-time fully-distributed algorithms for generalized games (except for that in [50], for aggregative games and specific equality constraints only) (§4.3-4.4);

- *Adaptive GNE seeking:* We conceive the first GNE seeking algorithm that can be tuned in a fully decentralized way and without requiring any global information. Specifically, we extend the result in [47] to generalized games and prove that convergence to an equilibrium can be ensured by adopting integral weights in place of a fixed, global, high-enough gain, whose choice would require the knowledge of the algebraic connectivity of the communication graph and of the Lipschitz and strong monotonicity constants of the game mapping (§4.3-4.4);

- *Generalized aggregative games:* We propose controllers for aggregative games with affine aggregation function, where the agents keep and exchange an estimate of the aggregation value only, thus reducing communication and computation cost. Differently from the existing results, e.g., [50], we can handle generic coupling constraints, thanks to a new variant of continuous-time dynamic tracking. Furthermore, we develop an adaptive algorithm that requires no a priori information and virtually no tuning (§4.5);

- *Heterogeneous multi-integrator agents:* We show how all our controllers can be adapted to solve GNE problems where *each* agent is described by mixed-order integrator dynamics, a class never considered before. Importantly, this allows us to address games played by arbitrary nonlinear agents with maximal relative degree, via feedback linearization. To the best of our knowledge, we are the first to study *generalized* games with higher-order dynamical agents (§4.6).

To improve readability, the proofs are in the chapter appendix. We refer to Appendices A, B, C for the basic notation and mathematical background.

4.1.1. PRELIMINARIES: PROJECTED DYNAMICAL SYSTEMS

Given a closed convex set $S \subseteq \mathbb{R}^n$, $\mathrm{T}_S : S \rightrightarrows \mathbb{R}^n : x \mapsto \mathrm{cl}(\bigcup_{\delta>0} \frac{1}{\delta}(S-x))$ is the tangent cone operator of S, where $\mathrm{cl}(\cdot)$ denotes the set closure. The projection on the tangent cone of S at x is $\Pi_S(x,v) := \mathrm{proj}_{\mathrm{T}_S(x)}(v) = \lim_{\delta\to 0^+} \frac{\mathrm{proj}_S(x+\delta v)-x}{\delta}$. By Moreau's Decomposition Theorem [8, Th. 6.30], $v = \mathrm{proj}_{\mathrm{T}_S(x)}(v) + \mathrm{proj}_{\mathrm{N}_S(x)}(v)$ and $\mathrm{proj}_{\mathrm{T}_S(x)}(v)^\top \mathrm{proj}_{\mathrm{N}_S(x)}(v) = 0$, for any $v \in \mathbb{R}^n$.

The following is from [40]. Given an operator $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ and a closed convex set $S \subseteq \mathbb{R}^n$, we consider the projected dynamical system

$$\dot{x} = \Pi_S(x, \mathcal{F}(x)), \qquad x(0) = x_0 \in S. \tag{4.1}$$

In (4.1), the projection operator is possibly discontinuous on the boundary of S. If $\mathcal{F}$ is Lipschitz on S, the system (4.1) admits a unique global Carathéodory solution, i.e., there exists a unique absolutely continuous function $x : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$ such that $x(0) = x_0$, $\dot{x}(t) = \Pi_S(x, g(x))$ for almost all t. Moreover, $x(t) \in S$ for all $t \geq 0$, as on the boundary of S the projection operator restricts the flow of $\mathcal{F}$ such that the solution of (4.1) remains in S (while $\Pi_S(x, \mathcal{F}(x)) = \mathcal{F}(x)$ if $x \in \mathrm{int(S)}$).

Lemma 4.1. Let $S \subseteq \mathbb{R}^q$ be a nonempty closed convex set. For any $y, y' \in S$ and any $\xi \in \mathbb{R}^q$, it holds that $(y-y')^\top \Pi_S(y,\xi) \leq (y-y')^\top \xi$. In particular, if $\Pi_S(y,\xi) = 0$, then $(y-y')^\top \xi \geq 0$ (i.e., $\xi \in \mathrm{N}_S(y)$). □

Proof. By Moreau's theorem, $\big(\xi - \Pi_S(y,\xi)\big) \in \mathrm{N}_S(y)$; thus $\forall y, y' \in S$, $(y'-y)^\top(\xi - \Pi_S(y,\xi)) \leq 0$. ■

4.2. MATHEMATICAL SETUP

We consider a group of agents $\mathcal{I} := \{1, \ldots, N\}$, where each agent $i \in \mathcal{I}$ shall choose its decision variable (i.e., strategy) x_i from its local decision set $\Omega_i \subseteq \mathbb{R}^{n_i}$. Let $x := \mathrm{col}((x_i)_{i\in\mathcal{I}}) \in \Omega$ denote the stacked vector of all the agents' decisions, $\Omega := \times_{i\in\mathcal{I}} \Omega_i \subseteq \mathbb{R}^n$ the overall action space and $n := \sum_{i=1}^N n_i$. The goal of each agent $i \in \mathcal{I}$ is to minimize its objective function $J_i(x_i, x_{-i})$, which depends both on the local strategy x_i and on the decision variables of the other agents $x_{-i} := \mathrm{col}((x_j)_{j\in\mathcal{I}\setminus\{i\}})$. Furthermore, we address *generalized* games, where the coupling among the agents arises also via their feasible decision sets. In particular, we consider separable coupling constraints, so that the overall feasible set is $\mathcal{X} := \Omega \cap \{x \in \mathbb{R}^n \mid g(x) \leq \mathbf{0}_m\}$, where $g : \mathbb{R}^n \to \mathbb{R}^m$, $g(x) := \sum_{i\in\mathcal{I}} g_i(x_i)$, and $g_i : \mathbb{R}^{n_i} \to \mathbb{R}^m$ is a private function of agent i. The game is then represented by N inter-dependent optimization problems:

$$\forall i \in \mathcal{I} : \operatorname*{argmin}_{y_i \in \mathbb{R}^{n_i}} J_i(y_i, x_{-i}) \ \text{ s.t. } \ (y_i, x_{-i}) \in \mathcal{X}. \tag{4.2}$$

The technical problem we consider here is the computation of a GNE, a joint action from which no agent has interest to unilaterally deviate.

Definition 4.1. A collective strategy $x^* = \mathrm{col}((x_i^*)_{i\in\mathcal{I}})$ is a generalized Nash equilibrium if, for all $i \in \mathcal{I}$,

$$x_i^* \in \underset{y_i}{\operatorname{argmin}}\, J_i\left(y_i, x_{-i}^*\right) \text{ s.t. } (y_i, x_{-i}^*) \in \mathcal{X}. \qquad \square$$

Next, we formulate standard convexity and regularity assumptions for the constraints and cost functions ([85, Asm. 1]; [109, Asm. 1]).

Assumption 4.1. For each $i \in \mathcal{I}$, the set Ω_i is closed and convex; g_i is componentwise convex and twice continuously differentiable; $\mathcal{X}$ satisfies Slater's constraint qualification; J_i is continuously differentiable and the function $J_i(\cdot, x_{-i})$ is convex for every x_{-i}. $\square$

Under Assumption 4.1, x^* is a GNE of the game in (4.2) if and only if there exist dual variables $\lambda_i^* \in \mathbb{R}^m$ such that the following KKT conditions are satisfied, for all $i \in \mathcal{I}$ [55, Th. 4.6]:

$$\begin{aligned} \mathbf{0}_{n_i} &\in \nabla_{x_i} J_i(x_i^*, x_{-i}^*) + \tfrac{\partial}{\partial x_i} g_i(x_i^*)^\top \lambda_i^* + \mathrm{N}_{\Omega_i}\left(x_i^*\right) \\ \mathbf{0}_m &\in -g(x^*) + \mathrm{N}_{\mathbb{R}^m_{\geq 0}}^m\left(\lambda_i^*\right). \end{aligned} \tag{4.3}$$

Specifically, we focus on the subclass of v-GNEs [55, Def. 3.11], namely GNEs with equal dual variables, i.e. $\lambda_i^* = \lambda^* \in \mathbb{R}^m$ for all $i \in \mathcal{I}$, for which the KKT conditions read as

$$\mathbf{0}_n \in F\left(x^*\right) + \tfrac{\partial}{\partial x} g(x^*)^\top \lambda^* + \mathrm{N}_\Omega(x^*) \tag{4.4a}$$

$$\mathbf{0}_m \in -g(x^*) + \mathrm{N}_{\mathbb{R}^m_{\geq 0}}^m\left(\lambda^*\right). \tag{4.4b}$$

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

$$F(x) := \mathrm{col}\left(\left(\nabla_{x_i} J_i(x_i, x_{-i})\right)_{i\in\mathcal{I}}\right). \tag{4.5}$$

Variational equilibria enjoys important structural properties, such as economic fairness [55]. For example, in electricity markets, the dual variables correspond to unitary prices charged for the use of the infrastructure by an administrator that aim at maximizing its revenue while ensuring certain operating conditions, and it is reasonable to assume that the administrator cannot charge discriminatory prices to different energy producers [85]. A sufficient condition for the existence and uniqueness of a v-GNE is the strong monotonicity of the pseudo-gradient [149, Th. 1, Rem. 1], which was always postulated in continuous-time NE seeking under partial-decision information ([63, Asm. 2]; [50, Asm. 3]). It implies strong convexity of the functions $J_i(\cdot, x_{-i})$ for any x_{-i} [137, Rem. 1], but not necessarily convexity of J_i in the full argument.

Assumption 4.2. The game mapping F in (4.5) is:

(i) μ-strongly monotone, for some $\mu > 0$;

(ii) θ_0-Lipschitz continuous, for some $\theta_0 > 0$. $\square$

Algorithm 4.1. Constant gain

Initialization: set $c > \underline{c} := \frac{(\theta_0+\theta)^2+4\mu\theta}{4\mu\lambda_2(L)}$; $\forall i \in \mathcal{I}$, set $\boldsymbol{x}_{i,-i}(0) \in \mathbb{R}^{n-n_i}$, $z_i(0) = \mathbf{0}_m$, $\lambda_i(0) \in \mathbb{R}^m_{\geq 0}$;

Dynamics: $\forall i \in \mathcal{I}$,

$$\begin{aligned}
\dot{x}_i &= u_i = \Pi_{\Omega_i}\left(x_i, -\nabla_{x_i} J_i(x_i, \boldsymbol{x}_{i,-i}) - \tfrac{\partial}{\partial x_i} g_i(x_i)^\top \lambda_i - c\textstyle\sum_{j\in\mathcal{N}_i}(x_i - \boldsymbol{x}_{j,i})\right)\\
\dot{\boldsymbol{x}}_{i,-i} &= -c\textstyle\sum_{j\in\mathcal{N}_i}(\boldsymbol{x}_{i,-i} - \boldsymbol{x}_{j,-i})\\
\dot{z}_i &= \textstyle\sum_{j\in\mathcal{N}_i}(\lambda_i - \lambda_j)\\
\dot{\lambda}_i &= \Pi_{\mathbb{R}^m_{\geq 0}}\left(\lambda_i, g_i(x_i) - z_i - \textstyle\sum_{j\in\mathcal{N}_i}(\lambda_i - \lambda_j)\right)
\end{aligned}$$

4

4.3. FULLY-DISTRIBUTED EQUILIBRIUM SEEKING

IN this section, we consider the game in (4.2), where each agent is associated with the following dynamical system:

$$\forall i \in \mathcal{I}: \quad \dot{x}_i = u_i,\ x_i(0) \in \Omega_i. \tag{4.6}$$

Our aim is to design the inputs $u_i \in \mathbb{R}^{n_i}$ to seek a v-GNE in a fully-distributed way. Specifically, each agent $i \in \mathcal{I}$ only knows its own feasible set Ω_i, the portion g_i of the coupling constraints, and its own cost function J_i. Moreover, the agents cannot access the strategies of all the competitors x_{-i}. Instead, each agent only relies on the information exchanged locally with some neighbors over a communication network $\mathcal{G}(\mathcal{I},\mathcal{E})$. The pairs $(i,j),(j,i)$ belong to the set of edges $\mathcal{E}$ if and only if agent i and j can exchange information. We denote by $W \in \mathbb{R}^{N\times N}$ the symmetric weight matrix of $\mathcal{G}$, with $[W]_{i,j} > 0$ if $(i,j) \in \mathcal{E}$, $[W]_{i,j} = 0$ otherwise; L the symmetric Laplacian matrix of $\mathcal{G}$; $\mathcal{N}_i := \{j \mid (i,j) \in \mathcal{E}\}$ the set of neighbors of agent i. For ease of notation, we assume that the graph is unweighted, i.e., $[W]_{i,j} = 1$ if $(i,j) \in \mathcal{E}$, but our results still hold for the weighted case.

Assumption 4.3. The communication graph $\mathcal{G}(\mathcal{I},\mathcal{E})$ is undirected, unweighted and connected. □

Our first algorithm is inspired by the discrete-time primal-dual gradient iteration in [109, Alg. 1]. To cope with the lack of knowledge, the general assumption for the partial-decision information scenario is that each agent keeps an estimate of all other agents' actions [109], [137]. Let $\boldsymbol{x}_i := \mathrm{col}((\boldsymbol{x}_{i,j})_{j\in\mathcal{I}}) \in \mathbb{R}^n$, where $\boldsymbol{x}_{i,i} := x_i$ and $\boldsymbol{x}_{i,j}$ is agent i's estimate of agent j's action, for all $j \neq i$; $\boldsymbol{x}_{j,-i} := \mathrm{col}((\boldsymbol{x}_{j,\ell})_{\ell\in\mathcal{I}\setminus\{i\}})$. Each agent also keeps an estimate $\lambda_i \in \mathbb{R}^m_{\geq 0}$ of the dual variable and an auxiliary variable $z_i \in \mathbb{R}^m$ to allow for distributed consensus of the dual estimates. Our proposed dynamics are summarized in Algorithm 4.1, where $c > 0$ is a *global* fixed parameter (and θ is a constant defined in Lemma 4.3).

We note that the agents exchange $\{\boldsymbol{x}_i, \lambda_i\}$ with their neighbors only, therefore the controller can be implemented distributedly. Importantly, each agent i evaluates the partial gradient of its cost $\nabla_{x_i} J_i$ on its local estimate $\boldsymbol{x}_i$, not on the actual joint strategy x. In steady state, the agents should agree on their estimates, i.e., $\boldsymbol{x}_i = \boldsymbol{x}_j$, $\lambda_i = \lambda_j$, for all

Algorithm 4.2. Adaptive gains

Initialization: $\forall i \in \mathcal{I}$, set $\gamma_i > 0$, $\boldsymbol{x}_{i,-i}(0) \in \mathbb{R}^{n-n_i}$, $k_i(0) \in \mathbb{R}$, $z_i(0) = \mathbf{0}_m$, $\lambda_i(0) \in \mathbb{R}^m$;

Dynamics: $\forall i \in \mathcal{I}$,

$$\begin{aligned}
\dot{x}_i &= u_i = \Pi_{\Omega_i}\left(x_i, -\nabla_{x_i} J_i(x_i, \boldsymbol{x}_{i,-i}) - \tfrac{\partial}{\partial x_i} g_i(x_i)^\top \lambda_i - \textstyle\sum_{j\in\mathcal{N}_i}(k_i\rho_{i,i} - k_j\rho_{j,i})\right)\\
\dot{\boldsymbol{x}}_{i,-i} &= -\textstyle\sum_{j\in\mathcal{N}_i}(k_i\rho_{i,-i} - k_j\rho_{j,-i})\\
\dot{k}_i &= \gamma_i \|\rho_i\|^2, \qquad \rho_i = \textstyle\sum_{j\in\mathcal{N}_i}(\boldsymbol{x}_i - \boldsymbol{x}_j)\\
\dot{z}_i &= \textstyle\sum_{j\in\mathcal{N}_i}\left(\lambda_i - \lambda_j\right)\\
\dot{\lambda}_i &= \Pi_{\mathbb{R}^m_{\geq 0}}\left(\lambda_i, g_i(x_i) - z_i - \textstyle\sum_{j\in\mathcal{N}_i}(\lambda_i - \lambda_j)\right)
\end{aligned}$$

$i, j \in \mathcal{I}$. This motivates the presence of consensual terms for both primal and dual variables. For any integer q, we denote $\boldsymbol{E}_q := \{\boldsymbol{y} \in \mathbb{R}^{Nq} : \boldsymbol{y} = \mathbf{1}_N \otimes y, y \in \mathbb{R}^q\}$ the consensus subspace of dimension q, and $\boldsymbol{E}_q^\perp := \{\boldsymbol{y} \in \mathbb{R}^{Nq} : \left(\mathbf{1}_N^\top \otimes I_q\right)\boldsymbol{y} = \mathbf{0}_q\}$ its orthogonal complement; Specifically, $\boldsymbol{E}_n$ and $\boldsymbol{E}_m$ are the action and multiplier consensus subspaces, respectively. Moreover, $P_q := \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top \otimes I_q$ is the projection matrix onto $\boldsymbol{E}_q$, i.e., $P_q\boldsymbol{y} = \text{proj}_{\boldsymbol{E}_q}(\boldsymbol{y})$, and $P_q^\perp := I_{Nq} - P_q$ the projection matrix onto the disagreement subspace $\boldsymbol{E}_q^\perp$.

While Algorithm 4.1 is fully-distributed, choosing the gain c requires global knowledge about the graph $\mathcal{G}$, i.e., the algebraic connectivity, and about the game mapping, i.e., the strong monotonicity and Lipschitz constants. These parameters are unlikely to be available locally in a network system. To overcome this limitation and enhance scalability, [47] proposed a controller where the communication gains are tuned online, thus relaxing the need for global information, for games without coupling constraints. Here we extend their solution to the GNE problem.

Our proposed controller is given in Algorithm 4.2. For all $i \in \mathcal{I}$, k_i is the adaptive gain of agent i, $\gamma_i > 0$ is an arbitrary local constant and $\rho_i := \text{col}((\rho_{i,j})_{j\in\mathcal{I}})$. We emphasize that Algorithm 4.2 allows for a fully uncoupled tuning: each agent chooses *locally* the initial conditions and the parameter γ_i, independently of the other agents and without need for coordination or global knowledge.

Remark 4.1. Algorithm 4.2 uses second order information, as each agent sends the quantity ρ_i, which depends on the estimates of its neighbors. In case of delayed communication, this means dealing with twice the transmission latency with respect to a controller that exploits first order information only, e.g., Algorithm 4.1. In a discrete-time setting, a sampled version of Algorithm 4.2 can be implemented by allowing the agents to communicate twice per iteration, a common assumption for GNE seeking on networks [64], [109]. □

To rewrite the closed-loop dynamics in Algorithms 4.1, 4.2 in compact form, let us define $\boldsymbol{x} := \text{col}((\boldsymbol{x}_i)_{i\in\mathcal{I}})$ and, as in [63, Eq. 11], for all $i \in \mathcal{I}$,

$$\mathcal{R}_i := [\ \mathbf{0}_{n_i \times n_{<i}} \quad I_{n_i} \quad \mathbf{0}_{n_i \times n_{>i}}\ ], \tag{4.7}$$

where $n_{<i} := \sum_{j<i, j\in\mathcal{I}} n_j$, $n_{>i} := \sum_{j>i, j\in\mathcal{I}} n_j$; let also $\mathcal{R} := \text{diag}\left((\mathcal{R}_i)_{i\in\mathcal{I}}\right)$. In simple terms, $\mathcal{R}_i$ selects the i-th n_i dimensional component from an n-dimensional vector, i.e., $\mathcal{R}_i\boldsymbol{x}_i = \boldsymbol{x}_{i,i} = x_i$ and $x = \mathcal{R}\boldsymbol{x}$.

Let $\boldsymbol{\lambda} := \operatorname{col}((\lambda_i)_{i\in\mathcal{I}})$, $\boldsymbol{z} := \operatorname{col}((z_i)_{i\in\mathcal{I}})$, $\boldsymbol{\Omega} := \{\boldsymbol{x} \in \mathbb{R}^{nN} \mid \mathcal{R}\boldsymbol{x} \in \Omega\}$, $\boldsymbol{g}(x) := \operatorname{col}((g_i(x_i)_{i\in\mathcal{I}}))$, $\mathbb{G}(x) := \frac{\partial}{\partial x}\boldsymbol{g}(x) = \operatorname{diag}((\frac{\partial}{\partial x_i} g_i(x_i))_{i\in\mathcal{I}})$, $\boldsymbol{k} := \operatorname{col}((k_i)_{i\in\mathcal{I}})$, $\boldsymbol{\rho} := \operatorname{col}((\rho_i)_{i\in\mathcal{I}})$, $K := \operatorname{diag}((k_i I_n)_{i\in\mathcal{I}})$, $D(\boldsymbol{\rho}) := \operatorname{diag}((\rho_i)_{i\in\mathcal{I}})$, $\Gamma := \operatorname{diag}((\gamma_i)_{i\in\mathcal{I}})$, and, for any integer $q > 0$, $\boldsymbol{L}_q := L \otimes I_q$. Furthermore, we define the *extended pseudo-gradient* mapping $\boldsymbol{F}$ as:

$$\boldsymbol{F}(\boldsymbol{x}) := \operatorname{col}((\nabla_{x_i} J_i(x_i, \boldsymbol{x}_{i,-i}))_{i\in\mathcal{I}}). \tag{4.8}$$

Therefore, Algorithm 4.1, in compact form, reads as

$$\dot{\boldsymbol{x}} = \Pi_{\boldsymbol{\Omega}}\left(\boldsymbol{x}, -\mathcal{R}^\top(\boldsymbol{F}(\boldsymbol{x}) + \mathbb{G}(\mathcal{R}\boldsymbol{x})^\top\boldsymbol{\lambda}) - c\boldsymbol{L}_n\boldsymbol{x}\right) \tag{4.9a}$$

$$\dot{\boldsymbol{z}} = \boldsymbol{L}_m\boldsymbol{\lambda} \tag{4.9b}$$

$$\dot{\boldsymbol{\lambda}} = \Pi_{\mathbb{R}^{Nm}_{\geq 0}}(\boldsymbol{\lambda}, \boldsymbol{g}(\mathcal{R}\boldsymbol{x}) - \boldsymbol{z} - \boldsymbol{L}_m\boldsymbol{\lambda}), \tag{4.9c}$$

and Algorithm 4.2 as

$$\dot{\boldsymbol{x}} = \Pi_{\boldsymbol{\Omega}}(\boldsymbol{x}, -\mathcal{R}^\top(\boldsymbol{F}(\boldsymbol{x}) + \mathbb{G}(\mathcal{R}\boldsymbol{x})^\top\boldsymbol{\lambda}) - \boldsymbol{L}_n K\boldsymbol{\rho}) \tag{4.10a}$$

$$\dot{\boldsymbol{k}} = D\left(\boldsymbol{\rho}\right)^\top(\Gamma \otimes I_n)\,\boldsymbol{\rho}, \qquad \boldsymbol{\rho} = \boldsymbol{L}_n\boldsymbol{x} \tag{4.10b}$$

$$\dot{\boldsymbol{z}} = \boldsymbol{L}_m\boldsymbol{\lambda} \tag{4.10c}$$

$$\dot{\boldsymbol{\lambda}} = \Pi_{\mathbb{R}^{Nm}_{\geq 0}}(\boldsymbol{\lambda}, \boldsymbol{g}(\mathcal{R}\boldsymbol{x}) - \boldsymbol{z} - \boldsymbol{L}_m\boldsymbol{\lambda}). \tag{4.10d}$$

4.4. CONVERGENCE ANALYSIS

In this section, we show the convergence of our dynamics to a v-GNE. We focus on the analysis of Algorithm 4.2, which presents more technical difficulties; the convergence of Algorithm 4.1 can be shown analogously.

We start by noting an invariance property of our controllers, namely that if $\boldsymbol{z}(0) \in \boldsymbol{E}_m^\perp$ (for instance, $\boldsymbol{z}(0) = \boldsymbol{0}_m$), then $\boldsymbol{z} \in \boldsymbol{E}_m^\perp$ along any solution of (4.10), by (4.10c). The next lemma relates a class of equilibria of the system in (4.10) to the v-GNE of the game in (4.2).

Lemma 4.2. Under Assumptions 4.1, 4.2, 4.3, the following statements hold:

i) Any equilibrium point $\operatorname{col}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}})$ of (4.10) with $\bar{\boldsymbol{z}} \in \boldsymbol{E}_m^\perp$ is such that $\bar{\boldsymbol{x}} = \boldsymbol{1}_N \otimes x^*$, $\bar{\boldsymbol{\lambda}} = \boldsymbol{1}_N \otimes \lambda^*$, where the pair (x^*, λ^*) satisfies the KKT conditions in (4.4), hence x^* is the v-GNE of the game in (4.2).

ii) The system (4.10) admits at least one equilibrium $\operatorname{col}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}})$ with $\bar{\boldsymbol{z}} \in \boldsymbol{E}_m^\perp$. □

We remark that in Algorithm 4.2 (or 4.1) each agent i evaluates the quantity $\nabla_{x_i} J_i$ in its local estimate $\boldsymbol{x}_i$, not on x. As a consequence, the operator $\mathcal{R}^\top\boldsymbol{F}$ is very rarely monotone, even under strong monotonicity of the game mapping F. The loss of monotonicity is indeed the main technical difficulty arising in the partial-decision information scenario. Following [109], [47], we deal with this issue by leveraging a restricted monotonicity property, which can be guaranteed for any game satisfying Assumptions 4.1-4.3, without additional hypotheses, as shown in the next lemmas. The proof relies on the decomposition of $\boldsymbol{x}$ along the consensus space $\boldsymbol{E}_n$, where $\boldsymbol{F}$ is strongly monotone, and the disagreement space $\boldsymbol{E}_n^\perp$, where $\boldsymbol{L}_n$ is strongly monotone.

Lemma 4.3 ([23, Lem. 3])**.** Let Assumption 4.2 hold. Then, the mapping $\boldsymbol{F}$ in (4.8) is θ-Lipschitz continuous, for some $\theta \in [\mu, \theta_0]$. □

Lemma 4.4. Let Assumptions 4.2, 4.3 hold, and let

$$M_1 := \begin{bmatrix} \frac{\mu}{N} & -\frac{\theta_0+\theta}{2\sqrt{N}} \\ -\frac{\theta_0+\theta}{2\sqrt{N}} & k^*\lambda_2(L)^2 - \theta \end{bmatrix}, \quad \underline{k} := \frac{(\theta_0+\theta)^2+4\mu\theta}{4\mu\lambda_2(L)^2}. \tag{4.11}$$

For any $k^* > \underline{k}$ and $K^* = I_{Nn}k^*$, for any $\boldsymbol{x} \in R^{Nn}$ and any $\boldsymbol{y} \in \boldsymbol{E}_n$, it holds that $M_2 \succ 0$ and also that

$$\left(\boldsymbol{x}-\boldsymbol{y}\right)^\top \left(\mathcal{R}^\top\left(\boldsymbol{F}(\boldsymbol{x})-\boldsymbol{F}\left(\boldsymbol{y}\right)\right)+\boldsymbol{L}_n K^* \boldsymbol{L}_n\left(\boldsymbol{x}-\boldsymbol{y}\right)\right) \geq \lambda_{\min}(M_1)\left\|\boldsymbol{x}-\boldsymbol{y}\right\|^2. \qquad \square$$

We can now present the main result of this section.

Theorem 4.1 (*Convergence of Algorithm 4.2*)**.** Let Assumptions 4.1, 4.2, 4.3 hold. For any initial condition in $S = \boldsymbol{\Omega} \times \mathbb{R}^N \times \boldsymbol{E}_m^\perp \times \mathbb{R}^{mN}_{\geq 0}$, the system in (4.10) has a unique Carathéodory solution, which belongs to S for all $t \geq 0$. The solution converges to an equilibrium $\operatorname{col}\left(\bar{\boldsymbol{x}}, \bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}}\right)$, with $\bar{\boldsymbol{x}} = \mathbf{1}_N \otimes x^*$, $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$, and (x^*, λ^*) satisfies the KKT conditions in (4.4), hence x^* is the v-GNE of the game in (4.2). □

A similar result holds also for the the dynamics in (4.9).

Theorem 4.2 (*Convergence of Algorithm 4.1*)**.** Let Assumptions 4.1, 4.2, 4.3 hold. Let $c > \underline{c}$, with $\underline{c}$ as in Algorithm 4.1. For any initial condition in $S = \boldsymbol{\Omega} \times \boldsymbol{E}_m^\perp \times \mathbb{R}^{mN}_{\geq 0}$ the system in (4.9) has a unique Carathéodory solution, which belongs to S for all $t \geq 0$. The solution converges to an equilibrium $\operatorname{col}\left(\bar{\boldsymbol{x}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}}\right)$, with $\bar{\boldsymbol{x}} = \mathbf{1}_N \otimes x^*$, $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$, and (x^*, λ^*) satisfies the KKT conditions in (4.4), hence x^* is the v-GNE of the game in (4.2). □

Remark 4.2. As for Euclidean projections, evaluating $\Pi_{\Omega_i}(x, v)$ can be computationally expensive. If, for some $i \in \mathcal{I}$ and some twice continuously differentiable mapping g_i^{loc}, $\Omega_i = \{x_i \in \mathbb{R}^{n_i} \mid g_i^{\text{loc}}(x_i) \leq \mathbf{0}_p\}$, then the following alternative updates can be used in Algorithm 4.1 (and similarly in Algorithm 4.2):

$$\begin{aligned} \dot{x}_i &= -\nabla_{x_i} J_i(x_i, \boldsymbol{x}_{i,-i}) - \tfrac{\partial}{\partial x_i} g_i(x_i)^\top \lambda_i - \tfrac{\partial}{\partial x_i} g_i^{\text{loc}}(x_i)^\top \lambda_i^{\text{loc}} - c\textstyle\sum_{j\in\mathcal{N}_i}(x_i - \boldsymbol{x}_{j,i}) \\ \dot{\lambda}_i^{\text{loc}} &= \Pi_{\mathbb{R}^p_{\geq 0}}(\lambda_i, g_i^{\text{loc}}(x_i)). \end{aligned}$$

In simple terms, the local constraints are dualized like the coupling constraints; but the corresponding dual variables are managed locally. The drawback of this primal-dual approach is that the satisfaction of the local constraints can only be ensured asymptotically. □

4.5. GENERALIZED AGGREGATIVE GAMES

In this section, we study aggregative games, where the cost function of each agent depends only on the local action and on an aggregation value $\psi(x) := \frac{1}{N}\sum_{i\in\mathcal{I}}\psi_i(x_i)$,

Algorithm 4.3. Constant gain (aggregative games)

Initialization: set $c > \underline{c} := \frac{(\tilde{\theta}_\sigma)^2}{4\mu\lambda_2(L)}$; $\forall i \in \mathcal{I}$, set $\varsigma_i = \mathbf{0}_{\bar{n}}$, $z_i(0) = \mathbf{0}_m$, $\lambda_i(0) \in \mathbb{R}^m_{\geq 0}$;

Dynamics: $\forall i \in \mathcal{I}$,

$$\begin{aligned}
&\dot{x}_i = u_i = \Pi_{\Omega_i}(x_i, -\nabla_{x_i} f_i(x_i,\sigma_i) - \tfrac{\partial}{\partial x_i} g_i(x_i)^\top \lambda_i - cB_i^\top \textstyle\sum_{j\in\mathcal{N}_i}(\sigma_i - \sigma_j))\\
&\dot{\varsigma}_i = -c\textstyle\sum_{j\in\mathcal{N}_i}(\sigma_i - \sigma_j), \qquad \sigma_i = \psi_i(x_i) + \varsigma_i\\
&\dot{z}_i = \textstyle\sum_{j\in\mathcal{N}_i}(\lambda_i - \lambda_j)\\
&\dot{\lambda}_i = \Pi_{\mathbb{R}^m_{\geq 0}}(\lambda_i, g_i(x_i) - z_i - \textstyle\sum_{j\in\mathcal{N}_i}(\lambda_i - \lambda_j))
\end{aligned}$$

Algorithm 4.4. Adaptive gains (aggregative games)

Initialization: $\forall i \in \mathcal{I}$, set $\gamma_i > 0$, $\varsigma_i = \mathbf{0}_{\bar{n}}$, $k_i(0) \in \mathbb{R}$, $z_i(0) = \mathbf{0}_m$, $\lambda_i(0) \in \mathbb{R}^m_{\geq 0}$;

Dynamics: $\forall i \in \mathcal{I}$,

$$\begin{aligned}
&\dot{x}_i = u_i = \Pi_{\Omega_i}(x_i, -\nabla_{x_i} f_i(x_i,\sigma_i) - \tfrac{\partial}{\partial x_i} g_i(x_i)^\top \lambda_i - B_i^\top \textstyle\sum_{j\in\mathcal{N}_i}(k_i\rho_i - k_j\rho_j))\\
&\dot{\varsigma}_i = -\textstyle\sum_{j\in\mathcal{N}_i}(k_i\rho_i - k_j\rho_j) \qquad \sigma_i = \psi_i(x_i) + \varsigma_i\\
&\dot{k}_i = \gamma_i\|\rho_i\|^2 \qquad \rho_i = \textstyle\sum_{j\in\mathcal{N}_i}(\sigma_i - \sigma_j)\\
&\dot{z}_i = \textstyle\sum_{j\in\mathcal{N}_i}(\lambda_i - \lambda_j)\\
&\dot{\lambda}_i = \Pi_{\mathbb{R}^m_{\geq 0}}(\lambda_i, g_i(x_i) - z_i - \textstyle\sum_{j\in\mathcal{N}_i}(\lambda_i - \lambda_j))
\end{aligned}$$

where $\psi_i : \mathbb{R}^{n_i} \to \mathbb{R}^{\bar{n}}$, for all $i \in \mathcal{I}$. It follows that, for each $i \in \mathcal{I}$, there is a function $f_i : \mathbb{R}^{n_i} \times \mathbb{R}^{\bar{n}} \to \mathbb{R}$ such that the original cost function J_i in (4.2) can be written as

$$J_i(x_i, x_{-i}) = f_i(x_i, \psi(x)). \tag{4.12}$$

In particular, we focus on games with affine aggregation functions, where, for all $i \in \mathcal{I}$, $\psi_i(x_i) = B_i x_i + d_i$, for some $B_i \in \mathbb{R}^{\bar{n}\times n_i}$, $d_i \in \mathbb{R}^{\bar{n}}$. As a special case, this class includes the common (weighted) average aggregative games.

Since an aggregative game is only a particular instance of the game in (4.2), Algorithms 4.1-4.2 could still be used to drive the system (4.6) to the v-GNE. This would require each agent i to keep (and exchange) an estimate of all other agents' actions, i.e., a vector of $n - n_i$ components; however, the cost of each agent is only a function of the aggregation value $\psi(x)$, whose dimension $\bar{n}$ is independent of the number of agents. To reduce the communication and computation burden, we introduce two distributed controllers that are scalable with the number of agents, specifically designed to seek a v-GNE in aggregative games. Our proposed dynamics are shown in Algorithms 4.3 and 4.4.

Since the agents rely on local information only, they do not have access to the actual value of the aggregation $\psi(x)$. Hence, we embed each agent with an auxiliary error variable $\varsigma_i \in \mathbb{R}^{\bar{n}}$, which is an estimate of $\psi(x) - \psi_i(x_i)$. Each agent aims at asymptotically reconstructing the true aggregation value, based on the information received from its neighbors. We use the notation

$$\nabla_{x_i} f_i(x_i,\sigma_i) := \nabla_y f_i(y,\sigma_i)|_{y=x_i} + \tfrac{1}{N} B_i^\top \nabla_y f_i(x_i,y)|_{y=\sigma_i}.$$

We note that, in Algorithms 4.3 and 4.4, the agents exchange the quantities $\sigma_i \in \mathbb{R}^{\bar{n}}$, instead of the variables $\boldsymbol{x}_i, \rho_i \in \mathbb{R}^n$, like in Algorithms 4.1 and 4.2. Let $\boldsymbol{\sigma} := \operatorname{col}((\sigma_i)_{i\in\mathcal{I}})$. We define the *extended pseudo-gradient* mapping $\tilde{\boldsymbol{F}}$ as

$$\tilde{\boldsymbol{F}}(x,\boldsymbol{\sigma}) := \operatorname{col}\big(\big(\nabla_{x_i} f_i(x_i,\sigma_i)\big)_{i\in\mathcal{I}}\big). \tag{4.13}$$

Assumption 4.4. The mapping $\tilde{\boldsymbol{F}}$ in (4.13) is $\tilde{\theta}$-Lipschitz continuous, for some $\tilde{\theta} > 0$. Hence, $\tilde{\boldsymbol{F}}(x,\cdot)$ is $\tilde{\theta}_\sigma$-Lipschitz continuous, for some $\tilde{\theta}_\sigma \in (0,\tilde{\theta}]$, $\forall x \in \mathbb{R}^n$. □

Assumption 4.4 is standard ([64, Asm. 4]; [82, Asm. 3]) and can be shown to hold under Assumption 4.2 if the matrix $[B_1 \dots B_N]$ is full row rank, e.g., for average aggregative games.

By defining $\boldsymbol{\varsigma} := \operatorname{col}((\varsigma_i)_{i\in\mathcal{I}})$, $K := \operatorname{diag}((k_i I_{\bar{n}})_{i\in\mathcal{I}})$, $\boldsymbol{\psi}(x) := \operatorname{col}((\psi_i(x_i))_{i\in\mathcal{I}})$, $B := \operatorname{diag}((B_i)_{i\in\mathcal{I}})$, the dynamics in Algorithms 4.3 and 4.4 read, in compact form, as

$$\dot{x} = \Pi_\Omega(x, -\tilde{\boldsymbol{F}}(x,\boldsymbol{\sigma}) - \mathbb{G}(x)^\top \boldsymbol{\lambda} - cB^\top \boldsymbol{L}_{\bar{n}}\boldsymbol{\sigma}) \tag{4.14a}$$
$$\dot{\boldsymbol{\varsigma}} = -c\boldsymbol{L}_{\bar{n}}\boldsymbol{\sigma}, \qquad \boldsymbol{\sigma} = \boldsymbol{\psi}(x) + \boldsymbol{\varsigma} \tag{4.14b}$$
$$\dot{\boldsymbol{z}} = \boldsymbol{L}_m \boldsymbol{\lambda} \tag{4.14c}$$
$$\dot{\boldsymbol{\lambda}} = \Pi_{\mathbb{R}^{Nm}_{\geq 0}}\big(\boldsymbol{\lambda}, \boldsymbol{g}(x) - \boldsymbol{z} - \boldsymbol{L}_m\boldsymbol{\lambda}\big), \tag{4.14d}$$

and

$$\dot{x} = \Pi_\Omega(x, -\tilde{\boldsymbol{F}}(x,\boldsymbol{\sigma}) - \mathbb{G}(x)^\top \boldsymbol{\lambda} - B^\top \boldsymbol{L}_{\bar{n}} K\boldsymbol{\rho}) \tag{4.15a}$$
$$\dot{\boldsymbol{\varsigma}} = -\boldsymbol{L}_{\bar{n}} K\boldsymbol{\rho}, \qquad \boldsymbol{\sigma} = \boldsymbol{\psi}(x) + \boldsymbol{\varsigma} \tag{4.15b}$$
$$\dot{\boldsymbol{k}} = D\big(\boldsymbol{\rho}\big)^\top (\Gamma \otimes I_{\bar{n}})\,\boldsymbol{\rho}, \qquad \boldsymbol{\rho} = \boldsymbol{L}_{\bar{n}}\boldsymbol{\sigma} \tag{4.15c}$$
$$\dot{\boldsymbol{z}} = \boldsymbol{L}_m \boldsymbol{\lambda} \tag{4.15d}$$
$$\dot{\boldsymbol{\lambda}} = \Pi_{\mathbb{R}^{Nm}_{\geq 0}}\big(\boldsymbol{\lambda}, \boldsymbol{g}(x) - \boldsymbol{z} - \boldsymbol{L}_m\boldsymbol{\lambda}\big), \tag{4.15e}$$

respectively. We note that only if the estimates of all the agents coincide with the actual value, i.e., $\boldsymbol{\sigma} = \mathbf{1}_N \otimes \psi(x)$, we can conclude that $\tilde{\boldsymbol{F}}(x,\boldsymbol{\sigma}) = F(x)$, F as in (4.5).

Remark 4.3. From the updates in (4.14b) (or (4.15b)), we can infer an invariance property of the closed-loop system (4.14) (or (4.15)), namely that, at any time, $\frac{1}{N}\sum_{i\in\mathcal{I}} \varsigma_i = \mathbf{0}_{\bar{n}}$, and thus $\frac{1}{N}\sum_{i\in\mathcal{I}} \sigma_i = \psi(x)$ (or equivalently, $P_{\bar{n}}\boldsymbol{\sigma} = \mathbf{1}_N \otimes \psi(x)$), provided that $\boldsymbol{\varsigma}(0) = \mathbf{0}_{N\bar{n}}$. In fact, the dynamics of σ_i in Algorithm 4.3 can be regarded as a continuous-time dynamic tracking for the time-varying quantity $\psi(x)$, i.e., $\sigma_i(0) = \psi_i(x_i(0))$ and

$$\dot{\sigma}_i = -c\textstyle\sum_{j\in\mathcal{N}_i}(\sigma_i - \sigma_j) + \frac{d}{dt}(\psi_i(x_i)). \tag{4.16}$$

We emphasize that in Algorithm 4.3 there is no agent that knows the quantity $\psi(x)$. This is the main difference with respect to Algorithm 4.1, where the consensus of the estimates works instead as a leader-follower protocol. If the actions x are constant, the dynamics in (4.16) reduce to a standard average consensus algorithm and ensure that

$\boldsymbol{\sigma} \to \mathbf{1}_N \otimes \psi(x)$ exponentially, under Assumption 4.3. Therefore, when the action dynamics (4.14a) are *input-to-state-stable* (ISS) with respect to the estimation error, convergence can be ensured via small-gain arguments (for c big enough) – a similar approach was used in [50]. However, in the presence of generic coupling constraints (even affine), this robustness cannot be guaranteed; to still ensure convergence, we design an extra consensual term for the action updates, i.e., $cB^\top \boldsymbol{L}_{\bar{n}} \boldsymbol{\sigma}$. Furthermore, via the *error* variable $\boldsymbol{\varsigma}$, we avoid studying the discontinuous dynamics in (4.16). We finally note that we consider games with affine ψ (a broader class than [64]), but nonlinear aggregation functions are also studied [49], [50], [154]. However, [49] and [154] postulate strong monotonicity of an augmented operator, a condition much more restrictive than our Assumption 4.2(i) [49, Rem. 2]; instead, the approach in [50] is not suitable to deal with generic coupling constraints, as discussed above. □

By leveraging the invariance property in Remark 4.3, we can obtain a refinement of Lemma 4.4.

Lemma 4.5. Let Assumptions 4.2(i), 4.3, 4.4 hold, and let

$$M_2 = \begin{bmatrix} \mu & -\frac{\tilde{\theta}_\sigma}{2} \\ -\frac{\tilde{\theta}_\sigma}{2} & k^* \lambda_2(L)^2 \end{bmatrix}, \quad \underline{k} = \frac{\tilde{\theta}_\sigma^2}{4\mu\lambda_2(L)^2} \tag{4.17}$$

For any $k^* > \underline{k}$ and $K^* = I_{N\bar{n}} k^*$, for any $(x, \boldsymbol{\sigma})$ such that $P_{\bar{n}}\boldsymbol{\sigma} = P_{\bar{n}}\boldsymbol{\psi}(x)$ and any $(x', \boldsymbol{\sigma}')$ such that $\boldsymbol{\sigma}' = P_{\bar{n}}\boldsymbol{\psi}(x') = \mathbf{1}_N \otimes \psi(x')$, it holds that $M_2 \succ 0$, and that

$$\begin{aligned} &(x - x')^\top (\tilde{\boldsymbol{F}}(x, \boldsymbol{\sigma}) - \tilde{\boldsymbol{F}}(x', \boldsymbol{\sigma}')) + (\boldsymbol{\sigma} - \boldsymbol{\sigma}')^\top \boldsymbol{L}_{\bar{n}} K^* \boldsymbol{L}_{\bar{n}} (\boldsymbol{\sigma} - \boldsymbol{\sigma}') \\ &\quad \geq \lambda_{\min}(M_2) \left\| \operatorname{col}\left(x - x', \boldsymbol{\sigma} - \mathbf{1}_N \otimes \psi(x)\right) \right\|^2. \end{aligned}$$ □

Next, we exploit Lemma 4.5 to prove the convergence of Algorithm 4.4. An analogous result holds for Algorithm 4.3.

Theorem 4.3 (*Convergence of Algorithm 4.4*)**.** Let Assumptions 4.1, 4.2(i), 4.3, 4.4 hold. Then, for any initial condition in $S = \Omega \times \boldsymbol{E}_{\bar{n}}^\perp \times \mathbb{R}^N \times \boldsymbol{E}_m^\perp \times \mathbb{R}_{\geq 0}^{mN}$ the system in (4.15) has a unique Carathéodory solution, which belongs to S for all $t \geq 0$. The solution converges to an equilibrium $\operatorname{col}\left(\bar{x}, \bar{\boldsymbol{\varsigma}}, \bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}}\right)$, with $\boldsymbol{\psi}(\bar{x}) + \bar{\boldsymbol{\varsigma}} = \mathbf{1}_N \otimes \psi(\bar{x})$, $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$, and $(\bar{x}, \lambda^*)$ satisfies the KKT conditions in (4.4), hence $\bar{x}$ is the v-GNE of the game in (4.2). □

Theorem 4.4 (*Convergence of Algorithm 4.3*)**.** Let Assumptions 4.1, 4.2(i), 4.3, 4.4 hold, and let $c > \underline{c}$, with $c > \underline{c}$ as in Algorithm 4.3. Then, for any initial condition in $S = \Omega \times \boldsymbol{E}_{\bar{n}}^\perp \times \boldsymbol{E}_m^\perp \times \mathbb{R}_{\geq 0}^{mN}$ the system in (4.14) has a unique Carathéodory solution, which belongs to S for all $t \geq 0$. The solution converges to an equilibrium $\operatorname{col}(\bar{x}, \bar{\boldsymbol{\varsigma}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}})$, with $\boldsymbol{\psi}(\bar{x}) + \bar{\boldsymbol{\varsigma}} = \mathbf{1}_N \otimes \psi(\bar{x})$, $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$, and $(\bar{x}, \lambda^*)$ satisfies the KKT conditions in (4.4), hence $\bar{x}$ is the v-GNE of the game in (4.2). □

4.6. MULTI-INTEGRATOR AGENTS

In this section, we consider the game in (4.2) under the following additional assumption, which is standard for NE problems with higher-order dynamical agents ([116, Asm. 1]; [49, Def. 1]).

Assumption 4.5. $\Omega = \mathbb{R}^n$. □

Besides, we study problems where each agent is represented by a system of (mixed-order) multi-integrators:

$$\forall i \in \mathcal{I}: \qquad \left\{ x_{i,k}^{(r_{i,k})} = u_{i,k}, \quad k \in \{1, \dots, n_i\}, \right. \tag{4.18}$$

where $r_{i,k} \geq 1$ (with the superscript "$(r_{i,k})$" denoting the $r_{i,k}$-th time derivative), and we denote by $x_{i,k} := [x_i]_k$, $u_{i,k} := [u_i]_k$ the k-th scalar component of agent i strategy and control input, respectively. Our aim is to drive the agents' actions (i.e., the x_i coordinates of each agent state) to a v-GNE of the game in (4.2). We emphasize that the agents are not able to directly control their strategy x_i in (4.18).

Remark 4.4. We consider the general form in (4.18) – instead of homogeneous multi-integrator systems $x_i^{(r_i)} = u_i$ as in [116] – because these dynamics often arise from feedback linearization of multi-input multi-output (nonlinear) systems. As an example, the feedback linearized model of a quadrotor in [95, Eq. 18] is a combination of triple and double integrators. In general, consider any input-affine system

$$\forall i \in \mathcal{I}: \quad \dot{z}_i = \mathfrak{f}_i(z_i) + \mathfrak{g}_i(z_i)\bar{u}_i, \quad x_i = \mathfrak{h}_i(z_i), \tag{4.19}$$

for smooth mappings $\mathfrak{f}_i : \mathbb{R}^{q_i} \to \mathbb{R}^{q_i}$, $\mathfrak{g}_i : \mathbb{R}^{q_i} \to \mathbb{R}^{q_i \times n_i}$, $\mathfrak{h} : \mathbb{R}^{q_i} \to \mathbb{R}^{n_i}$; the objective is to drive the controlled outputs x_i to a v-GNE. Assume that the systems in (4.19) have, for all $z_i \in \mathbb{R}^{q_i}$, vector relative degree [76, §5.1] $\{r_{i,1}, \dots, r_{i,n_i}\}$, with $r_{i,1}, \dots, r_{i,n_i} \geq 1$ and $r_1 + \cdots + r_{n_i} = q_i$. This class includes, e.g., the Euler–Lagrangian dynamics considered in [49]. Then, for all $i \in \mathcal{I}$, there is a change of coordinates $\xi_i = T_i(z_i)$ and a state feedback $\bar{u}_i = \alpha(\xi_i) + \beta(\xi_i)u_i$ such that the closed-loop system, in the new coordinates and with transformed input u_i, is exactly (4.18) [76, §5.2]. In practice, the problem of driving the systems in (4.19) to a v-GNE can be recast, via a linearizing feedback, as that of controlling the multi-integrator agents in (4.18) to a v-GNE. □

Let $\mathcal{K}_i := \{1, \dots, n_i\}$ and $\mathcal{M}_i := \{k \in \mathcal{K}_i \mid r_{i,k} > 1\}$, for all $i \in \mathcal{I}$. We assume that each agent is able to measure its full state. Similarly to [116], in (4.18), for each $i \in \mathcal{I}$, we consider the controllers

$$\forall k \in \mathcal{K}_i: \quad u_{i,k} = \tilde{u}_{i,k} - \textstyle\sum_{j=1}^{r_{i,k}-1} c_{i,k,j-1} x_{i,k}^{(j)}, \tag{4.20}$$

where $\tilde{u}_{i,k}$ is a translated input to be chosen, and $(c_{i,k,0} := 1, \dots, c_{i,k,r_{i,k}-2}, c_{i,k,r_{i,k}-1})$ are the ascending coefficients of any Hurwitz polynomial of order $(r_{i,k} - 1)$, for all $i \in \mathcal{K}_i$. Moreover, for all $i \in \mathcal{I}$, we define the coordinate transformation

$$\operatorname{col}\left(\left(\operatorname{col}\left((x_{i,k}, \dots, x_{i,k}^{(r_{i,k}-1)}) \right) \right)_{k \in \mathcal{K}_i} \right) \to \operatorname{col}(\zeta_i, v_i), \tag{4.21}$$

where $v_i := \operatorname{col}((v_{i,k})_{k \in \mathcal{M}_i})$ and $\zeta_i := \operatorname{col}((\zeta_{i,k})_{k \in \mathcal{K}_i})$, with $v_{i,k} := \operatorname{col}(x_{i,k}^{(1)}, \dots, x_{i,k}^{(r_{i,k}-1)})$, and

$$\zeta_{i,k} := \begin{cases} x_{i,k} + \sum_{j=1}^{r_{i,k}-1} c_{i,k,j} x_{i,k}^{(j)} & \text{if } k \in \mathcal{M}_i \\ x_{i,k} & \text{if } k \notin \mathcal{M}_i. \end{cases} \tag{4.22}$$

Algorithm 4.5. Multi-integrator agents (adaptive gains)

Initialization: $\forall i \in \mathcal{I}$, set $\gamma_i > 0$, $\boldsymbol{\zeta} i,-i(0) \in \mathbb{R}^{n-n_i}$, $k_i(0) \in \mathbb{R}$, $z_i(0) = \mathbf{0}_m$, $\lambda_i(0) \in \mathbb{R}^m$;

Dynamics: for all $i \in \mathcal{I}$, for all $k \in \mathcal{K}_i$,

$$
\begin{aligned}
x_{i,k}^{(r_{i,k})} &= u_{i,k} = \tilde{u}_{i,k} - \textstyle\sum_{j=1}^{r_{i,k}-1} c_{i,k,j-1} x_{i,k}^{(j)} \\
\tilde{u}_i &= -\nabla_{x_i} J_i(\boldsymbol{\zeta}_{i,i}, \boldsymbol{\zeta}_{i,-i}) - \tfrac{\partial}{\partial x_i} g_i(\boldsymbol{\zeta}_{i,i})^\top \lambda_i - \textstyle\sum_{j \in \mathcal{N}_i} (k_i \rho_{i,i} - k_j \rho_{j,i}) \\
\dot{\boldsymbol{\zeta}}_{i,-i} &= -\textstyle\sum_{j \in \mathcal{N}_i} (k_i \rho_{i,-i} - k_j \rho_{j,-i}) \\
\boldsymbol{\zeta}_{i,i} &= \zeta_i \\
\dot{k}_i &= \gamma_i \|\rho_i\|^2 \qquad\qquad \rho_i = \textstyle\sum_{j \in \mathcal{N}_i} (\boldsymbol{\zeta}_j - \boldsymbol{\zeta}_i) \\
\dot{z}_i &= \textstyle\sum_{j \in \mathcal{N}_i} (\lambda_i - \lambda_j) \\
\dot{\lambda}_i &= \Pi_{\mathbb{R}^m_{\geq 0}}(\lambda_i, g_i(\boldsymbol{\zeta}_{i,i}) - z_i - \textstyle\sum_{j \in \mathcal{N}_i} (\lambda_i - \lambda_j))
\end{aligned}
$$

We note that, for the closed loop systems in the new coordinates, it holds, for all $i \in i \in \mathcal{I}$,

$$
\forall k \in \mathcal{M}_i : \quad \begin{cases} \dot{\zeta}_{i,k} = \tilde{u}_{i,k} & \text{(4.23a)} \\ \dot{v}_{i,k} = E_{i,k} v_i + G_{i,k} \tilde{u}_{i,k}, & \text{(4.23b)} \end{cases}
$$

where

$$
E_{i,k} = \begin{bmatrix} \mathbf{0}_{r_i-2} & I_{r_i-2} \\ 1 & -c_{i,k}^\top \end{bmatrix}, \qquad G_{i,k} = \begin{bmatrix} \mathbf{0}_{r_i-2} \\ 1 \end{bmatrix},
$$

and $c_{i,k} := \operatorname{col}(c_{i,k,1}, \dots, c_{i,k,r_{i,k}-2})$.

We conclude that the system in (4.18), with the control inputs (4.20), in the new coordinates (4.21), reads as

$$
\forall i \in \mathcal{I} : \quad \begin{cases} \dot{\zeta}_i = \tilde{u}_i & \text{(4.24a)} \\ \dot{v}_i = E_i v_i + G_i \tilde{u}_i, & \text{(4.24b)} \end{cases}
$$

where $\tilde{u}_i := \operatorname{col}((\tilde{u}_{i,k})_{k \in \mathcal{K}_i})$, $E_i := \operatorname{diag}((E_{i,k})_{k \in \mathcal{M}_i})$, $G_i := \operatorname{diag}((G_{i,k})_{k \in \mathcal{M}_i})$, for all $i \in \mathcal{I}$.

The dynamics of ζ_i in (4.24a) are identical to the single-integrator in (4.6), with translated input $\tilde{u}_i$. As such, we are in a position to design $\tilde{u}_i$ according to Algorithm 4.2 (or 4.1, or 4.3 or 4.4 for aggregative games), to drive the variable $\zeta := \operatorname{col}((\zeta_i)_{i \in \mathcal{I}})$ to an equilibrium $\bar{\zeta} = x^*$, where x^* is the v-GNE for the game in (4.2). In the following, we show that this choice is sufficient to also control the original variables x_i to the v-GNE.

The resulting dynamics are shown in Algorithm 4.5. Here, $\boldsymbol{\zeta}_i := (\operatorname{col}(\boldsymbol{\zeta}_{i,j})_{j \in \mathcal{I}})$, and $\boldsymbol{\zeta}_{i,j}$ represents agent i's estimation of the quantity ζ_j, for $j \neq i$, while $\boldsymbol{\zeta}_{i,i} := \zeta_i$, $\boldsymbol{\zeta}_{i,-i} := \operatorname{col}((\boldsymbol{\zeta}_{i,j})_{j \in \mathcal{I} \setminus \{i\}})$. Let also $\boldsymbol{\zeta} := \operatorname{col}((\boldsymbol{\zeta}_i)_{i \in \mathcal{I}})$.

Theorem 4.5 (Convergence of Algorithm 4.5)**.** Let Assumptions 4.1, 4.2, 4.3, 4.5 hold. For any initial condition, the system in Algorithm 4.5 has a unique Carathéodory solution. The solution converges to an equilibrium $\operatorname{col}(\bar{x}, \bar{\boldsymbol{\zeta}}, \bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}})$, with $\bar{x} = x^*$, $\bar{\boldsymbol{\zeta}} = \mathbf{1}_N \otimes x^*$, $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$, and (x^*, λ^*) satisfies the KKT conditions in (4.4), hence x^* is the v-GNE of the game in (4.2). □

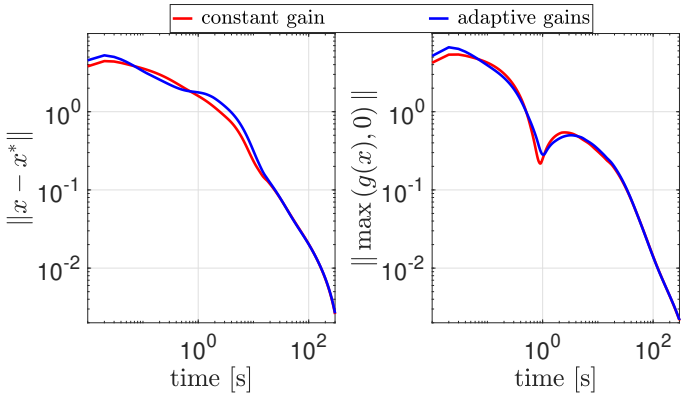

Figure 4.1: Results of Algorithms 4.1-4.2 for velocity-actuated vehicles.

We emphasize that the proof of Theorem 4.5 is not based on the specific structure of Algorithm 4.2; in fact, the result still holds if another secondary controller with analogous convergence properties is employed to design $\tilde{u}_i$ in (4.24). For instance, by choosing the controller in [50, Eq. 11], we can address aggregative games played by multi-integrator agents over balanced digraphs. [116] follow a similar approach (for homogeneous multi-integrators and NE problems), and handle the presence of deterministic disturbances by leveraging the ISS properties of their selected secondary controller [63, Eq. 47]. We have not guaranteed this robustness for our dynamics. However, the algorithm in [116] is designed for unconstrained games. On the contrary, Algorithm 4.5 drives the system in (4.18) to the v-GNE of a generalized game, and ensures asymptotic satisfaction of the coupling constraints. We finally remark that we assumed the absence of local constraints (Assumption 4.5); nevertheless, if some are present, they can be dualized and satisfied asymptotically, as in Remark 4.2.

4.7. ILLUSTRATIVE NUMERICAL EXAMPLES

4.7.1. OPTIMAL POSITIONING IN MOBILE SENSOR NETWORKS

We consider a connectivity problem formulated as a game, as in [130]. A group $\mathcal{I} = \{1, \dots, N = 5\}$ of mobile sensor devices have to coordinate their actions via wireless communication, to perform some task, e.g., exploration or surveillance. Mathematically, each sensor i aims at autonomously finding the position $x_i = \mathrm{col}(p_i^x, p_i^y)$ in a plane to optimize some private primary objective $c_i(x_i)$, but not rolling away too much from the other devices. This goal is represented by the following cost functions, for all $i \in \mathcal{I}$:

$$J_i(x_i, x_{-i}) := c_i(x_i) + \textstyle\sum_{j \in \mathcal{I}} \left\| x_i - x_j \right\|^2.$$

Here, $c_i(x_i) := x_i^T x_i + d_i^\top x_i + \sin(p_i^x)$, with $d_i \in \mathbb{R}^2$ randomly generated local parameters, for each $i \in \mathcal{I}$. The useful space is restricted by the local constraints $0.1 \le p_i^y \le 0.5$, $\forall i \in \mathcal{I}$. The sensors communicate over a random undirected connected graph $\mathcal{G}(\mathcal{I}, \mathcal{E})$. To preserve connectivity, the Chebyschev distance between any two neighboring agents

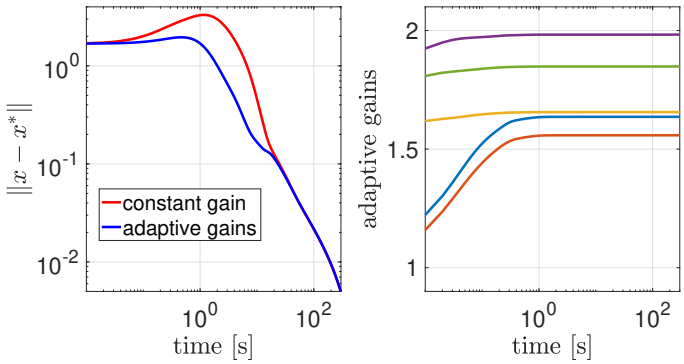

Figure 4.2: Results of Algorithm 4.5 for Euler–Lagrangian vehicles linearized via feedback linearization.

has to be smaller than $\frac{1}{5}$, resulting in the coupling constraints $\max\{|p_i^x - p_j^x|, |p_i^y - p_j^y|\} \leq \frac{1}{5}, \forall (i,j) \in \mathcal{E}$. After the deployment, all the sensors start sending the data they collect to a base station, located at $\bar{x} = \text{col}(0, 0.3)$, via wireless communication. To maintain acceptable levels of transmission power consumption, the average steady state distance from the base is limited as $\frac{1}{N}\sum_{j\in\mathcal{N}_i}(x_i - \bar{x})^\top (x_i - \bar{x}) \leq \frac{1}{2}$. This setup satisfies Assumptions 4.1-4.2. We set $c = 30$ to satisfy the condition in Theorem 4.2; $\gamma_i = 1, \forall i \in \mathcal{I}$; initial conditions are chosen randomly. We consider two different cases for the sensor physical dynamics.

Velocity-actuated vehicles: Each agent is a single-integrator as in (4.6). Figure 4.1 compares the results for Algorithms 4.1 and 4.2 (in logarithmic scale) and shows convergence of both to the unique v-GNE and asymptotic satisfaction of the coupling constraints. In the first phase, the controllers are mostly driven by the consensual dynamics; we remark that, when the agents agree on their estimates, the two algorithms coincide.

Euler–Lagrangian vehicles: Each mobile sensor $i \in \mathcal{I}$ is modeled as an Euler–Lagrangian systems of the form $I_i(x_i)\ddot{x}_i + C_i(x_i, \dot{x}_i) + U_i(x_i) = u_i$, where $U_i = \text{col}(0, -1)$,

$$I_i(x_i) = \begin{bmatrix} 2 + 0.6 * \cos(p_i^y) & 0.5 + 0.3\cos(p_i^y) \\ 0.5 + 0.3\cos(p_i^y) & 0.5 \end{bmatrix},$$
$$C_i(x_i, \dot{x}_i) = \begin{bmatrix} -0.3\sin(p_i^y)\dot{p}_i^y & -0.3\sin(p_i^y)(\dot{p}_i^x + \dot{p}_i^y) \\ 0.3\sin(p_i^y)\dot{p}_i^x & 0 \end{bmatrix}.$$

The systems satisfy the conditions in Remark 4.4 with uniform vector relative degree $\{2,2\}$. Therefore, we first apply a linearizing feedback; the problem then reduces to the control of double-integrator agents, and we choose the transformed input (see Remark 4.4) according to Algorithm 4.5 and the analogous algorithm with constant gain (obtained by choosing $\tilde{u}_i$ in (4.24a) according to Algorithm 4.1). The local constraints are dualized as in Remark 4.2. The results are illustrated in Figure 4.2. Finally, in Figure 4.3, we compare the trajectories of the vehicles in the velocity-actuated and Euler–Lagrangian cases. Importantly, the local constraints are satisfied along the whole trajectory for single-integrator agents, only asymptotically for the higher-order agents.

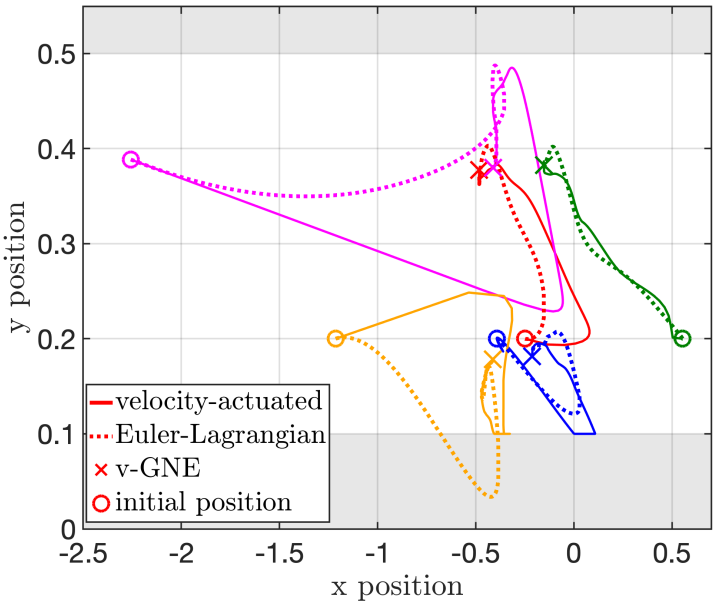

Figure 4.3: Cartesian trajectories of velocity-actuated and Euler–Lagrangian vehicles, with adaptive gains.

4.7.2. COMPETITION IN POWER MARKETS AS AGGREGATIVE GAME

We consider a Cournot competition model [73], [109]. A group $\mathcal{I} = \{1,\dots,N\}$ of firms produces energy for a set of markets $\mathcal{J} = \{1,\dots,m\}$, each corresponding to a different location. Each firm $i \in \mathcal{I}$ controls a production plant in $n_i \leq m$ of the locations, and decides on the power outputs $x_i \in \mathbb{R}^{n_i}$ of its generators. Power is only dispatched in the location of production. Each plant has a maximal capacity, described by the local constraints $\mathbf{0}_{n_i} \leq x_i \leq X_i$. Moreover, an *independent system operator* (ISO) imposes an upper bound on the market share of the producers, so that $\mathbf{1}_{n_i}^\top x_i \leq C_i$. Market clearance is guaranteed by the ISO via external control mechanisms, but the overall power generation is bounded by markets capacities $r = \mathrm{col}((r_j)_{j\in\mathcal{J}})$. Thus, the firms share the constraints $Ax \leq r$. Here, $A = [A_1 \dots A_N]$, and $A_i \in \mathbb{R}^{m\times n_i}$ with $[A_i]_{j,k} = 1$ if $[x_i]_k$ is the power generated in location $j \in \mathcal{J}$ by agent i, $[A_i]_{j,k} = 0$ otherwise, for all $j \in \mathcal{J}$, $k = 1,\dots,n_i$. In simple terms, $Ax \in \mathbb{R}^m$ is the vector of total power generations for each market. Each firm $i \in \mathcal{I}$ aims at maximizing its profit, i.e., minimizing the cost

$$J_i(x_i, x_{-i}) = c_i(x_i) - p(Ax)^\top A_i x_i + w(\mathbf{1}_{n_i}^\top x_i),$$

where $c_i(x_i) = \sum_{k=1}^{n_i} Q_{i,k}([x_i]_k)^2 + q_{i,k}[x_i]_k$ is the generation cost, $p(Ax)^\top A_i x_i$ is the revenue, where $p : \mathbb{R}^m \to \mathbb{R}^m$ associates to each market a unitary price depending on the offer and $[p(Ax)]_j = P_j - \chi_j[Ax]_j$, $w(y) = w_2 y - w_1 y^2$ is a price charged by the ISO for the use of the infrastructure. We set $N = 20$, $m = 7$ and randomly select which firms participate in each market. We choose with uniform distribution X_i in $[0.3, 1.3]$, C_i in $[1,2]$, r_j in $[1,2]$, $Q_{i,k}$ in $[8,16]$, q_i in $[1,2]$, P_j in $[10,20]$, χ_j in $[1,3]$, for all $i \in \mathcal{I}$, $j \in \mathcal{J}$,

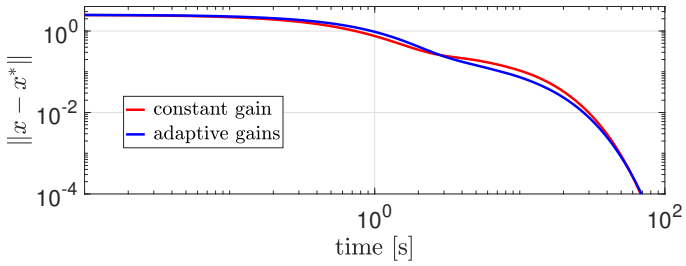

Figure 4.4: Distance from the v-GNE, for the power production in the electricity market.

$k = 1,\dots,n_i$, w_1 in $[0.5,1]$, w_2 in $[0,0.1]$. The firms cannot access the productions of all the competitors, but can communicate with some neighbors on a connected graph. The turbine of generator i is governed by the dynamics [49]

$$\dot{P}_{i,k} = -\alpha^1_{i,k} P_{i,k} + \alpha^2_{i,k} R_{i,k}$$
$$\dot{R}_{i,k} = -\alpha^3_{i,k} R_{i,k} + \alpha^4_{i,k} u_{i,k},$$

with $P_{i,k} = [x_i]_k$; $R_{i,k}$ and $u_{i,k}$ are the steam valve opening and control input; the parameters $\alpha^{\cdot}_{i,k}$'s are set as in [49]. Via feedback linearization, the problem for each generator reduces to the control of a double-integrator. The competition among the firms is described as an aggregative game with aggregation value $\psi(x) = Ax$ (this is advantageous with respect to the formulation in [109], as the firms only keep an estimate of the aggregation and firm i does not need to know the quantities A_j, $j \neq i$). We numerically check that this setup satisfies Assumptions 4.2, 4.4. We simulate the equivalent of Algorithm 4.5 for aggregative games, obtained by choosing $\tilde{u}_i$ in (4.24) according to Algorithms 4.3, 4.4; we deal with the local constraints as in Remark 4.2. The results are shown in Figure 4.4 and indicate fast convergence of the firms' production to the unique v-GNE.

4.8. CONCLUSION AND OUTLOOK

GENERALIZED games played by nonlinear systems with maximal relative degree can be solved via continuous-time fully-distributed primal-dual pseudogradient controllers, provided that the game mapping is strongly monotone and Lipschitz continuous. Convergence can be ensured even without a-priori knowledge on the game parameters, via integral consensus. Seeking an equilibrium when the agents are characterized by constrained dynamics is currently an open problem. The extension of our results to the case of direct communication, noise and parameter uncertainties is left as future work.

4.9. APPENDIX

4.9.1. PROOF OF LEMMA 4.2

Under Assumption 4.3, we have, for any $q > 0$,

$$\text{Range}\left(\boldsymbol{L}_q\right) = \text{Null}\left(\mathbf{1}_N^\top \otimes I_q\right) \quad = \boldsymbol{E}_q^\perp, \tag{4.25}$$

$$\text{Null}\left(\boldsymbol{L}_q\right) = \text{Range}\left(\mathbf{1}_N \otimes I_q\right) = \boldsymbol{E}_q. \tag{4.26}$$

i) For any equilibrium $\text{col}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}})$ of (4.10), with $\bar{\boldsymbol{z}} \in \boldsymbol{E}_m^\perp$, it holds that

$$\mathbf{0} \in \mathcal{R}^\top(\boldsymbol{F}(\bar{\boldsymbol{x}}) + \mathbb{G}(\mathcal{R}\bar{\boldsymbol{x}})^\top \bar{\boldsymbol{\lambda}} + \mathrm{N}_\Omega(\mathcal{R}\bar{\boldsymbol{x}})) + \boldsymbol{L}_n \bar{K} \bar{\boldsymbol{\rho}} \tag{4.27}$$

$$\mathbf{0} = D\left(\bar{\boldsymbol{\rho}}\right)^\top (\Gamma \otimes I_n)\, \bar{\boldsymbol{\rho}}, \quad \bar{\boldsymbol{\rho}} = \boldsymbol{L}_n \bar{\boldsymbol{x}} \tag{4.28}$$

$$\mathbf{0} = \boldsymbol{L}_m \bar{\boldsymbol{\lambda}} \tag{4.29}$$

$$\mathbf{0} \in -\boldsymbol{g}(\mathcal{R}\bar{\boldsymbol{x}}) + \bar{\boldsymbol{z}} + \boldsymbol{L}_m \bar{\boldsymbol{\lambda}} + \mathrm{N}_{\mathbb{R}^{Nm}_{\geq 0}}\left(\bar{\boldsymbol{\lambda}}\right), \tag{4.30}$$

where $\bar{K} = \text{diag}((\bar{k}_i I_n)_{i\in\mathcal{I}})$. By (4.28) we have $\bar{\boldsymbol{\rho}} = \mathbf{0}_{Nn}$, i.e., $\bar{\boldsymbol{x}} \in \boldsymbol{E}_n$ by (4.26), and by (4.29) and (4.26), we have $\bar{\boldsymbol{\lambda}} \in \boldsymbol{E}_m$. Hence, $\bar{\boldsymbol{x}} = \mathbf{1}_N \otimes x^*$ and $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$, for some $x^* \in \mathbb{R}^n$, $\lambda^* \in \mathbb{R}^m$. By left multiplying both sides of (4.27) by $(\mathbf{1}_N^\top \otimes I_n)$, by (4.26) and since $(\mathbf{1}_N^\top \otimes I_n)\mathcal{R}^\top = I_n$, $\boldsymbol{F}(\mathbf{1}_N \otimes x^*) = F(x^*)$, $\mathcal{R}\bar{\boldsymbol{x}} = x^*$, and $\mathbb{G}(\mathcal{R}\bar{\boldsymbol{x}})^\top(\mathbf{1}_N \otimes \lambda^*) = \frac{\partial}{\partial x} g(x^*)^\top \lambda^*$, we retrieve the first KKT condition in (4.4). We obtain the second condition in (4.4) by left multiplying both sides of (4.30) by $(\mathbf{1}_N^\top \otimes I_m)$ and by using that $(\mathbf{1}_N^\top \otimes I_m)\boldsymbol{g}(\mathcal{R}\bar{\boldsymbol{x}}) = g(x^*)$, $(\mathbf{1}_N^\top \otimes I_m)\bar{\boldsymbol{z}} = 0$ and $(\mathbf{1}_N^\top \otimes I_m)\mathrm{N}_{\mathbb{R}^{Nm}_{\geq 0}}(\mathbf{1}_N \otimes \lambda^*) = N\mathrm{N}_{\mathbb{R}^m_{\geq 0}}(\lambda^*) = \mathrm{N}_{\mathbb{R}^m_{\geq 0}}(\lambda^*)$.

ii) Let (x^*, λ^*) be any pair that satisfies the KKT conditions in (4.4). By taking $\bar{\boldsymbol{x}} = \mathbf{1}_N \otimes x^*$, $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$ and any $\bar{\boldsymbol{k}}$, (4.27)-(4.29) are satisfied as above. It suffices to show that there exists $\bar{\boldsymbol{z}} \in \boldsymbol{E}_m^\perp$ such that (4.30) holds, i.e., that $(-\boldsymbol{g}(\mathcal{R}\bar{\boldsymbol{x}}) + \bar{v}) \in \boldsymbol{E}_m^\perp$, for some $\bar{v} \in \mathrm{N}_{\mathbb{R}^{Nm}_{\geq 0}}(\bar{\boldsymbol{\lambda}})$. By (4.4), there exists $v^* \in \mathrm{N}_{\mathbb{R}^m_{\geq 0}}(\lambda^*)$ such that $g(x^*) = v^*$. Since $\mathrm{N}_{\mathbb{R}^{Nm}_{\geq 0}}(\mathbf{1}_N \otimes \lambda^*) = \times_{i\in\mathcal{I}} \mathrm{N}_{\mathbb{R}^m_{\geq 0}}(\lambda^*)$, it follows by properties of cones that $\text{col}(v_1^*, \dots, v_N^*) \in \mathrm{N}_{\mathbb{R}^{Nm}_{\geq 0}}(\bar{\boldsymbol{\lambda}})$ with $v_1^* = \cdots = v_N^* = \frac{1}{N} v^*$. Therefore, $(\mathbf{1}_N^\top \otimes I_m)(-\boldsymbol{g}(\mathcal{R}\bar{\boldsymbol{x}}) + \bar{v}) = -g(x^*) + v^* = \mathbf{0}_m$, or $(-\boldsymbol{g}(\mathcal{R}\bar{\boldsymbol{x}}) + \bar{v}) \in \boldsymbol{E}_m^\perp$. ■

4.9.2. PROOF OF LEMMA 4.4

Let $\boldsymbol{y} = \mathbf{1}_n \otimes y$, for some $y \in \mathbb{R}^n$. We decompose $\boldsymbol{x} = \boldsymbol{x}^\perp + \boldsymbol{x}^\parallel$, where $\boldsymbol{x}^\parallel := P_n \boldsymbol{x}$, $\boldsymbol{x}^\perp := P_n^\perp \boldsymbol{x}$. Therefore, $\boldsymbol{x}^\parallel = \mathbf{1}_N \otimes \hat{x}$, for some $\hat{x} \in \mathbb{R}^n$. By [109, Eq. 50],

$$\begin{aligned}&(\boldsymbol{x} - \boldsymbol{y})^\top \mathcal{R}^\top (\boldsymbol{F}(\boldsymbol{x}) - \boldsymbol{F}(\boldsymbol{y})) \\ &\quad \geq -\theta \|\hat{x} - y\| \|\boldsymbol{x}^\perp\| + \mu \|\hat{x} - y\|^2 - \theta \|\boldsymbol{x}^\perp\|^2 - \theta_0 \|\boldsymbol{x}^\perp\| \|\hat{x} - y\|.\end{aligned}$$

For any $k^* > \underline{k} > 0$, we have $K^* > 0$ and, by (4.26), $\text{Null}\,(\boldsymbol{L}_n K^* \boldsymbol{L}_n) = \boldsymbol{E}_n$. Therefore it holds that $(\boldsymbol{x}-\boldsymbol{y})^\top \boldsymbol{L}_n K^* \boldsymbol{L}_n (\boldsymbol{x}-\boldsymbol{y}) \geq k^* \lambda_2(L)^2 \|\boldsymbol{x}^\perp\|^2$. By $\left\|\hat{x} - y\right\| = \frac{1}{\sqrt{N}} \left\|\boldsymbol{x}^\parallel - \boldsymbol{y}\right\|$, we conclude that

$$\begin{aligned}&\left(\boldsymbol{x} - \boldsymbol{y}\right)^\top \left(\mathcal{R}^\top \left(\boldsymbol{F}(\boldsymbol{x}) - \boldsymbol{F}\left(\boldsymbol{y}\right)\right) + \boldsymbol{L}_n K^* \boldsymbol{L}_n \left(\boldsymbol{x} - \boldsymbol{y}\right)\right) \\ &\geq \text{col}(\|\boldsymbol{x}^\perp\|, \|\boldsymbol{x}^\parallel - \boldsymbol{y}\|)^\top M_1 \text{col}(\|\boldsymbol{x}^\perp\|, \|\boldsymbol{x}^\parallel - \boldsymbol{y}\|),\end{aligned}$$

with M_1 as in (4.11) and, for $k^* > \underline{k}$, $M_1 > 0$ by Silvester's criterion. The conclusion follows since, by orthogonality, $\|\boldsymbol{x}^\parallel - \boldsymbol{y}\|^2 + \|\boldsymbol{x}^\perp\|^2 = \|\boldsymbol{x} - \boldsymbol{y}\|^2$. ■

4.9.3. PROOF OF THEOREM 4.1

We first rewrite the dynamics in (4.10) as

$$\dot{\boldsymbol{\omega}} = \Pi_{\Xi}(\boldsymbol{\omega}, -\mathcal{A}(\boldsymbol{\omega}) - \mathcal{B}(\boldsymbol{\omega})), \tag{4.31}$$

where $\boldsymbol{\omega} := \mathrm{col}\,(\boldsymbol{x}, \boldsymbol{k}, \boldsymbol{z}, \boldsymbol{\lambda})$, $\Xi := \boldsymbol{\Omega} \times \mathbb{R}^N \times \mathbb{R}^{Nm} \times \mathbb{R}^{mN}_{\geq 0}$,

$$\mathcal{A}(\boldsymbol{\omega}) := \begin{bmatrix} \mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{L}_n K \boldsymbol{L}_n \boldsymbol{x} \\ -D(\boldsymbol{\rho})^\top (\Gamma \otimes I_n)\, \boldsymbol{\rho} \\ \mathbf{0}_{Nm} \\ \boldsymbol{L}_m \boldsymbol{\lambda} \end{bmatrix}, \; \mathcal{B}(\boldsymbol{\omega}) := \begin{bmatrix} \mathcal{R}^\top \mathbb{G}(\mathcal{R}\boldsymbol{x})^\top \boldsymbol{\lambda} \\ \mathbf{0}_N \\ -\boldsymbol{L}_m \boldsymbol{\lambda} \\ -\boldsymbol{g}(\mathcal{R}\boldsymbol{x}) + \boldsymbol{z} \end{bmatrix}.$$

By Assumption 4.1 and Lemma 4.3, $\mathcal{A}$ and $\mathcal{B}$ are locally Lipschitz; therefore, for any initial condition in Ξ, the system (4.31) has a unique *local* Carathéodory solution, contained in Ξ [40]. Moreover, we note that the set $S = \{\boldsymbol{w} \in \Xi \mid \boldsymbol{z} \in \boldsymbol{E}_m^{\perp}\}$ is invariant for the system (4.31), since for all $\boldsymbol{\omega} \in \Xi$, $\frac{\partial}{\partial \boldsymbol{\omega}}((\mathbf{1}_N \otimes I_m)^\top \boldsymbol{z})\dot{\boldsymbol{\omega}} = (\mathbf{1}_N^\top \otimes I_m)\boldsymbol{L}_m \boldsymbol{\lambda} = \mathbf{0}_m$.

Let $\Phi := P_m + \boldsymbol{L}_m^+$, where $\boldsymbol{L}_m^+$ is the Moore-Penrose pseudo-inverse of $\boldsymbol{L}_m$, and we recall that $P_m = \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top \otimes I_m$ is the projection matrix on $\boldsymbol{E}_m$. By properties of the pseudo-inverse and (4.26), $\boldsymbol{L}_m^+ = \boldsymbol{L}_m^{+\top}$, $\boldsymbol{L}_m^+ \succeq 0$ and $\mathrm{Null}(\boldsymbol{L}_m^+) = \boldsymbol{E}_m$. Since $\mathrm{Null}(P_m) = \boldsymbol{E}_m^{\perp}$ and $P_m \succeq 0$, we have $\Phi \succ 0$. Also, $\boldsymbol{L}_m^+ \boldsymbol{L}_m = I_{Nm} - P_m = P_m^{\perp}$ is the projector matrix on $\mathrm{Range}(\boldsymbol{L}_m) = \boldsymbol{E}_m^{\perp}$. We define the quadratic Lyapunov function

$$\begin{aligned} V &= \tfrac{1}{2}\|\boldsymbol{\omega} - \bar{\boldsymbol{\omega}}\|_Q^2 := (\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})^\top Q (\boldsymbol{\omega} - \bar{\boldsymbol{\omega}}) \\ &= \tfrac{1}{2}(\|\boldsymbol{x} - \bar{\boldsymbol{x}}\|^2 + \|\boldsymbol{k} - \bar{\boldsymbol{k}}\|_{\Gamma^{-1}}^2 + \|\boldsymbol{z} - \bar{\boldsymbol{z}}\|_{\Phi}^2 + \|\boldsymbol{\lambda} - \bar{\boldsymbol{\lambda}}\|^2), \end{aligned}$$

where $Q =: \mathrm{diag}(I_{Nn}, \Gamma^{-1}, \Phi, I_{Nm})$, and $\bar{\boldsymbol{x}} = \mathbf{1}_N \otimes x^*$, $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$, where the pair (x^*, λ^*) satisfies the KKT conditions in (4.4), $\bar{\boldsymbol{k}}$ such that $k^* := \min(\bar{\boldsymbol{k}}) \geq \underline{k}$, with $\underline{k}$ as in (4.11), $\bar{\boldsymbol{z}} \in \boldsymbol{E}_m^{\perp}$ chosen such that $\bar{\boldsymbol{\omega}} := \mathrm{col}\,(\bar{\boldsymbol{x}}, \bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}})$ is an equilibrium for (4.10), and such a $\bar{\boldsymbol{z}}$ exists by the proof of Lemma 4.2. Therefore, for any $\boldsymbol{\omega} \in S$, we have

$$\begin{aligned} \dot{V}(\boldsymbol{\omega}) :&= \nabla V(\boldsymbol{\omega})\dot{\boldsymbol{\omega}} = (\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})^\top Q \dot{\boldsymbol{\omega}} = \\ &= (\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})^\top Q \Pi_{\Xi}(\boldsymbol{\omega}, -\mathcal{A}(\boldsymbol{\omega}) - \mathcal{B}(\boldsymbol{\omega})) \\ &\leq (\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})^\top Q (-\mathcal{A}(\boldsymbol{\omega}) - \mathcal{B}(\boldsymbol{\omega})), \end{aligned} \tag{4.32}$$

where the last inequality follows from Lemma 4.1 and by exploiting the structure of Q and Ξ. By Lemma 4.1, it also holds that $(\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})^\top Q(-\mathcal{A}(\bar{\boldsymbol{\omega}}) - \mathcal{B}(\bar{\boldsymbol{\omega}})) \leq 0$. By subtracting this term from (4.32), we obtain

$$\dot{V}(\boldsymbol{\omega}) \leq -(\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})^\top Q (\mathcal{A}(\boldsymbol{\omega}) - \mathcal{A}(\bar{\boldsymbol{\omega}}) + \mathcal{B}(\boldsymbol{\omega}) - \mathcal{B}(\bar{\boldsymbol{\omega}})).$$

Besides, for any $\boldsymbol{z} \in \boldsymbol{E}_m^{\perp}$, by $\boldsymbol{L}_m^+ \boldsymbol{L}_m = P_m^{\perp}$ and (4.26), we have $(\boldsymbol{z} - \bar{\boldsymbol{z}})^\top \Phi \boldsymbol{L}_m (\boldsymbol{\lambda} - \bar{\boldsymbol{\lambda}}) = (\boldsymbol{z} - \bar{\boldsymbol{z}})^\top (\boldsymbol{\lambda} - \bar{\boldsymbol{\lambda}})$, and hence

$$\begin{aligned} &(\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})^\top Q (\mathcal{B}(\boldsymbol{\omega}) - \mathcal{B}(\bar{\boldsymbol{\omega}})) \\ &= (\boldsymbol{x} - \bar{\boldsymbol{x}})^\top \mathcal{R}^\top (\mathbb{G}(\mathcal{R}\boldsymbol{x})^\top \boldsymbol{\lambda} - \mathbb{G}(\mathcal{R}\bar{\boldsymbol{x}})^\top \bar{\boldsymbol{\lambda}}) \\ &\quad + (\boldsymbol{\lambda} - \bar{\boldsymbol{\lambda}})^\top (-\boldsymbol{g}(\mathcal{R}\boldsymbol{x}) + \boldsymbol{g}(\mathcal{R}\bar{\boldsymbol{x}})) \\ &= (x - x^*)^\top (\nabla_y (\boldsymbol{g}(y)^\top \boldsymbol{\lambda})|_{y=x} - \nabla_y (\boldsymbol{g}(y)^\top \bar{\boldsymbol{\lambda}})|_{y=x^*}) \\ &\quad - (\boldsymbol{\lambda} - \bar{\boldsymbol{\lambda}})^\top (\nabla_y (\boldsymbol{g}(x)^\top y)|_{y=\boldsymbol{\lambda}} - \nabla_y (\boldsymbol{g}(x^*)^\top y)|_{y=\bar{\lambda}}) \geq 0 \end{aligned}$$

and the last inequality holds, for any $\boldsymbol{\omega} \in S$, by applying [115, Th. 1] (since $\boldsymbol{\lambda}, \bar{\boldsymbol{\lambda}} \in \mathbb{R}^{Nm}_{\geq 0}$ and by Assumption 4.1). Therefore, for any $\boldsymbol{\omega} \in S$, it holds that:

$$\begin{aligned}\dot{V}(\boldsymbol{\omega}) &\leq -(\boldsymbol{\omega}-\bar{\boldsymbol{\omega}})^\top Q\left(\mathcal{A}(\boldsymbol{\omega})-\mathcal{A}(\bar{\boldsymbol{\omega}})\right)\\ &= -(\boldsymbol{x}-\bar{\boldsymbol{x}})^\top \mathcal{R}^\top\left(\boldsymbol{F}(\boldsymbol{x})-\boldsymbol{F}(\bar{\boldsymbol{x}})\right)\\ &\quad -(\boldsymbol{x}-\bar{\boldsymbol{x}})^\top\left(\boldsymbol{L}_n K \boldsymbol{L}_n(\boldsymbol{x}-\bar{\boldsymbol{x}})\right)\\ &\quad +(\boldsymbol{k}-\bar{\boldsymbol{k}})^\top \Gamma^{-1} D(\boldsymbol{\rho})^\top\left(\Gamma \otimes I_n\right)\boldsymbol{\rho}\\ &\quad -(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^\top \boldsymbol{L}_m(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}}),\end{aligned} \tag{4.33}$$

where we used that $\bar{\boldsymbol{\rho}} := \boldsymbol{L}_n\bar{\boldsymbol{x}} = 0$. For the last addend in (4.33), we can write $(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^\top \boldsymbol{L}_m(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}}) = \boldsymbol{\lambda}^\top \boldsymbol{L}_m \boldsymbol{\lambda}$ by (4.26) and, by [8, Th. 18.15], $\boldsymbol{\lambda}^\top \boldsymbol{L}_m \boldsymbol{\lambda} \geq \frac{1}{2\lambda_{\max}(L)}\|\boldsymbol{L}_m\boldsymbol{\lambda}\|^2$. The third addend in (4.33) can be rewritten as $(\boldsymbol{k}-\bar{\boldsymbol{k}})^\top \Gamma^{-1} D(\boldsymbol{\rho})^\top(\Gamma\otimes I_n)\boldsymbol{\rho} = \sum_{i=1}^N \left(k_i-\bar{k}_i\right)\boldsymbol{\rho}^{i\top}\boldsymbol{\rho}^i = \boldsymbol{\rho}^\top(K-\bar{K})\boldsymbol{\rho} = \boldsymbol{x}^\top \boldsymbol{L}_n(K-\bar{K})\boldsymbol{L}_n\boldsymbol{x} = (\boldsymbol{x}-\bar{\boldsymbol{x}})^\top \boldsymbol{L}_n(K-\bar{K})\boldsymbol{L}_n(\boldsymbol{x}-\bar{\boldsymbol{x}})$, where $\bar{K} := \operatorname{diag}((\bar{k}_i I_n)_{i\in\mathcal{I}})$. Therefore, the sum of the second and third term in (4.33) is $-(\boldsymbol{x}-\bar{\boldsymbol{x}})^\top \boldsymbol{L}_n\bar{K}\boldsymbol{L}_n(\boldsymbol{x}-\bar{\boldsymbol{x}}) \leq -(\boldsymbol{x}-\bar{\boldsymbol{x}})^\top \boldsymbol{L}_n K^* \boldsymbol{L}_n(\boldsymbol{x}-\bar{\boldsymbol{x}})$, where $K^* := k^* I_{Nn}$. By Lemma 4.4, we finally get

$$\dot{V} \leq -\lambda_{\min}(M_1)\|\boldsymbol{x}-\bar{\boldsymbol{x}}\|^2 - \tfrac{1}{2\lambda_{\max}(L)}\|\boldsymbol{L}_m\boldsymbol{\lambda}\|^2, \tag{4.34}$$

with $M_1 \succ 0$ as in Lemma 4.4.

Let $\bar{\mathcal{P}}$ be any compact sublevel set of V (notice that V is radially unbounded) containing the initial condition $\boldsymbol{\omega}(0) \in S$. $\bar{\mathcal{P}}$ is invariant for the dynamics, since $\dot{V}(\boldsymbol{\omega}) \leq 0$ by (4.34). The set $\mathcal{P} := \bar{\mathcal{P}} \cap S$ is compact, convex and invariant, therefore, by exploiting Lemma 4.3 and the continuous differentiability in Assumption 4.1, we conclude that $\mathcal{A}+\mathcal{B}$ is Lipschitz continuous on $\mathcal{P}$. Therefore, for any initial condition, there exists a unique *global* Carathéodory solution to (4.10), that belongs to $\mathcal{P}$ (and therefore is bounded) for every t [40, Prop. 2.2]. Moreover, by [47, Th. 2], the solution converges to the largest invariant set $\mathcal{O} \subseteq \{\boldsymbol{\omega} \in \mathcal{P} \text{ s.t. } \dot{V}(\boldsymbol{\omega}) = 0\}$.

We can already conclude that $\boldsymbol{x}$ converges to the point $\mathbf{1}_N \otimes x^*$, with x^* the unique v-GNE of the game in (4.2). We next show convergence of the other variables. Take any point $\underline{\boldsymbol{\omega}} := \operatorname{col}(\underline{\boldsymbol{x}}, \underline{\boldsymbol{k}}, \underline{\boldsymbol{z}}, \underline{\boldsymbol{\lambda}}) \in \mathcal{O}$. Since $\dot{V}(\underline{\boldsymbol{\omega}}) = 0$, by (4.34) we have $\underline{\boldsymbol{x}} = \bar{\boldsymbol{x}} = \mathbf{1}_N \otimes x^*$, and $\underline{\boldsymbol{\lambda}} \in \boldsymbol{E}_m$, i.e. $\underline{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \underline{\lambda}$, for some $\underline{\lambda} \in \mathbb{R}^m_{\geq 0}$. Therefore, by expanding (4.32), by $\underline{\boldsymbol{x}} = \bar{\boldsymbol{x}}$, $\underline{\boldsymbol{\rho}} := \boldsymbol{L}_n\underline{\boldsymbol{x}} = \mathbf{0}_{Nn}$ and (4.26), we have

$$\begin{aligned}0 &= (\underline{\boldsymbol{\lambda}}-\bar{\boldsymbol{\lambda}})^\top(\boldsymbol{g}(R\bar{\boldsymbol{x}})-\underline{\boldsymbol{z}}) = (\underline{\lambda}-\lambda^*)^\top g(x^*)\\ &= \underline{\lambda}^\top g(x^*) = \underline{\boldsymbol{\lambda}}^\top(\boldsymbol{g}(\mathcal{R}\bar{\boldsymbol{x}})-\underline{\boldsymbol{z}}),\end{aligned} \tag{4.35}$$

where in the second equality we have used that $\underline{\boldsymbol{z}} \in \boldsymbol{E}_m^\perp$ and the third equality follows from the KKT conditions in (4.4b). Then, let $\underline{\boldsymbol{\omega}}(t) = \operatorname{col}(\underline{\boldsymbol{x}}(t), \underline{\boldsymbol{k}}(t), \underline{\boldsymbol{z}}(t), \underline{\boldsymbol{\lambda}}(t))$ be the trajectory of (4.31) starting at $\underline{\boldsymbol{\omega}}$. By invariance of $\mathcal{O}$, $\underline{\boldsymbol{x}}(t) = \bar{\boldsymbol{x}}$ and $\underline{\lambda}(t) \in \boldsymbol{E}_m$, for all t. Therefore, by (4.10b)-(4.10c), it holds that $\underline{\boldsymbol{k}}(t) \equiv \underline{\boldsymbol{k}}$, $\underline{\boldsymbol{z}}(t) \equiv \underline{\boldsymbol{z}}$, for all t. Hence, the quantity $v := (\boldsymbol{g}(\mathcal{R}\underline{\boldsymbol{x}}(t)) - \boldsymbol{L}_m\underline{\boldsymbol{\lambda}}(t) - \underline{\boldsymbol{z}}(t))$ is a constant along the trajectory $\underline{\boldsymbol{\omega}}(t)$. Suppose by contradiction that $[v]_k > 0$, for some $k = \{1,\dots,Nm\}$. Then, $[\underline{\dot{\boldsymbol{\lambda}}}(t)]_k = [v]_k$ for almost all t, by (4.10d), and $\underline{\boldsymbol{\lambda}}(t)$ grows indefinitely. Since all the solutions of (4.10) are bounded, this is

a contradiction. Therefore, $v \leq \mathbf{0}_m$, and $\underline{\boldsymbol{\lambda}}^\top v = 0$ by (4.35). Equivalently, $v \in N_{\mathbb{R}^{Nm}_{\geq 0}}(\underline{\boldsymbol{\lambda}})$, hence $\underline{\boldsymbol{\lambda}}(t) \equiv \underline{\boldsymbol{\lambda}}$, for all t. We conclude that the points in $\mathcal{O}$ are equilibria.

Moreover, the set $\Lambda(\boldsymbol{\omega}_0)$ of ω-limit points[1] of the solution to (4.10) starting at any $\boldsymbol{\omega}_0 \in S$ is nonempty (by Bolzano-Weierstrass theorem, since all the trajectories of (4.10) are bounded), and $\Lambda(\boldsymbol{\omega}_0) \subseteq \mathcal{O}$ (see the proof of [47, Th.2]). Hence, all the ω-limit points are equilibria. We next show that, for any for any $\boldsymbol{\omega}_0 \in S$, $\Lambda(\boldsymbol{\omega}_0)$ is a singleton; as a consequence, the solution converges to that point [8, Lemma 1.14]. For the sake of contradiction, assume that there are two ω-limit points $\boldsymbol{\omega}_1 = \mathrm{col}(\bar{\boldsymbol{x}}, \hat{\boldsymbol{k}}, \boldsymbol{z}_1, \boldsymbol{\lambda}_1)$, $\boldsymbol{\omega}_2 = \mathrm{col}(\bar{\boldsymbol{x}}, \hat{\boldsymbol{k}}, \boldsymbol{z}_2, \boldsymbol{\lambda}_2) \in \Lambda(\boldsymbol{\omega}_0)$, with $\boldsymbol{\omega}_1 \neq \boldsymbol{\omega}_2$. We note that $\boldsymbol{\omega}_1$ and $\boldsymbol{\omega}_2$ must have the same vector of adaptive gains $\hat{\boldsymbol{k}}$ by definition of ω-limit point, since the k_i's in Algorithm 4.2 are nonincreasing. Let $\boldsymbol{\omega}_3 = \mathrm{col}(\bar{\boldsymbol{x}}, \hat{\boldsymbol{k}}+\mathbf{1}\alpha, \boldsymbol{z}_1, \boldsymbol{\lambda}_1)$, $\alpha \in \mathbb{R}$ chosen such that $\min(\hat{\boldsymbol{k}}+\mathbf{1}\alpha) > \underline{k}$, $\underline{k}$ as in (4.11). By (4.34), $\|\boldsymbol{\omega}(t) - \boldsymbol{\omega}_3\|_Q$ is nonincreasing along the trajectory $\boldsymbol{\omega}(t)$ of (4.31) starting at $\boldsymbol{\omega}_0$. Thus, by definition of ω-limit point, it holds that $\|\boldsymbol{\omega}_1 - \boldsymbol{\omega}_3\|_Q = \|\boldsymbol{\omega}_2 - \boldsymbol{\omega}_3\|_Q$, or $\|\mathrm{col}(\mathbf{0}_{Nn}, \alpha\mathbf{1}_N, \mathbf{0}_{Nm}, \mathbf{0}_{Nm})\|_Q = \|\mathrm{col}(\mathbf{0}_{Nn}, \alpha\mathbf{1}_N, \boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2, \boldsymbol{z}_2 - \boldsymbol{z}_1)\|_Q$. Equivalently, $\boldsymbol{\omega}_1 = \boldsymbol{\omega}_2$, that is a contradiction. ■

4.9.4. PROOF OF THEOREM 4.2

The proof follows as for Theorem 4.1, by defining $\boldsymbol{\omega} := \mathrm{col}\,(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\lambda})$, $\Xi := \boldsymbol{\Omega} \times \mathbb{R}^{Nm} \times \mathbb{R}^{mN}_{\geq 0}$,

$$\mathcal{A}(\boldsymbol{\omega}) := \begin{bmatrix} \mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}) + c\boldsymbol{L}_n \boldsymbol{x} \\ \mathbf{0}_{Nm} \\ \boldsymbol{L}_m \boldsymbol{\lambda} \end{bmatrix}, \; \mathcal{B}(\boldsymbol{\omega}) := \begin{bmatrix} \mathcal{R}^\top \mathbb{G}(\mathcal{R}\boldsymbol{x})^\top \boldsymbol{\lambda} \\ -\boldsymbol{L}_m \boldsymbol{\lambda} \\ -\boldsymbol{g}(\mathcal{R}\boldsymbol{x}) + \boldsymbol{z} \end{bmatrix},$$

with Lyapunov function $V(\boldsymbol{\omega}) = \frac{1}{2}(\|\boldsymbol{x} - \bar{\boldsymbol{x}}\|^2 + \|\boldsymbol{z} - \bar{\boldsymbol{z}}\|^2_\Phi + \|\boldsymbol{\lambda} - \bar{\boldsymbol{\lambda}}\|^2)$, and by exploiting, in place of Lemma 4.4, Lemma 3 in [109]. ■

4.9.5. PROOF OF LEMMA 4.5

By Assumptions 4.2 and 4.4, we have

$$\begin{aligned} &(x - x')^\top (\tilde{\boldsymbol{F}}(x, \boldsymbol{\sigma}) - \tilde{\boldsymbol{F}}(x', \boldsymbol{\sigma}') \\ &= (x - x')^\top (\tilde{\boldsymbol{F}}(x, \boldsymbol{\sigma}) - \tilde{\boldsymbol{F}}(x, P_{\bar{n}}\boldsymbol{\psi}(x)) + \tilde{\boldsymbol{F}}(x, P_{\bar{n}}\boldsymbol{\psi}(x)) - \tilde{\boldsymbol{F}}(x', P_{\bar{n}}\boldsymbol{\psi}(x'))) \\ &\geq \mu \|x - x'\|^2 - \tilde{\theta}_\sigma \|x - x'\| \|\boldsymbol{\sigma} - \mathbf{1}_N \otimes \psi(x)\|. \end{aligned}$$

Moreover, we note that $(\boldsymbol{\sigma} - \mathbf{1}_N \otimes \psi(x)) \in \boldsymbol{E}^\perp_{\bar{n}}$, since $P_{\bar{n}}\boldsymbol{\sigma} = P_{\bar{n}}\boldsymbol{\psi}(x)$, and $\boldsymbol{\sigma}' \in \boldsymbol{E}_{\bar{n}}$. Hence, by (4.26), we have $(\boldsymbol{\sigma} - \boldsymbol{\sigma}')^\top \boldsymbol{L}_{\bar{n}} K^* \boldsymbol{L}_{\bar{n}} (\boldsymbol{\sigma} - \boldsymbol{\sigma}') \geq k^* \lambda_2(L)^2 \|\boldsymbol{\sigma} - P_{\bar{n}}\boldsymbol{\psi}(x)\|^2$, and the conclusion follows readily. ■

[1] $z : [0, \infty) \to \mathbb{R}^n$ has an ω-limit point at $\bar{z}$ if there exists a nonnegative diverging sequence $\{t_k\}_{k \in \mathbb{N}}$ such that $z(t_k) \to \bar{z}$

4.9.6. PROOF OF THEOREM 4.3

The dynamics (4.15) can be recast in the form (4.31), with $\boldsymbol{\omega} = \text{col}\,(x, \boldsymbol{\varsigma}, \boldsymbol{k}, \boldsymbol{z}, \boldsymbol{\lambda})$, $\Xi = \Omega \times \mathbb{R}^{N\bar{n}} \times \mathbb{R}^{N} \times \mathbb{R}^{Nm} \times \mathbb{R}^{Nm}_{\geq 0}$,

$$\mathcal{A}(\boldsymbol{\omega}) = \begin{bmatrix} \tilde{\boldsymbol{F}}(x,\boldsymbol{\sigma}) + B^\top \boldsymbol{L}_{\bar{n}} K \boldsymbol{L}_{\bar{n}} \boldsymbol{\sigma} \\ \boldsymbol{L}_{\bar{n}} K \boldsymbol{L}_{\bar{n}} \boldsymbol{\sigma} \\ -D(\boldsymbol{\rho})^\top (\Gamma \otimes I_{\bar{n}})\, \boldsymbol{\rho} \\ \boldsymbol{0}_{Nm} \\ \boldsymbol{L}_m \boldsymbol{\lambda} \end{bmatrix}, \; \mathcal{B}(\boldsymbol{\omega}) = \begin{bmatrix} \mathbb{G}(x)^\top \boldsymbol{\lambda} \\ \boldsymbol{0}_{N\bar{n}} \\ \boldsymbol{0}_{N} \\ -\boldsymbol{L}_m \boldsymbol{\lambda} \\ -\boldsymbol{g}(x) + \boldsymbol{z} \end{bmatrix}.$$

By proceeding as in the proof of Theorem 4.1, we note that the set S is invariant for the dynamics, since, for all $\boldsymbol{\omega} \in S$, $\frac{\partial}{\partial \boldsymbol{\omega}}(P_{\bar{n}}\boldsymbol{\varsigma})\dot{\boldsymbol{\omega}} = \boldsymbol{0}_{N\bar{n}}$, $\frac{\partial}{\partial \boldsymbol{\omega}}(P_m \boldsymbol{z})\dot{\boldsymbol{\omega}} = \boldsymbol{0}_{Nm}$.

Analogously to the proof of Lemma 4.2, it can be shown that any equilibrium point $\bar{\boldsymbol{\omega}} := \text{col}\left(\bar{x}, \bar{\boldsymbol{\varsigma}}, \bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}}\right) \in S$ of (4.15) is such that $\bar{\boldsymbol{\lambda}} = \boldsymbol{1}_N \otimes \lambda^*$, the pair $(\bar{x}, \lambda^*)$ satisfies the KKT conditions in (4.4), and $\bar{\boldsymbol{\sigma}} := \boldsymbol{\psi}(\bar{x}) + \bar{\boldsymbol{\varsigma}} = \boldsymbol{1}_N \otimes \psi(\bar{x})$. Moreover, for any pair (x^*, λ^*) satisfying the KKT conditions in (4.4), there exists $\bar{\boldsymbol{z}} \in \mathbb{R}^{mN}$ such that $\text{col}(x^*, \boldsymbol{1}_N \otimes \psi(x^*) - \boldsymbol{\psi}(x^*), \boldsymbol{k}, \bar{\boldsymbol{z}}, \boldsymbol{1}_N \otimes \lambda^*) \in S$ is an equilibrium for (4.15), for any $\boldsymbol{k} \in \mathbb{R}^N$. The proof is omitted.

Let $\bar{\boldsymbol{\omega}} = \text{col}\left(\bar{x}, \bar{\boldsymbol{\varsigma}}, \bar{\boldsymbol{k}}, \bar{\boldsymbol{z}}, \bar{\boldsymbol{\lambda}}\right) \in S$ be an equilibrium of (4.15) such that $k^* = \min(\bar{\boldsymbol{k}}) > \underline{k}$, $\underline{k}$ as in (4.17), and consider the quadratic Lyapunov function $V = \frac{1}{2}\|\boldsymbol{\omega} - \bar{\boldsymbol{\omega}}\|^2_Q$, where $Q = \text{diag}(I_n, I_{N\bar{n}}, \Gamma^{-1}, P_m + \boldsymbol{L}_m^+, I_{Nm})$. Analogously to the proof of Theorem 4.2, it holds that $(\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})^\top Q(\mathcal{B}(\boldsymbol{\omega}) - \mathcal{B}(\bar{\boldsymbol{\omega}})) \geq 0$, and that $\dot{V}(\boldsymbol{\omega}) \leq -(\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})^\top Q\,(\mathcal{A}(\boldsymbol{\omega}) - \mathcal{A}(\bar{\boldsymbol{\omega}}))$, for all $\boldsymbol{\omega} \in S$. Also we note that

$$\begin{aligned} &(x - \bar{x})^\top B^\top \boldsymbol{L}_{\bar{n}} K \boldsymbol{L}_{\bar{n}} (\boldsymbol{\sigma} - \bar{\boldsymbol{\sigma}}) + (\boldsymbol{\varsigma} - \bar{\boldsymbol{\varsigma}})^\top \boldsymbol{L}_{\bar{n}} K \boldsymbol{L}_{\bar{n}} (\boldsymbol{\sigma} - \bar{\boldsymbol{\sigma}}) \\ &= (\boldsymbol{\varsigma} + Bx + d - (\bar{\boldsymbol{\varsigma}} + B\bar{x} + d))^\top \boldsymbol{L}_{\bar{n}} K \boldsymbol{L}_{\bar{n}} (\boldsymbol{\sigma} - \bar{\boldsymbol{\sigma}}) \\ &= (\boldsymbol{\sigma} - \bar{\boldsymbol{\sigma}})^\top \boldsymbol{L}_{\bar{n}} K \boldsymbol{L}_{\bar{n}} (\boldsymbol{\sigma} - \bar{\boldsymbol{\sigma}}), \end{aligned}$$

where $d := \text{col}((d_i)_{i \in \mathcal{I}})$, and that $(\boldsymbol{k} - \bar{\boldsymbol{k}})^\top \Gamma^{-1} D(\boldsymbol{\rho})^\top (\Gamma \otimes I_n)\boldsymbol{\rho} = (\boldsymbol{\sigma} - \bar{\boldsymbol{\sigma}})^\top \boldsymbol{L}_n (K - \bar{K}) \boldsymbol{L}_n (\boldsymbol{\sigma} - \bar{\boldsymbol{\sigma}})$ as in the proof of Theorem 4.2. Hence, by Lemma 4.5, we obtain, for all $\boldsymbol{\omega} \in S$

$$\begin{aligned} \dot{V}(\boldsymbol{\omega}) \leq -\lambda_{\min}(M_2)(\|x - \bar{x}\|^2 + \|\boldsymbol{\sigma} - \boldsymbol{1}_N \otimes \psi(x)\|^2) \\ - \tfrac{1}{2\lambda_{\max}(L)} \|\boldsymbol{L}_m \boldsymbol{\lambda}\|^2, \end{aligned}$$

with $M_2 \succ 0$ as in (4.17). Then, existence of a unique global solution for the system in (4.15) and convergence to an equilibrium point follows as for Theorem 4.1. ■

4.9.7. PROOF OF THEOREM 4.4

Analogously to Lemma 4.5, it can be shown that for any $c > \underline{c}$, for any $(x, \boldsymbol{\sigma})$ such that $P_{\bar{n}}\boldsymbol{\sigma} = P_{\bar{n}}\boldsymbol{\psi}(x)$ and any $(x', \boldsymbol{\sigma}')$ such that $\boldsymbol{\sigma}' = P_{\bar{n}}\boldsymbol{\psi}(x')$, it holds that $(x - x')^\top(\tilde{\boldsymbol{F}}(x, \boldsymbol{\sigma}) - \tilde{\boldsymbol{F}}(x', \boldsymbol{\sigma}')) + c(\boldsymbol{\sigma} - \boldsymbol{\sigma}')^\top \boldsymbol{L}_{\bar{n}} (\boldsymbol{\sigma} - \boldsymbol{\sigma}') \geq \delta \|\text{col}(x - x', \boldsymbol{\sigma} - \boldsymbol{1}_N \otimes \psi(x))\|^2$, for some $\delta > 0$. Then, the proof follows analogously to Theorem 4.3. ■

4.9.8. PROOF OF THEOREM 4.5

Under the coordinate transformations in (4.21), the dynamics in Algorithm 4.5 read as (4.24), where the input $\tilde{u}_i$ in Algorithm 4.5 has been chosen by design according to Algorithm 4.2, under Assumption 4.5. Therefore, existence of a unique bounded solution

and convergence of ζ_i to x_i^* (and of the variables $\boldsymbol{\zeta}_i, k_i, z_i, \lambda_i$), for all $i \in \mathcal{I}$, follows from Theorem 4.1. On the other hand, we note that, for all $i \in \mathcal{I}$ and all $k \in \mathcal{M}_i$, $E_{i,k}$ is Hurwitz, because it is in canonical controllable form and the coefficients of the last row are by design the coefficients of an Hurwitz polynomial. Therefore, E_i is also Hurwitz, and hence the dynamics in (4.24b) are ISS with respect to the input $\tilde{u}_i$ [79, Lemma 4.6]. In turn, the input $\tilde{u}_i$ is bounded, by boundedness of trajectories in Theorem 4.1, Assumption 4.1 and Lemma 4.3; moreover, by the convergence in Theorem 4.1, the KKT conditions in (4.4) and by continuity, we have that $\tilde{u}_i \to \mathbf{0}_{n_i}$ for $t \to \infty$. Hence, for all $i \in \mathcal{I}$, $v_i \to \mathbf{0}$ asymptotically (this follows by definition of ISS, see [79, Ex. 4.58]). By the definition of ζ_i, we also have $x_i \to x_i^*$, for all $i \in \mathcal{I}$. ■

III

GRAPHICAL STRUCTURES IN GAMES UNDER PARTIAL-DECISION INFORMATION

5

NE SEEKING OVER TIME-VARYING NETWORKS WITH LINEAR RATE

Competition is the best form of motivation.

Cordae

There are many situations that could be modelled as games, but where it doesn't make a lot of sense to do so.

Conor Muldon

We design a distributed algorithm for learning Nash equilibria over time-varying communication networks. Our algorithm is based on projected pseudo-gradient dynamics, augmented with consensual terms. Under strong monotonicity and Lipschitz continuity of the game mapping, we provide a simple proof of linear convergence, based on a contractivity property of the iterates. Compared to similar solutions proposed in literature, we also allow for time-varying communication and derive tighter bounds on the step sizes that ensure convergence. In fact, in our numerical simulations, our algorithm outperforms the existing gradient-based methods, when the step sizes are set to their theoretical upper bounds. Finally, to relax the assumptions on the network structure, we propose a different pseudo-gradient algorithm, which is guaranteed to converge on time-varying balanced directed graphs.

Parts of this chapter have been published in [25].

5.1. INTRODUCTION

NASH equilibrium problems arise in several network systems, where multiple selfish decision-makers, or agents, aim at optimizing their individual, yet inter-dependent, objective functions. Engineering applications include communication networks [57], demand-side management in the smart grid [119], charging of electric vehicles [71] and demand response in competitive markets [88]. From a game-theoretic perspective, the challenge is to assign the agents behavioral rules that eventually ensure the attainment of a NE, a joint action from which no agent has an incentive to unilaterally deviate.

Literature review: Typically, NE seeking algorithms are designed under the assumption that each agent can access the decisions of all the competitors [151], [13], [126]. This *full-decision information* hypothesis requires the presence of a coordinator, that broadcast the data to the network, and it is impractical for some applications [68], [30]. One example is the Nash-Cournot competition model described in [82], where the profit of each of a group of firms depends not only on its own production, but also on the whole amount of sales, a quantity not directly accessible by any of the firms. Therefore, in recent years, there has been an increased attention for fully-distributed algorithms that allow to compute NEs relying on local information only. In this dissertation, we consider the so-called *partial-decision information* scenario, where the agents engage in nonstrategic information exchange with some neighbors on a network; based on the data received, they can estimate and eventually reconstruct the actions of all the competitors. This setup has only been introduced very recently. In particular, most of the results available resort to (projected) gradient and consensus dynamics, both in continuous time [146], [63], and discrete time. For the discrete time case, fixed-step algorithms were proposed in [120], [136], [109] (the latter for generalized games), all exploiting a certain restricted monotonicity property. Alternatively, the authors of [135] developed a gradient-play scheme by leveraging contractivity properties of doubly stochastic matrices. Nevertheless, in all these approaches theoretical guarantees are provided only for step sizes that are typically very small, affecting the speed of convergence. Furthermore, all the methods cited are designed for a time-invariant, undirected network. To the best of our knowledge, switching communication topologies have only been addressed with diminishing step sizes. For instance, the early work [82] considered aggregative games over time-varying jointly connected undirected graphs. This result was extended by the authors of [10] to games with coupling constraints. In [123], an asynchronous gossip algorithm was presented to seek a NE over directed graphs. The drawback is that vanishing steps typically result in slow convergence.

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

- We propose a fully-distributed projected gradient-play method, that is guaranteed to converge with linear rate when the network weight matrix is doubly stochastic. With respect to [135], we consider a time-varying communication network and we allow for constrained action sets. Moreover, differently from the state of the art, we provide an upper bound on the step size that does not vanish as the number of agents increases (§5.3);

- We show via numerical simulations that, even in the case of fixed networks, our algorithm outperforms the existing pseudo-gradient based dynamics, when the step sizes are set to their theoretical upper bounds (§5.5);
- We prove that linear convergence to a NE on time varying weight-balanced directed graphs can be achieved via a forward-backward algorithm [56, §12.7.2], which has been studied in [109], [136], but only for the special case of fixed undirected networks (§5.4).

To improve readability, the proofs are in the chapter appendix. We refer to Appendices A, B, C for the basic notation and mathematical background.

5.2. MATHEMATICAL SETUP

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

$$\forall i \in \mathcal{I}: \quad \min_{y_i \in \Omega_i} J_i(y_i, x_{-i}). \tag{5.1}$$

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

Definition 5.1. A Nash equilibrium is a set of strategies $x^* = \text{col}\left((x_i^*)_{i\in\mathcal{N}}\right) \in \Omega$ such that, for all $i \in \mathcal{I}$:

$$J_i\left(x_i^*, x_{-i}^*\right) \leq \inf\{J_i\left(y_i, x_{-i}^*\right) \mid y_i \in \Omega_i\}. \qquad \square$$

The following regularity assumptions are common for NE problems, see, e.g., [109, Ass. 1], [136, Ass. 1].

Standing Assumption 5.1 (Regularity and convexity)**.** For each $i \in \mathcal{I}$, the set Ω_i is nonempty, closed and convex; J_i is continuous and the function $J_i(\cdot, x_{-i})$ is convex and continuously differentiable for every x_{-i}. $\square$

Under Standing Assumption 5.1, a joint action x^* is a NE of the game in (5.1) if and only if it solves the variational inequality VI(F, Ω) [56, Prop. 1.4.2], or, equivalently, if and only if, for any $\alpha > 0$ [56, Prop. 1.5.8],

$$x^* = \text{proj}_\Omega(x^* - \alpha F(x^*)), \tag{5.2}$$

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

$$F(x) := \text{col}\left((\nabla_{x_i} J_i(x_i, x_{-i}))_{i\in\mathcal{I}}\right). \tag{5.3}$$

Next, we postulate a sufficient condition for the existence of a unique NE, namely the strong monotonicity of the pseudo-gradient [56, Th. 2.3.3]. This assumption is always

used for (G)NE seeking under partial-decision information with fixed step sizes, e.g., in [136, Ass. 2], [109, Ass. 3]. It implies strong convexity of the functions $J_i(\cdot, x_{-i})$ for every x_{-i}, but not necessarily (strong) convexity of J_i in the full argument.

Standing Assumption 5.2. The pseudo-gradient mapping in (5.3) is μ-strongly monotone and θ_0-Lipschitz continuous, for some $\mu, \theta_0 > 0$. □

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

5

Standing Assumption 5.3. For each $k \in \mathbb{N}$, the graph $\mathcal{G}^k$ is strongly connected. □

Assumption 5.1. For all $k \in \mathbb{N}$, the following hold:

(i) *Self-loops:* $w^k_{i,i} > 0$ for all $i \in \mathcal{I}$;

(ii) *Double stochasticity:* $W_k \mathbf{1}_N = \mathbf{1}_N$, $\mathbf{1}_N^\top W_k = \mathbf{1}_N^\top$. □

Remark 5.1. Assumption 5.1(i) is intended just to ease the notation. Instead, Assumption 5.1(ii) is stronger. It is typically used for networked problems on undirected symmetric graphs, e.g., in [82, Ass. 6], [10, Ass. 3], [135, Ass. 3], justified by the fact that it can be satisfied by assigning the following Metropolis weights to the communication:

$$\tilde{w}^k_{i,j} = \begin{cases} w^k_{i,j}/(\max\{d^k_i, d^k_j\} + 1) & \text{if} \quad j \in \mathcal{N}^k_i \backslash \{i\}; \\ 0 & \text{if} \quad j \notin \mathcal{N}^k_i; \\ 1 - \sum_{j\in\mathcal{N}^k_i\backslash\{i\}} \tilde{w}^k_{i,j} & \text{if} \quad i = j. \end{cases}$$

In practice, to satisfy Assumption 5.1(ii) in case of symmetric communication, even under time-varying topology, it suffices for the agents to exchange their in-degree with their neighbors at every time step. Therefore, Standing Assumption 5.3 and Assumption 5.1 are easily fulfilled for undirected graphs connected at each step. For directed graphs, given any strongly connected topology, weights can be assigned such that the resulting weight matrix (with self-loops) is doubly stochastic, via an iterative distributed process [69]. However, this can be impractical if the network is time-varying. □

Under Assumption 5.1, it holds that $\sigma_{N-1}(W_k) < 1$, for all k, where $\sigma_{N-1}(W_k)$ denotes the second largest singular value of W_k. Moreover, for any $y \in \mathbb{R}^N$,

$$\|W_k(y - \mathbf{1}_N \bar{y})\| \le \sigma_{N-1}(W_k)\|y - \mathbf{1}_N \bar{y}\|, \tag{5.4}$$

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

Assumption 5.2. There exists $\bar{\sigma} \in (0, 1)$ such that $\sigma_{N-1}(W_k) \le \bar{\sigma}$, for all $k \in \mathbb{N}$. □

Algorithm 5.1. Combine-then-adapt pseudo-gradient method

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

- Distributed averaging:

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

- Local variables update:

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

5.3. DISTRIBUTED NASH EQUILIBRIUM SEEKING

IN this section, we present a pseudo-gradient algorithm to seek a NE of the game (5.1) in a fully-distributed way. To cope with partial-decision information, each agent keeps an estimate of all other agents' actions. Let $\boldsymbol{x}_i = \mathrm{col}((\boldsymbol{x}_{i,j})_{j\in\mathcal{I}}) \in \mathbb{R}^{Nn}$, where $\boldsymbol{x}_{i,i} := x_i$ and $\boldsymbol{x}_{i,j}$ is agent i's estimate of agent j's action, for all $j \neq i$; also, $\boldsymbol{x}_{j,-i} = \mathrm{col}((\boldsymbol{x}_{j,l})_{l\in\mathcal{I}\setminus\{i\}})$. The agents aim at asymptotically reconstructing the true value of the opponents' actions, based on the data received by their neighbors. The procedure is summarized in Algorithm 5.1. Each agent updates its estimates according to consensus dynamics, then its action via a gradient step. We remark that each agent computes the partial gradient of its cost in its local estimates $\boldsymbol{x}_i$, not on the actual joint action x.

To write the algorithm in compact form, let $\boldsymbol{x} = \mathrm{col}((\boldsymbol{x}_i)_{i\in\mathcal{I}})$; as in [109, Eq. 13-14], let, for all $i \in \mathcal{I}$,

$$\mathcal{R}_i := \begin{bmatrix} \mathbf{0}_{n_i \times n_{<i}} & I_{n_i} & \mathbf{0}_{n_i \times n_{>i}} \end{bmatrix} \in \mathbb{R}^{n_i \times n}, \tag{5.5}$$

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

$$\boldsymbol{F}(\boldsymbol{x}) := \mathrm{col}\left((\nabla_{x_i} J_i\left(x_i, \boldsymbol{x}_{i,-i}\right))_{i\in\mathcal{I}}\right). \tag{5.6}$$

Therefore, Algorithm 5.1 reads in compact form as:

$$\boldsymbol{x}^{k+1} = \mathrm{proj}_{\boldsymbol{\Omega}}(\boldsymbol{W}_k \boldsymbol{x}^k - \alpha \mathcal{R}^\top \boldsymbol{F}(\boldsymbol{W}_k \boldsymbol{x}^k)), \tag{5.7}$$

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

Lemma 5.1 ([23, Lemma 3])**.** The mapping $\boldsymbol{F}$ in (5.6) is θ-Lipschitz continuous, for some $\mu \le \theta \le \theta_0$. □

Theorem 5.1. Let Assumptions 5.1-5.2 hold and let

$$M_\alpha = \begin{bmatrix} 1 - \frac{2\alpha\mu}{N} + \frac{\alpha^2\theta_0^2}{N} & \left(\frac{\alpha(\theta+\theta_0)+\alpha^2\theta_0\theta}{\sqrt{N}}\right)\bar{\sigma} \\ \left(\frac{\alpha(\theta+\theta_0)+\alpha^2\theta_0\theta}{\sqrt{N}}\right)\bar{\sigma} & \left(1+2\alpha\theta+\alpha^2\theta^2\right)\bar{\sigma}^2 \end{bmatrix}. \tag{5.8}$$

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

$$\rho_\alpha := \lambda_{\max}(M_\alpha) = \|M_\alpha\| < 1, \tag{5.9}$$

then, for any initial condition, the sequence $(\boldsymbol{x}^k)_{k\in\mathbb{N}}$ generated by Algorithm 5.1 converges to $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, where x^* is the NE of the game in (5.1), with linear rate: for all $k \in \mathbb{N}$,

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

Lemma 5.2. The condition in (5.9) holds if $\alpha > 0$ and

$$\alpha < \tfrac{\bar{\sigma}}{3\theta_0}, \tag{5.10a}$$

$$\alpha < \tfrac{2\mu}{\theta_0^2}, \tag{5.10b}$$

$$\begin{aligned} 0 < {} & 2\mu(1-\bar{\sigma}^2) - \alpha(\bar{\sigma}^2(2\theta_0\theta + \theta^2 + 4\mu\theta + 2\theta_0^2) - \theta_0^2) \\ & - \alpha^2(\theta_0\theta^2 + \mu\theta^2 + 2\theta_0^2\theta)2\bar{\sigma}^2 - \alpha^3 2\theta_0^2\theta^2\bar{\sigma}^2. \end{aligned} \tag{5.10c}$$

$\square$

Proof. The condition in (5.10a) implies that $M_\alpha \succ 0$ (by diagonal dominance and positivity of the diagonal elements, as can be checked by recalling that $\theta \leq \theta_0$, $\mu \leq \theta_0$, $N \geq 2$, $\bar{\sigma} < 1$). The inequalities in (5.10b)-(5.10c) are the Sylvester's criterion for the matrix $I_2 - M_\alpha$: they impose that $[I_2 - M_\alpha]_{1,1} > 0$ (5.10b) and $\det(I_2 - M_\alpha) > 0$ (5.10c), hence $I_2 - M_\alpha \succ 0$. Altogether, this implies $\|M_\alpha\| < 1$. ■

Remark 5.2. The conditions in (5.10) always hold for α small enough, since, in the monomial inequality (5.10c), the constant term is $2\mu(1-\bar{\sigma}^2) > 0$. While explicit solutions are known for cubic equations, we prefer the compact representation in (5.10c). The bounds in (5.10) are not tight, and in practice better bounds on the step size α are obtained by simply checking the Euclidean norm of the 2×2 matrix M_α in (5.8). Instead, the key observation is that the conditions in (5.10) do not depend on the number of agents: given the parameters $\bar{\sigma}$, μ, θ_0 and θ, a constant α that ensures convergence can be chosen independently of N. On the contrary, the rate $\sqrt{\rho_\alpha}$ does depend on N and, in fact, it approaches 1 as N grows unbounded (analogously to the results in [120], [136], [135]). $\square$

Remark 5.3. Compared to [135, Alg. (7)] (or [136, Alg. 1]), in Algorithm 5.1 the agents *first* exchange information with their neighbors, and *then* evaluate their gradient term. Moreover, differently from [135, Th. 1], Theorem 1 provides a contractivity property for the iterates in (5.7) that holds at *each* step. This has beneficial consequences in terms of robustness, see Remark 5.6. $\square$

5.3.1. TECHNICAL DISCUSSION

In Algorithm 5.1, the partial gradients $\nabla_{x_i} J_i$ are evaluated on the local estimates $\boldsymbol{x}_{i,-i}$, not on the actual strategies x_{-i}. Only if the estimates of all the agents coincide with the actual value, i.e., $\boldsymbol{x} = \mathbf{1}_N \otimes x$, we have that $\boldsymbol{F}(\boldsymbol{x}) = F(x)$. As a consequence, the mapping $\mathcal{R}^\top \boldsymbol{F}$ is not necessarily monotone, not even under strong monotonicity of the game mapping. Indeed, the loss of monotonicity is the main technical difficulty arising from the partial-decision information setup. Some works [63], [120], [136], [109], [23] deal with this issue by leveraging a restricted strong monotonicity property, which can be ensured, by opportunely choosing the parameter γ, for the augmented mapping $\boldsymbol{F}_{\mathrm{a}}(\boldsymbol{x}) := \gamma \mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{L}\boldsymbol{x}$, where $\boldsymbol{L} = L \otimes I_n$ and L is the Laplacian of a fixed undirected connected network. Since the unique solution of the VI$(\boldsymbol{F}_{\mathrm{a}}, \boldsymbol{\Omega})$ is $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, with x^* the unique NE of the game in (5.1) [136, Prop. 1], one can design NE seeking algorithms via standard solution methods for variational inequalities (or the corresponding monotone inclusions, [109]). For instance, in [136], a FB algorithm [56, p. 12.4.2] is proposed to solve VI$(\boldsymbol{F}_{\mathrm{a}}, \boldsymbol{\Omega})$, resulting in the algorithm

$$\boldsymbol{x}^{k+1} = \mathrm{proj}_{\boldsymbol{\Omega}}\big(\boldsymbol{x}^k - \tau(\boldsymbol{F}_{\mathrm{a}}(\boldsymbol{x}))\big). \tag{5.11}$$

We also recover this iteration when considering [109, Alg. 1] in the absence of coupling constraints. However, exploiting the monotonicity of $\boldsymbol{F}_{\mathrm{a}}$ results in conservative upper bounds on the parameters τ and γ, and hence in slow convergence (see §5.4-5.5). More recently, the authors of [135] studied the convergence of (5.11) based on contractivity of the iterates, in the case of a fixed undirected network with doubly stochastic weight matrix W, unconstrained action sets (i.e., $\Omega = \mathbb{R}^n$), and by fixing $\tau = 1$, which results in the algorithm:

$$\boldsymbol{x}^{k+1} = (W \otimes I_N)\boldsymbol{x} - \alpha \mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}^k). \tag{5.12}$$

Nonetheless, the upper bound on α provided in [135, Th. 1] is decreasing to zero when the number of agents N grows unbounded (in contrast with that in Theorem 5.1, see Lemma 5.2).

5.4. BALANCED DIRECTED GRAPHS

In this section, we relax the double stochasticity condition in Assumption 5.1 to the following.

Assumption 5.3. For all $k \in \mathbb{N}$, the communication graph $\mathcal{G}_k$ is weight balanced: $(\mathbf{1}_N^\top W_k)^\top = W_k \mathbf{1}_N$. □

For weight-balanced digraphs, in-degree and out-degree of each node coincide. Therefore, the matrix $\tilde{L}_k := (L_k + L_k^\top)/2 = D_k - (W_k + W_k^\top)/2$ is itself the symmetric Laplacian of an undirected graph. Besides, such a graph is connected by Standing Assumption 5.3; hence $\tilde{L}_k$ has a simple eigenvalue in 0, and the others are positive, i.e., $\lambda_2(\tilde{L}_k) > 0$.

Assumption 5.4. There exist $\bar{\sigma}, \bar{\lambda} > 0$ such that $\sigma_{\max}(L_k) \le \bar{\sigma}$ and $\lambda_2(\tilde{L}_k) \ge \bar{\lambda}$, for all $k \in \mathbb{N}$. □

Remark 5.4. Assumptions 5.2 and 5.4 always hold if the networks are chosen among a finite family. Yet, $\bar{\sigma}$, $\tilde{\sigma}$ and $\bar{\lambda}$ are global parameters, that could be difficult to compute in a

Algorithm 5.2. Forward-backward for fully-distributed NE seeking

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

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

distributed way; upper/lower bounds might be available for special classes of networks, e.g., unweighted graphs. □

To seek a NE over switching balanced digraphs, we propose the iteration in Algorithm 5.2. In compact form, it reads as

$$\boldsymbol{x}^{k+1} = \mathrm{proj}_{\boldsymbol{\Omega}}\big(\boldsymbol{x}^k - \tau(\gamma \mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}^k) + \boldsymbol{L}_k \boldsymbol{x}^k)\big), \tag{5.13}$$

where $\boldsymbol{L}_k = L_k \otimes I_n$. Clearly, (5.13) is the same scheme of (5.11), just adapted to take the switching topology into account. In fact, the proof of convergence of Algorithm 5.2 is based on a restricted strong monotonicity property of the operator

$$\boldsymbol{F}_{\mathrm{a}}^k(\boldsymbol{x}) := \gamma \mathcal{R}^\top \boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{L}_k \boldsymbol{x}, \tag{5.14}$$

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

Theorem 5.2. Let Assumptions 5.3-5.4 hold, and let

$$\begin{aligned} M &:= \gamma \begin{bmatrix} \frac{\mu}{N} & -\frac{\theta_0+\theta}{2\sqrt{N}} \\ -\frac{\theta_0+\theta}{2\sqrt{N}} & \frac{\bar{\lambda}}{\gamma} - \theta \end{bmatrix},\\ \bar{\mu} &:= \lambda_{\min}(M),\\ \gamma_{\max} &:= \tfrac{4\mu\bar{\lambda}}{(\theta_0+\theta)^2+4\mu\theta},\\ \bar{\theta} &:= \theta + \tilde{\sigma}\\ \tau_{\max} &:= 2\bar{\mu}/\bar{\theta}^2,\\ \rho_{\gamma,\tau} &:= 1 - 2\tau\bar{\mu} + \tau^2\bar{\theta}^2. \end{aligned} \tag{5.15}$$

If $\gamma \in (0, \gamma_{\max})$, then $M \succ 0$ and, for any $\tau \in (0, \tau_{\max})$, for any initial condition, the sequence $(\boldsymbol{x}^k)_{k\in\mathbb{N}}$ generated by Algorithm 5.2 converges to $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, where x^* is the unique NE of the game in (5.1), with linear rate: for all $k \in \mathbb{N}$,

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

Remark 5.5. Differently from the bound $\alpha_{\max}$ in (5.8), $\tau_{\max}$ in (5.15) vanishes as N grows (fixed the other parameters), as $\bar{\mu}$ decreases to 0 (by continuity of the eigenvalues). □

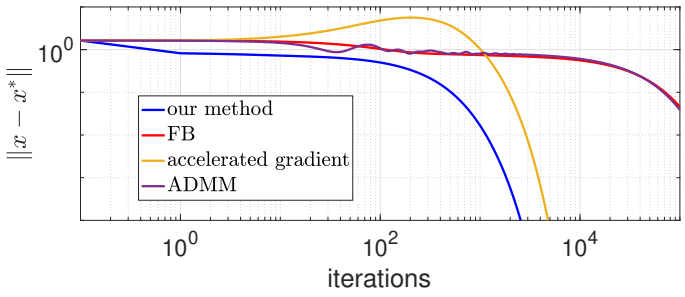

Figure 5.1: Distance from the NE with step sizes that guarantee convergence, for different pseudo-gradient NE seeking methods: our method (Algorithm 5.1), the FB method (Algorithm 5.2), the accelerated gradient method in [136, Alg. 2], the ADMM-based method in [120, Alg. 1].

Remark 5.6. Based on Theorems 5.1, 5.2, it can be proven that the discrete-time systems (5.7), (5.13) are ISS with respect to additive disturbances, with ISS-Lyapunov function $\|\boldsymbol{x}-\boldsymbol{x}^*\|^2$. By Lipschitz continuity of the updates, this implies ISS for noise both on the communication and in the evaluation of the partial gradients. □

5

5.5. NUMERICAL EXAMPLE: A NASH-COURNOT GAME

We consider the Nash-Cournot game in [109, §6]. N firms produce a commodity that is sold to m markets. Each firm $i \in \mathcal{I} = \{1,\dots,N\}$ can only participate in $n_i \le m$ of the markets; its action $x_i \in \mathbb{R}^{n_i}$ is the vector of quantities of product to be sent to these n_i markets, bounded by the local constraints $\mathbf{0}_{n_i} \le x_i \le X_i$. Let $A_i \in \mathbb{R}^{m\times n_i}$ be the matrix that specifies which markets firm i participates in. Specifically, $[A_i]_{k,j} = 1$ if $[x_i]_j$ is the amount of product sent to the k-th market by agent i, $[A_i]_{k,j} = 0$ otherwise, for all $k = 1,\dots,m$, $j = 1,\dots,n_i$. Let $A := [A_1 \dots A_N]$; then $Ax = \sum_{i=1}^N A_i x_i \in \mathbb{R}^m$ are the quantities of total product delivered to each market. Firm i aims at maximizing its profit, i.e., minimizing the cost function $J_i(x_i, x_{-i}) = c_i(x_i) - p(Ax)^\top A_i x_i$. Here, $c_i(x_i) = x_i^\top Q_i x_i + q_i^\top x_i$ is firm i's production cost, with $Q_i \in \mathbb{R}^{n_i\times n_i}$, $Q_i \succ 0$, $q_i \in \mathbb{R}^{n_i}$. Instead, $p : \mathbb{R}^m \to \mathbb{R}^m$ associates to each market a price that depends on the amount of product delivered to that market. Specifically, the price for the market k, for $k = 1,\dots,m$, is $[p(Ax)]_k = \bar{P}_k - \chi_k [Ax]_k$, where $\bar{P}_k$, $\chi_k > 0$. We set $N = 20$, $m = 7$. The market structure is as in [109, Fig. 1], that defines which firms are allowed to participate in which markets. Therefore, $x \in \mathbb{R}^n$, with $n = 32$. We select randomly with uniform distribution r_k in $[1,2]$, Q_i diagonal with diagonal elements in $[14,16]$, q_i in $[1,2]$, $\bar{P}_k$ in $[10,20]$, χ_k in $[1,3]$, X_i in $[5,10]$, for all $i \in \mathcal{I}$, $k = 1,\dots,m$. The resulting setup satisfies Standing Assumptions 5.1-5.2 [109, §VI]. The firms cannot access the production of all the competitors, but can communicate with some neighbors on a network.

We first consider the case of a fixed, undirected graph, under Assumption 5.1. Algorithm 5.2 in this case reduces to [136, Alg. 1]. We compare Algorithms 5.1-5.2 with the inexact ADMM in [120] and the accelerated gradient method in [136], for the step sizes that ensure convergence. Specifically, we set α as in Theorem 5.1 for Algorithm 5.1. The convergence of all the other Algorithms is based on the monotonicity of $\boldsymbol{F}_a$ in (5.14); hence we set γ as in Theorem 5.2. Instead of using the conservative bounds in (5.15) for the

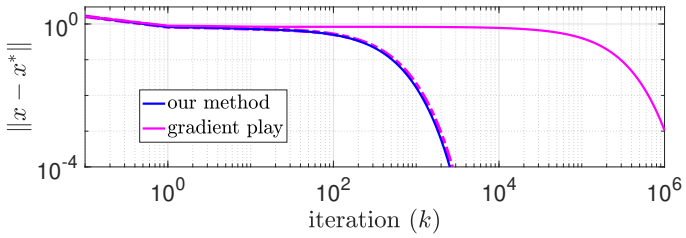

Figure 5.2: Distance from the NE for our method (Algorithm 5.1), with step size $\alpha = 2 * 10^{-3}$ (upper bound in Theorem 5.1), and the gradient play method in [135, Alg. 1], with step size $\alpha = 4 * 10^{-6}$ (upper bound in [135, Th. 1]). Algorithm 5.1 converges much faster, thanks to the larger step size. The scheme in [135, Alg. 1] still converges if we set $\alpha = 2 * 10^{-3}$ (dashed line, not supported theoretically).

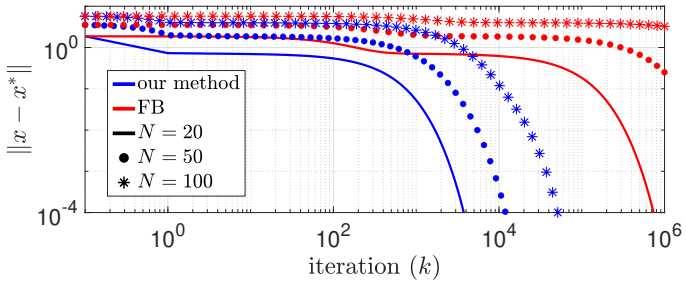

Figure 5.3: Comparison of our method (Algorithm 5.1) and the FB method (Algorithm 5.2), on a time-varying graph, for 20, 50 or 100 agents, with the step sizes set to their theoretical upper bounds.

parameters, $\bar{\mu}$ and $\bar{\theta}$, we obtain a better result by computing the values numerically. $\boldsymbol{F}_a$ is (non-restricted) strongly monotone for our parameters, hence also the convergence result for [136, Alg. 2] holds. Figure 5.1 shows that Algorithm 5.1 outperforms all the other methods (we also note that the accelerated gradient in [136, Alg. 2] requires two projections and two communications per iterations). As a numerical example, we also compare Algorithm 5.1 with the scheme in (5.12) by removing the local constraints, in Figure 5.2.

For the case of doubly stochastic time-varying networks, we randomly generate 5 connected graphs and for each iteration we pick one with uniform distribution. In Figure 5.3, we compare the performance of Algorithms 5.1-5.2, for step sizes set to their upper bounds as in Theorems 5.1-5.2. Since the theoretical convergence rate in Theorems 5.1-5.2 worsens as the number of agents grows, to show how the performance is affected in practice, we repeat the experiment for different values of N and random market structures (Figure 5.3).

Finally, in Figure 5.4, we test Algorithm 5.2 with communication topology chosen at each step with uniform distribution between two unweighted balanced directed graphs: the directed ring, where each agent i can send information to the agent $i+1$ (with the convention $N+1 \equiv 1$), and a graph where agent i is also allowed to transmit to agent $i+2$, for all $i \in \mathcal{I}$.

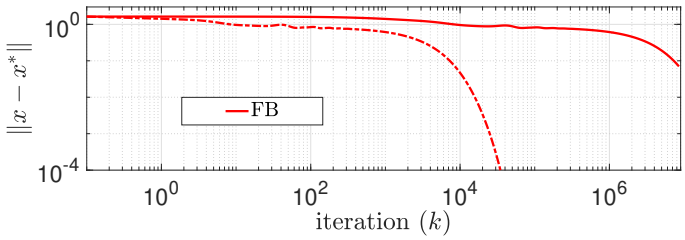

Figure 5.4: Distance from the NE for the FB method in Algorithm 5.2, on a time-varying digraph. Since the networks are sparse, Theorem 5.2 ensures convergence only for small step sizes ($\gamma = 5 * 10^{-4}$, $\tau = 3 * 10^{-4}$), and convergence is slow (solid line). However, the bounds are conservative: the iteration still converges with τ 1000 times larger than the theoretical value (dashed line).

5.6. CONCLUSION

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

5.7. APPENDIX

5.7.1. PROOF OF THEOREM 5.1

We define the estimate consensus subspace $\boldsymbol{E} := \{\boldsymbol{y} \in \mathbb{R}^{Nn} \mid \boldsymbol{y} = \mathbf{1}_N \otimes y,\ y \in \mathbb{R}^n\}$ and its orthogonal complement $\boldsymbol{E}_\perp = \{\boldsymbol{y} \in \mathbb{R}^{Nn} \mid (\mathbf{1}_N \otimes I_n)^\top \boldsymbol{y} = \mathbf{0}_n\}$. Thus, any vector $\boldsymbol{x} \in \mathbb{R}^{Nn}$ can be written as $\boldsymbol{x} = \boldsymbol{x}_\parallel + \boldsymbol{x}_\perp$, where $\boldsymbol{x}_\parallel = \text{proj}_{\boldsymbol{E}}(\boldsymbol{x}) = \frac{1}{N}(\mathbf{1}_N \mathbf{1}_N^\top \otimes I_n)\boldsymbol{x}$, $\boldsymbol{x}_\perp = \text{proj}_{\boldsymbol{E}_\perp}(\boldsymbol{x})$, and $\boldsymbol{x}_\parallel^\top \boldsymbol{x}_\perp = 0$. Also, we use the shorthand notation $\boldsymbol{F}\boldsymbol{x}$ and Fx in place of $\boldsymbol{F}(\boldsymbol{x})$ and $F(x)$. We recast the iteration in (5.7) as

$$\boldsymbol{x}^{k+1} = \text{proj}_{\boldsymbol{\Omega}}(\hat{\boldsymbol{x}}^k - \alpha \mathcal{R}^\top \boldsymbol{F}\hat{\boldsymbol{x}}^k),\ \hat{\boldsymbol{x}}^k = \boldsymbol{W}_k \boldsymbol{x}^k. \tag{5.16}$$

Let x^* be the unique NE of the game in (5.1), and $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$. We recall that $x^* = \text{proj}_\Omega(x^* - \alpha F x^*)$ by (5.2), and then $\boldsymbol{x}^* = \text{proj}_{\boldsymbol{\Omega}}(\boldsymbol{x}^* - \alpha \mathcal{R}^\top \boldsymbol{F}\boldsymbol{x}^*)$. Moreover, $\boldsymbol{W}_k \boldsymbol{x}^* = (W_k \otimes I_n)(\mathbf{1}_N \otimes x^*) = \mathbf{1}_N \otimes x^* = \boldsymbol{x}^*$; hence $\boldsymbol{x}^*$ is a fixed point for (5.16). Let $\boldsymbol{x}^k = \boldsymbol{x} \in \mathbb{R}^{Nn}$ and $\hat{\boldsymbol{x}} = \boldsymbol{W}_k \boldsymbol{x} = \hat{\boldsymbol{x}}_\parallel + \hat{\boldsymbol{x}}_\perp = \mathbf{1}_N \otimes \hat{x}_\parallel + \hat{\boldsymbol{x}}_\perp \in \mathbb{R}^{Nn}$. Thus, it holds that

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

$$
\begin{aligned}
&= \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|^2 + \|\hat{\boldsymbol{x}}_\perp\|^2 \\
&\quad + \alpha^2 \|\mathcal{R}^\top(\boldsymbol{F}\hat{\boldsymbol{x}} - \boldsymbol{F}\hat{\boldsymbol{x}}_\parallel + \boldsymbol{F}\hat{\boldsymbol{x}}_\parallel - \boldsymbol{F}\boldsymbol{x}^*)\|^2 \\
&\quad - 2\alpha(\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*)^\top \mathcal{R}^\top(\boldsymbol{F}\hat{\boldsymbol{x}} - \boldsymbol{F}\hat{\boldsymbol{x}}_\parallel) \\
&\quad - 2\alpha(\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*)^\top \mathcal{R}^\top(\boldsymbol{F}\hat{\boldsymbol{x}}_\parallel - \boldsymbol{F}\boldsymbol{x}^*) \\
&\quad - 2\alpha\hat{\boldsymbol{x}}_\perp^\top \mathcal{R}^\top(\boldsymbol{F}\hat{\boldsymbol{x}} - \boldsymbol{F}\hat{\boldsymbol{x}}_\parallel) \\
&\quad - 2\alpha\hat{\boldsymbol{x}}_\perp^\top \mathcal{R}^\top(\boldsymbol{F}\hat{\boldsymbol{x}}_\parallel - \boldsymbol{F}\boldsymbol{x}^*) \\
&\leq \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|^2 + \|\hat{\boldsymbol{x}}_\perp\|^2 + \alpha^2(\theta^2\|\hat{\boldsymbol{x}}_\perp\|^2 + \tfrac{\theta_0^2}{N}\|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|^2 \\
&\quad + \tfrac{2\theta_0\theta}{\sqrt{N}}\|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|\|\hat{\boldsymbol{x}}_\perp\|) + \tfrac{2\alpha\theta}{\sqrt{N}}\|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|\|\hat{\boldsymbol{x}}_\perp\| \\
&\quad - \tfrac{2\alpha\mu}{N}\|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|^2 + 2\alpha\theta\|\hat{\boldsymbol{x}}_\perp\|^2 + \tfrac{2\alpha\theta_0}{\sqrt{N}}\|\hat{\boldsymbol{x}}_\perp\|\|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|,
\end{aligned}
\tag{5.17}
$$

where the first inequality follows by nonexpansiveness of the projection ([8, Prop. 4.16]), and to bound the addends in (5.17) we used, in the order:

- 3rd term: $\|\mathcal{R}\| = 1$, Lipschitz continuity of $\boldsymbol{F}$, and $\|\boldsymbol{F}\hat{\boldsymbol{x}}_\parallel - \boldsymbol{F}\boldsymbol{x}^*\| = \|F\hat{x}_\parallel - Fx^*\| \leq \theta_0\|\hat{x}_\parallel - x^*\| = \frac{\theta_0}{\sqrt{N}}\|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|$;
- 4th term: $\|\mathcal{R}(\mathbf{1}\otimes(\hat{x}_\parallel - x^*))\| = \|\hat{x}_\parallel - x^*\| = \frac{1}{\sqrt{N}}\|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|$;
- 5th term: $(\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*)^\top\mathcal{R}^\top(\boldsymbol{F}\hat{\boldsymbol{x}}_\parallel - \boldsymbol{F}\boldsymbol{x}^*) = (\hat{x}_\parallel - x^*)^\top(F\hat{x}_\parallel - Fx^*) \geq \mu\|\hat{x}_\parallel - x^*\|^2 = \frac{1}{N}\|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|^2$;
- 6th term: Lipschitz continuity of $\boldsymbol{F}$;
- 7th term: $\|\boldsymbol{F}\hat{\boldsymbol{x}}_\parallel - \boldsymbol{F}\boldsymbol{x}^*\| \leq \frac{\theta_0}{\sqrt{N}}\|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{x}^*\|$ as above.

Besides, for every $\boldsymbol{x} = \boldsymbol{x}_\parallel + \boldsymbol{x}_\perp \in \mathbb{R}^{Nn}$ and for all $k \in \mathbb{N}$, it holds that $\hat{\boldsymbol{x}} = \boldsymbol{W}_k\boldsymbol{x} = \boldsymbol{x}_\parallel + \boldsymbol{W}_k\boldsymbol{x}_\perp$, where $\boldsymbol{W}_k\boldsymbol{x}_\perp \in \boldsymbol{E}_\perp$, by doubly stochasticity of W_k, and $\|\hat{\boldsymbol{x}}_\perp\| = \|\boldsymbol{W}_k\boldsymbol{x}_\perp\| \leq \bar{\sigma}\|\boldsymbol{x}_\perp\|$ by (5.4) and properties of the Kronecker product. Therefore we can finally write, for all $k \in \mathbb{N}$, for all $\boldsymbol{x}^k \in \mathbb{R}^{Nn}$,

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

■

5.7.2. PROOF OF THEOREM 5.2

Let x^* be the unique NE of the game in (5.1), and $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$. We recall that the null space $\mathrm{null}(\boldsymbol{L}_k) = \boldsymbol{E} = \{\boldsymbol{y} \in \mathbb{R}^{Nn} \mid \boldsymbol{y} = \mathbf{1}_N \otimes y,\ y \in \mathbb{R}^n\}$ by Standing Assumption 5.3. Therefore, $\boldsymbol{L}_k\boldsymbol{x}^* = \mathbf{0}_N$ and $\boldsymbol{x}^*$ is a fixed point of the iteration in (5.13) by (5.2). With $\boldsymbol{F}_\mathrm{a}^k$ as in (5.14), for all $k \in \mathbb{N}$, for any $\boldsymbol{x} \in \mathbb{R}^{Nn}$, it holds that $(\boldsymbol{x} - \boldsymbol{x}^*)^\top(\boldsymbol{F}_\mathrm{a}^k\boldsymbol{x} - \boldsymbol{F}_\mathrm{a}^k\boldsymbol{x}^*) = (\boldsymbol{x} - \boldsymbol{x}^*)^\top\gamma\mathcal{R}^\top(\boldsymbol{F}\boldsymbol{x} - \boldsymbol{F}\boldsymbol{x}^*) + (\boldsymbol{x} - \boldsymbol{x}^*)^\top\boldsymbol{L}_k(\boldsymbol{x} - \boldsymbol{x}^*) = (\boldsymbol{x} - \boldsymbol{x}^*)^\top\gamma\mathcal{R}^\top(\boldsymbol{F}\boldsymbol{x} - \boldsymbol{F}\boldsymbol{x}^*) + (\boldsymbol{x} - \boldsymbol{x}^*)^\top\tilde{\boldsymbol{L}}_k(\boldsymbol{x} - \boldsymbol{x}^*)$, where $\tilde{\boldsymbol{L}}_k = (\boldsymbol{L}_k + \boldsymbol{L}_k^\top)/2 = (L_k + L_k^\top)/2 \otimes I_n = \tilde{L}_k \otimes I_n$, and $\tilde{L}_k$ is the Laplacian of a connected graph (see §5.4) and $\lambda_2(\tilde{L}_k) > \bar{\lambda}$ by Assumption 5.4. Therefore we can apply [109, Lemma 3] to conclude that $(\boldsymbol{x} - \boldsymbol{x}^*)^\top(\boldsymbol{F}_\mathrm{a}^k\boldsymbol{x} - \boldsymbol{F}_\mathrm{a}^k\boldsymbol{x}^*) \geq \bar{\mu}\|\boldsymbol{x} - \boldsymbol{x}^*\|^2$, with $\bar{\mu} > 0$ as in (5.15). Also, $\boldsymbol{F}_\mathrm{a}^k$ is

Lipschitz continuous with constant $\bar{\theta} = \theta + \tilde{\sigma}$, $\tilde{\sigma}$ as in Assumption 5.4. Therefore we have

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

where in the first inequality we used [8, Prop. 4.16], and $\rho_{\gamma,\tau} \in (0,1)$ if τ is chosen as in Theorem 5.2. ■

6

NE SEEKING OVER DIRECTED GRAPHS

Oh yes, the past can hurt. But you can either run from it, or learn from it.

Rafiki (The Lion King, movie 1994)

Learning is rooted in repetition and convexity, meaning that the reading of a single text twice is more profitable than reading two different things once.

Nassim Nicholas Taleb

We consider the Nash equilibrium problem in a partial-decision information scenario. While the existing methods assume undirected or balanced communication, in this chapter we allow for non-balanced, directed graphs. We propose a fully-distributed pseudo-gradient scheme, which is guaranteed to converge with linear rate to a Nash equilibrium, under strong monotonicity and Lipschitz continuity of the game mapping. Our algorithm requires global knowledge of the communication structure, namely of the Perron-Frobenius eigenvector of the weight matrix and of a certain constant related to the graph connectivity. Therefore, we adapt the procedure to setups where the network is not known in advance, by computing the eigenvector online and by means of vanishing step sizes.

Parts of this chapter have been published in [27].

6.1. INTRODUCTION

GAME theory is a powerful tool to model and control the decision-making process of selfish agents, that aim at optimizing their individual, but inter-dependent, objective functions. This scenario arises in several relevant engineering applications, such as congestion control in traffic networks [6], smart-grid management [119], demand response in competitive markets [88] and analysis of social dynamics [68]. Often, the goal (either of the agents or of a coordinator that pursues network regulation by imposing incentives or behavioral rules) is the attainment of a NE, a joint strategy from which it is not convenient for any agent to unilaterally deviate.

In fact, a recent part of the literature focuses on designing distributed NE seeking algorithms, where the computational effort is partitioned among the agents [11], [46], [126]. Nonetheless, typically these methods still assume the presence of a central coordinator that can broadcast some data – for instance, the average of all the agents' strategies, in the case of aggregative games [12]. Unfortunately, this requirement is impractical in some domains [133]. To overcome this limitation, we consider *fully-distributed* schemes, where the agents only rely on the information locally exchanged over a network, via peer-to-peer communication. In particular, the main challenge is that the cost function of each agent may depend on the strategies of some other non-neighboring agents. One example is the Cournot competition model described in [82], where the profit of each of a group of firms depends not only on its own production, but also on the total supply, a quantity not directly accessible by any of the firms. To remedy the lack of knowledge, each agent can estimate and eventually reconstruct the strategies of all the competitors (or an aggregation value), based on the data received from its neighbors.

Such a *partial-decision information* setup has only been introduced very recently. Most of the available results resort to (projected) pseudo-gradient and consensus dynamics [10], [47], [83], [109], [137], [146]. Alternatively, schemes based on a proximal-point iteration were studied in [22]; a fully-distributed fictitious play algorithm was proposed in [133]. These approaches assume undirected communication, which might be unrealistic, e.g., in wireless systems, if the agents send signals at different power levels, implying unilateral transmission capability. Fewer works deal with asymmetric networks. Under the assumption of balanced weights, continuous-time dynamics were proposed in [50] for aggregative games; most recently, we also addressed generally-coupled-cost games via a fixed-step forward-backward method [25]. To the best of our knowledge, the only discrete-time NE seeking algorithm that takes into account non-balanced digraphs is the asynchronous gossip-based scheme in [123].

Even in the context of distributed optimization, most algorithms are designed with doubly stochastic weight matrices, which enjoy several convenient properties, not least that the average of the agents' estimates is preserved over time. However, doubly stochastic weights cannot be easily assigned over directed networks. An alternative is to rely on column stochastic graphs, which maintain the average invariance and only require the agents to know their out-degree. Yet, this is impractical in setups where the agents broadcast some information, but ignoring which of the other nodes can receive it; or if some of the communication links can fail. In contrast, distributed design of row stochastic matrices is straightforward, as it suffices for each agent to locally assign appropriate weights to the incoming information. However, the use of row stochastic graphs

comes with technical challenges, since many properties of doubly stochastic matrices are lost. Of major interest for this chapter is the approach in [143]: to correct the imbalance caused by employing row stochastic weights, the algorithm exploits the information contained in the *Perron-Frobenius* (PF) eigenvector of the weight matrix, which is computed online.

Contribution: Motivated by the above, we design the first synchronous, fully-distributed algorithm to compute a NE over directed non-balanced communication networks. Our contributions are summarized as follows:

- We prove that any row stochastic primitive matrix with positive diagonal enjoys a contractivity property, in a Hilbert space weighted by its PF eigenvector. We later exploit this general result to prove convergence of our equilibrium seeking dynamics (§6.2);
- We design a fully-distributed, fixed-step gradient algorithm to seek a NE over strongly connected directed graphs, which is guaranteed to converge with liner rate under strong monotonicity of the game mapping. In our method, the pseudo-gradient component is divided by the entries of the PF eigenvector of the network. Although this technique has already been adopted in distributed optimization [143], we give a new, powerful, monotone-operator-theoretic interpretation, which greatly simplifies our analysis (§6.3.1);
- We show that convergence is retained even if the graph is not known in advance and the PF eigenvector is computed online, provided that a small-enough step size is chosen. Since computing the upper bound distributedly can be troublesome, we also provided convergence guarantees for vanishing steps (§6.3.2).

To improve readability, the proofs are in the chapter appendix. We refer to Appendices A, B, C for the basic notation and mathematical background.

6.2. MATHEMATICAL SETUP

6.2.1. THE GAME

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

$$\forall i \in \mathcal{I}: \quad \min_{y_i \in \Omega_i} J_i(y_i, x_{-i}). \tag{6.1}$$

The technical problem we consider here is the distributed computation of a NE, as formalized next.

Definition 6.1. A collective strategy $x^* = \mathrm{col}\left((x_i^*)_{i\in\mathcal{I}}\right)$ is a Nash equilibrium if, for all $i \in \mathcal{I}$,

$$J_i\left(x_i^*, x_{-i}^*\right) \leq \inf\{J_i\left(y_i, x_{-i}^*\right) \mid (y_i, x_{-i}^*) \in \Omega\}. \qquad \square$$

Next, we postulate common regularity assumptions for the constraint sets and cost functions [109, Ass. 1], [137, Ass. 1].

Standing Assumption 6.1. For each $i \in \mathcal{I}$, the set Ω_i is non-empty, closed and convex; J_i is continuous and $J_i(\cdot, x_{-i})$ is convex and continuously differentiable for every x_{-i}. □

Under Standing Assumption 6.1, a collective strategy x^* is a NE of the game in (6.1) if and only if it is a solution of the variational inequality VI(F,Ω)[1] [56, Prop. 1.4.2], where F is the *pseudo-gradient* mapping of the game:

$$F(x) := \operatorname{col}\left((\nabla_{x_i} J_i(x_i, x_{-i}))_{i\in\mathcal{I}}\right). \tag{6.2}$$

Equivalently, x^* is a NE if and only if

$$\forall i \in \mathcal{I}: \quad x_i^* = \operatorname{proj}_{\Omega_i}(x_i^* - \beta_i \nabla_{x_i} J_i(x_i^*, x_{-i}^*)), \tag{6.3}$$

for arbitrary positive scalars β_i's [56, Prop. 1.5.8]. A sufficient condition for the existence and uniqueness of a NE is the strong monotonicity of the pseudo-gradient [56, Th. 2.3.3], as postulated next. This assumption has always been used for NE seeking under partial-decision information with fixed step sizes, e.g., [109, Ass. 2], [47, Ass. 4], [137, Ass. 2].

Standing Assumption 6.2. The pseudo-gradient mapping F in (6.2) is μ-strongly monotone and θ_0-Lipschitz continuous, for some $\mu, \theta_0 > 0$. □

6

6.2.2. NETWORK COMMUNICATION

The agents can exchange information with some neighbors over a directed communication network $\mathcal{G}(\mathcal{I},\mathcal{E})$. The ordered pair (i,j) belongs to the set of edges, $\mathcal{E}$, if and only if agent i can receive information from agent j. We denote $W \in \mathbb{R}^{N\times N}$ the weight matrix of $\mathcal{G}$ and $w_{i,j} := [W]_{i,j}$, with $w_{i,j} > 0$ if $(i,j) \in \mathcal{E}$, $w_{i,j} = 0$ otherwise; $d_i := \deg(i) = \sum_{j=1}^N w_{i,j}$ and $\mathcal{N}_i = \{j \mid (i,j) \in \mathcal{E}_k\}$ the in-degree and the set of in-neighbors of agent i, respectively.

Standing Assumption 6.3. The communication graph $\mathcal{G}$ is strongly connected. □

Standing Assumption 6.4. The weight matrix W satisfies the following conditions:

(i) *Self-loops:* $w_{i,i} > 0$ for all $i \in \mathcal{I}$;

(ii) *Row stochasticity:* $W_k \mathbf{1}_N = \mathbf{1}_N$. □

Remark 6.1. Standing Assumption 6.4 can be fulfilled on any digraph, if the agents can access their own in-degree, by *locally* assigning weights to the received information. □

Under Standing Assumptions 6.3-6.4, by the PF theorem, W has a simple eigenvalue in 1; all the other (complex) eigenvalues of W have absolute value strictly smaller than 1. Besides, there exist a vector $q = \operatorname{col}((q_i)_{i\in\mathcal{I}})$ such that

$$q \in \mathbb{R}^N_{>0}, \quad q^\top W = q^\top, \quad \mathbf{1}_N^\top q = 1. \tag{6.4}$$

[1] We recall that, given a set $S \subseteq \mathbb{R}^m$ and a mapping $\psi : S \to \mathbb{R}^m$, the VI$(\psi, S)$ is the problem of finding $\omega^* \in S$ such that $\langle \psi(\omega^*), \omega - \omega^* \rangle \geq 0$, for all $\omega \in S$.

We call q the (left) *Perron-Frobenius eigenvector* of W. Let

$$Q := \mathrm{diag}((q_i)_{i\in\mathcal{I}}). \tag{6.5}$$

Clearly, $Q \succ 0$. Unless W is doubly stochastic, W is not nonexpansive in $\mathcal{H}_I$, i.e., $\sigma_{\max}(W) > 1$. This is one of the main technical challenges to face when studying fixed-point iterations over directed graphs [123]. To deal with this complication, it was shown in [36, Lemma 1] that W is nonexpansive (averaged, indeed) in $\mathcal{H}_Q$. Next, we provide an additional contractivity result, which we exploit later on.

Lemma 6.1. For any $y \in \mathbb{R}^N$, $\|W(y - \mathbf{1}_N q^\top y)\|_Q \le \bar{\sigma}\|y - \mathbf{1}_N q^\top y\|_Q$, where $\bar{\sigma} := \sigma_{N-1}(Q^{\frac{1}{2}} W Q^{-\frac{1}{2}}) < 1$. □

If W is also column stochastic, Lemma 6.1 holds with $q = \frac{1}{N}\mathbf{1}_N$ and $Q = \frac{1}{N} I_N$, and we recover a well-known property of doubly stochastic matrices [25, Eq. 4].

Remark 6.2. [143, Lemma 1] states that there exist a norm and $\bar{\sigma} > 0$ such that the property in Lemma 6.1 holds; instead, we explicitly characterized both the norm and $\bar{\sigma}$, which proves very advantageous in our analysis (particularly so because the matrix Q is diagonal), see §6.3. □

6.2.3. PARTIAL-DECISION INFORMATION SCENARIO

In our setup, agent $i \in \mathcal{I}$ can only access its own feasible set Ω_i and an analytic expression of its own cost function J_i. However, the agents cannot evaluate the actual value of the cost $J_i(x_i, x_{-i})$ (or the partial derivative $\nabla_{x_i} J_i(x_i, x_{-i})$), since they cannot access the strategies of all the competitors x_{-i}. Instead, the agents only rely on the information exchanged locally with their neighbors over the communication graph $\mathcal{G}$. To cope with the lack of knowledge, the general assumption for this partial-decision information scenario is that each agent keeps an estimate of all other agents' actions [109], [82], [47]. Then, the agents aim at reconstructing the actual values, based on the data received from their neighbors. We denote $\boldsymbol{x}_i = \mathrm{col}((\boldsymbol{x}_{i,j})_{j\in\mathcal{I}}) \in \mathbb{R}^n$, where $\boldsymbol{x}_{i,i} := x_i$ and $\boldsymbol{x}_{i,j}$ is agent i's estimate of agent j's action, for all $j \neq i$; $\boldsymbol{x}_{j,-i} = \mathrm{col}((\boldsymbol{x}_{j,l})_{l\in\mathcal{I}\setminus\{i\}})$; $\boldsymbol{x} = \mathrm{col}((\boldsymbol{x}_i)_{i\in\mathcal{I}})$. As in [109, Eq.13-14], we define

$$\mathcal{R}_i := \begin{bmatrix} \mathbf{0}_{n_i \times n_{<i}} & I_{n_i} & \mathbf{0}_{n_i \times n_{>i}} \end{bmatrix}, \tag{6.6}$$

where $n_{<i} := \sum_{j<i, j\in\mathcal{I}} n_j$, $n_{>i} := \sum_{j>i, j\in\mathcal{I}} n_j$. In simple terms, $\mathcal{R}_i$ selects the i-th n_i-dimensional component from an n-dimensional vector, i.e., $\mathcal{R}_i \boldsymbol{x}_i = \boldsymbol{x}_{i,i} = x_i$. We denote by $\mathcal{R} := \mathrm{diag}\big((\mathcal{R}_i)_{i\in\mathcal{I}}\big)$; thus, we have $x = \mathcal{R}\boldsymbol{x}$. Moreover, we define the *extended pseudo-gradient* mapping $\boldsymbol{F}$ as

$$\boldsymbol{F}(\boldsymbol{x}) := \mathrm{col}\big((\nabla_{x_i} J_i\left(x_i, \boldsymbol{x}_{i,-i}\right))_{i\in\mathcal{I}}\big). \tag{6.7}$$

Lemma 6.2 (23, Lemma 3])**.** The mapping $\boldsymbol{F}$ in (6.7) is θ-Lipschitz continuous, for some $\theta \in [\mu, \theta_0]$: for any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^{Nn}$, $\|\boldsymbol{F}(\boldsymbol{x}) - \boldsymbol{F}(\boldsymbol{y})\| \le \theta\|\boldsymbol{x} - \boldsymbol{y}\|$. □

We remark that in (6.7), each agent i evaluates its partial gradients $\nabla_{x_i} J_i\left(x_i, \boldsymbol{x}_{i,-i}\right)$ on the local estimate $\boldsymbol{x}_{i,-i}$, not on the actual strategies x_{-i}. Only when the estimates of all the agents coincide with the actual value, i.e., $\boldsymbol{x} = \mathbf{1}_N \otimes x$, we have that $\boldsymbol{F}(\boldsymbol{x}) = F(x)$. As a consequence, the mapping $\mathcal{R}^\top \boldsymbol{F}$ is not monotone, not even under strong monotonicity of the game mapping F in Standing Assumption 6.2. Indeed, the loss of monotonicity is the main technical difficulty arising in the partial-decision information scenario [109], [137].

6.3. FULLY-DISTRIBUTED NASH EQUILIBRIUM SEEKING

In this section, we present a pseudo-gradient method (along with some variants) to seek a NE in a fully-distributed way. Before going into details, we need some definitions. Let

$$\bar{Q} := \operatorname{diag}((q_i I_{n_i})_{i\in\mathcal{I}}), \quad \boldsymbol{Q} := Q \otimes I_n. \tag{6.8}$$

We define the consensus subspace as $\boldsymbol{E} = \{\boldsymbol{y} \in \mathbb{R}^{Nn} | \boldsymbol{y} = \mathbf{1}_N \otimes y, y \in \mathbb{R}^n\}$ and its orthogonal complement in $\mathcal{H}_{\boldsymbol{Q}}$ as $\boldsymbol{E}_\perp^Q = \{\boldsymbol{y} \in \mathbb{R}^{Nn} | (q \otimes I_n)^\top \boldsymbol{y} = \mathbf{0}_n\}$. Let us also denote by $\operatorname{proj}_S^{\boldsymbol{Q}} : \mathbb{R}^{Nn} \to S$ the $\boldsymbol{Q}$-weighted Euclidea projection onto a closed convex set $S \subseteq \mathbb{R}^{Nn}$, i.e., $\operatorname{proj}_S^{\boldsymbol{Q}}(\boldsymbol{x}) = \operatorname{argmin}_{\xi\in S} \|\boldsymbol{x} - \xi\|_{\boldsymbol{Q}}$. Thus, any vector of estimates $\boldsymbol{x} \in \mathbb{R}^{Nn}$ can be written as $\boldsymbol{x} = \boldsymbol{x}_\parallel + \boldsymbol{x}_\perp$, where $\boldsymbol{x}_\parallel = \operatorname{proj}_{\boldsymbol{E}}^{\boldsymbol{Q}}(\boldsymbol{x}) = (\mathbf{1}_N q^\top \otimes I_n)\boldsymbol{x}$, $\boldsymbol{x}_\perp = \operatorname{proj}_{\boldsymbol{E}_\perp^Q}^{\boldsymbol{Q}}(\boldsymbol{x})$, and it holds that $\langle \boldsymbol{x}_\parallel, \boldsymbol{x}_\perp \rangle_{\boldsymbol{Q}} = 0$. Clearly, if the estimates of the agents $\boldsymbol{x} \in \boldsymbol{E}$, then $\boldsymbol{x}_i = x$ for all $i \in \mathcal{I}$, namely the estimate of each agent coincides with the actual collective strategy x.

6.3.1. CASE 1: KNOWN q AND $\bar{\sigma}$

Our basic fully-distributed NE seeking algorithm is summarized in Algorithm 6.1, where α is a fixed step size. Each agent update its estimates according to consensus dynamics, then its strategy via a projected pseudo-gradient step. We remark that each agent computes the partial gradient of its cost in its local estimate, not on the actual joint strategy x.

Compared to similar pseudo-gradient dynamics proposed in the literature [137], [25], the novelty of Algorithm 6.1 is that the cost related components $\nabla_{x_i} J_i$ are weighted by the reciprocal of the elements q_i of the PF eigenvector. This operation enables convergence on row stochastic graphs, and in fact it is not necessary for doubly stochastic graphs, for which $q = \mathbf{1}$. The idea behind this key modification is that $(W - \mathbf{1}q^\top)$ is contractive in $\mathcal{H}_Q$, while the game-mapping F is strongly monotone in $\mathcal{H}_I$; instead, we would like both properties to hold in the same space. Division by the PF eigenvector achieves this goal, as we show next. Let

$$\bar{F}(x) =: \bar{Q}^{-1} F(x), \quad \bar{\boldsymbol{F}}(\boldsymbol{x}) := \bar{Q}^{-1} \boldsymbol{F}(\boldsymbol{x}). \tag{6.9}$$

Lemma 6.3. The operator $\bar{F}$ is $\bar{\mu}$-strongly monotone in $\mathcal{H}_{\bar{Q}}$ and $\bar{\ell}_0$-Lipschitz continous in $\mathcal{H}_{\bar{Q}}$, for some $\bar{\mu}, \bar{\ell}_0 > 0$. The mapping $\bar{\boldsymbol{F}}$ is $\bar{\ell}$-Lipschitz continuous from $\mathcal{H}_{\boldsymbol{Q}}$ to $\mathcal{H}_{\bar{Q}}$, for some $\bar{\ell} > 0$, i.e., for any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^{Nn}$, $\|\bar{\boldsymbol{F}}(\boldsymbol{x}) - \bar{\boldsymbol{F}}(\boldsymbol{y})\|_{\bar{Q}} \leq \bar{\ell} \|\boldsymbol{x} - \boldsymbol{y}\|_{\boldsymbol{Q}}$. □

Algorithm 6.1. NE seeking with known PF eigenvector

Initialization: $\forall i \in \mathcal{I}$, set $x_i^0 \in \Omega_i$, $\boldsymbol{x}_{i,-i}^0 \in \mathbb{R}^{N-n_i}$.
Iterate until convergence: each agent $i \in \mathcal{I}$ does:

$$\begin{aligned}\hat{\boldsymbol{x}}_i^k &= \textstyle\sum_{j\in\mathcal{N}_i} w_{i,j}\boldsymbol{x}_j^k \\ x_i^{k+1} &= \mathrm{proj}_{\Omega_i}(\hat{\boldsymbol{x}}_{i,i}^k - \tfrac{\alpha}{q_i}\nabla_{x_i} J_i(\hat{\boldsymbol{x}}_i^k)) \\ \boldsymbol{x}_{i,-i}^{k+1} &= \hat{\boldsymbol{x}}_{i,-i}^k.\end{aligned}$$

Remark 6.3. Lemmas 6.1 and 6.3 provide a general, operator-theoretic interpretation of the approach in [143], where a similar technique is used in the context of distributed optimization. □

In compact form, Algorithm 6.1 reads as

$$\boldsymbol{x}^{k+1} = \mathrm{proj}_{\boldsymbol{\Omega}}(\mathcal{F}(\boldsymbol{x}^k)), \tag{6.10}$$

where $\boldsymbol{\Omega} := \{\boldsymbol{x} \in \mathbb{R}^{Nn} \mid \mathcal{R}\boldsymbol{x} \in \Omega\}$, $\boldsymbol{W} := W \otimes I_n$ and

$$\mathcal{F}(\boldsymbol{x}) := \boldsymbol{W}\boldsymbol{x} - \alpha\mathcal{R}^\top \bar{\boldsymbol{F}}(\boldsymbol{W}\boldsymbol{x}). \tag{6.11}$$

The following Lemma shows a contractivity property of the operator $\mathcal{F}$ and represents the cornerstone we use to prove convergence of our NE seeking schemes. The result is based on the strong monotonicity of $\bar{F}$ in $\mathcal{H}_{\bar{Q}}$ and on Lemma 6.1.

Lemma 6.4. Let

$$M_\alpha := \begin{bmatrix} 1-2\alpha\bar{\mu}\lambda_{\min}(Q)+\alpha^2\bar{\ell}^2 & (2\alpha\bar{\ell})\bar{\sigma} \\ (2\alpha\bar{\ell})\bar{\sigma} & (1+2\alpha\bar{\ell}+\alpha^2\bar{\ell}^2)\bar{\sigma}^2 \end{bmatrix} \tag{6.12}$$

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

$$\rho_\alpha := \lambda_{\max}(M_\alpha) = \|M_\alpha\| < 1, \tag{6.13}$$

then the operator $\mathcal{F}$ in (6.11) is $\sqrt{\rho_\alpha}$-restricted contractive in $\mathcal{H}_{\boldsymbol{Q}}$ with respect to the consensus subspace $\boldsymbol{E}$, i.e., for any $\boldsymbol{x} \in \mathbb{R}^{Nn}$, $\boldsymbol{y} \in \boldsymbol{E}$, it holds that $\|\mathcal{F}(\boldsymbol{x}) - \mathcal{F}(\boldsymbol{y})\|_{\boldsymbol{Q}} \leq \sqrt{\rho_\alpha}\|\boldsymbol{x}-\boldsymbol{y}\|_{\boldsymbol{Q}}$. □

Remark 6.4. The condition in (6.13) can always be satisfied by choosing α small enough; an explicit upper bound can be obtained as in [25, Lemma 2]. □

Theorem 6.1. Let $\alpha > 0$ satisfy the condition in (6.13). Then, for any initial condition, the sequence $(\boldsymbol{x}^k)_{k\in\mathbb{N}}$ generated by Algorithm 6.1 converges to $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, where x^* is the NE of the game in (6.1), with linear rate: for all $k \in \mathbb{N}$,

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

Algorithm 6.2. NE seeking with online estimation of PF eigenvector

Initialization: $\forall i \in \mathcal{I}$, set $x_i^0 \in \Omega_i$, $\boldsymbol{x}_{i,-i}^0 \in \mathbb{R}^{N-n_i}$, $\hat{q}_i^0 = \mathbf{e}_i^N$.
Iterate until convergence: each agent $i \in \mathcal{I}$ does:

$$\begin{aligned}
\hat{q}_i^{k+1} &= \textstyle\sum_{j\in\mathcal{N}_i} w_{i,j}\hat{q}_j^k \\
x_i^{k+1} &= \mathrm{proj}_{\Omega_i}(\hat{\boldsymbol{x}}_{i,i}^k - \alpha^k(\hat{q}_{i,i}^k)^{-1}\nabla_{x_i} J_i(\hat{\boldsymbol{x}}_i^k)) \\
\boldsymbol{x}_{i,-i}^{k+1} &= \hat{\boldsymbol{x}}_{i,-i}^k \qquad \hat{\boldsymbol{x}}_i^k = \textstyle\sum_{j\in\mathcal{N}_i} w_{i,j}\boldsymbol{x}_j^k.
\end{aligned}$$

Proof. By (6.3), we infer that x^* is the NE if and only if $x^* = \mathrm{proj}_\Omega(x^* - \alpha Q^{-1}F(x^*))$. Together with $\boldsymbol{W}\boldsymbol{x}^* = \boldsymbol{x}^*$ and $\boldsymbol{F}(\boldsymbol{x}^*) = F(x^*)$, this implies that $\boldsymbol{x}^*$ is a fixed point for the iteration in (6.10). Therefore we can write

$$\begin{aligned}
\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\|_{\boldsymbol{Q}} &= \|\mathrm{proj}_{\boldsymbol{\Omega}}(\mathcal{F}(\boldsymbol{x}^k)) - \mathrm{proj}_{\boldsymbol{\Omega}}(\mathcal{F}(\boldsymbol{x}^*))\|_{\boldsymbol{Q}} \\
&= \|\mathrm{proj}_{\boldsymbol{\Omega}}^{\boldsymbol{Q}}(\mathcal{F}(\boldsymbol{x}^k)) - \mathrm{proj}_{\boldsymbol{\Omega}}^{\boldsymbol{Q}}(\mathcal{F}(\boldsymbol{x}^*))\|_{\boldsymbol{Q}} \\
&\le \|\mathcal{F}(\boldsymbol{x}^k) - \mathcal{F}(\boldsymbol{x}^*)\|_{\boldsymbol{Q}} \\
&\le \sqrt{\rho_\alpha}\|\boldsymbol{x}^k - \boldsymbol{x}^*\|_{\boldsymbol{Q}},
\end{aligned}$$

where the second equality follows by $\boldsymbol{Q} = Q \otimes I_n$ and the definition of $\boldsymbol{\Omega}$ (note that $\mathrm{proj}_{\Omega_i}^{q_i I_{n_i}} = \mathrm{proj}_{\Omega_i}$), the first inequality follows by nonexpansiveness of the projection [8, Prop. 4.16], and the second inequality by Lemma 6.4. ■

We note that Algorithm 6.1 requires a priori knowledge of the communication graph $\mathcal{G}$, both to compute the PF eigenvector q and to tune the step size α. In the next subsection, we relax this hypothesis.

6.3.2. CASE 2: ONLINE COMPUTATION OF q

When the PF eigenvalue q is not known in advance, it can be computed online in a distributed fashion. The procedure is illustrated in Algorithm 6.2. Each agent $i \in \mathcal{I}$ keeps an extra variable $\hat{q}_i = \mathrm{col}((\hat{q}_{i,j})_{j\in\mathcal{I}})$, which is an estimate of q, initialized as the i-th vector of the canonical basis $\mathbf{e}_i^N \in \mathbb{R}^N$.

Notably, each estimate $\hat{q}_i$ converges to the real value q. In fact, the updates in Algorithm 6.2 can be written compactly as

$$\hat{\boldsymbol{q}}^{k+1} = (W \otimes I_N)\hat{\boldsymbol{q}}^k, \tag{6.14}$$

where $\hat{\boldsymbol{q}} := \mathrm{col}((\hat{q}_i)_{i\in\mathcal{I}})$. Therefore, by the PF theorem (and by Standing Assumptions 6.3-6.4), $\hat{\boldsymbol{q}}^k$ converges linearly to $(\mathbf{1}_N q^\top \otimes I_N)\hat{\boldsymbol{q}}^0 = \mathbf{1}_N \otimes q$. In particular, $\hat{q}_{i,i}^k \to q_i$. Also, $\hat{q}_{i,i}^k > 0$ for all $k \ge 0$, since $\hat{q}_{i,i}^0 > 0$ and W is nonnegative with positive diagonal. As such, Algorithm 6.2 is always well defined. We first show its convergence for a fixed step size.

Theorem 6.2. Let $\alpha > 0$ satisfy the condition in (6.13), and $\alpha^k = \alpha\ \forall k \in \mathbb{N}$. Then, for any initial condition, the sequence $(\boldsymbol{x}^k)_{k\in\mathbb{N}}$ generated by Algorithm 6.2 converges to $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, where x^* is the NE of the game in (6.1), with linear rate: for any $\epsilon > 0$, there exists $K > 0$ such that, for all $k \in \mathbb{N}$,

$$\|\boldsymbol{x}^k - \boldsymbol{x}^*\|_{\boldsymbol{Q}} \le K\left(\sqrt{\rho_\alpha} + \epsilon\right)^k \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|_{\boldsymbol{Q}}.$$ □

While in Algorithm 6.2 the PF eigenvector is estimated online, the upper bound on α in Theorem 6.2 is still a function of the network parameter $\bar{\sigma}$, which can be difficult to compute distributedly. Upper/lower bounds might be available for some classes of networks, e.g., unweighted graphs. This is analogous to [143, Th. 2], where q is computed online, but the step size depends on global, not easily accessible, information. In fact, this notion of fixed but small-enough step sizes is not uncommon in distributed algorithms literature.

When estimating a step α that satisfies (6.13) is impossible, convergence to a NE can still be guaranteed by allowing for diminishing step sizes. In this case, also the information on the game (i.e., Lipschitz and monotonicity constants of the pseudo-gradient) is not needed for the tuning.

Theorem 6.3. Let $(\alpha^k)_{k\in\mathbb{N}}$ be a positive nonincreasing sequence such that $\sum_{k\in\mathbb{N}} \alpha^k = \infty$ and $\lim_{k\to\infty} \alpha^k = 0$. Then, for any initial condition, the sequence $(\boldsymbol{x}^k)_{k\in\mathbb{N}}$ generated by Algorithm 6.2 converges to $\boldsymbol{x}^* = \mathbf{1}_N \otimes x^*$, where x^* is the NE of the game in (6.1). □

6.4. NUMERICAL EXAMPLE: A NASH-COURNOT GAME

We consider the Cournot competition model in [109, §6]. N firms produce an uniform commodity that is sold to m markets. Each firm $i \in \mathcal{I} = \{1,\dots,N\}$ is allowed to participate in $n_i \le m$ of the markets; its decision variable is the vector $x_i \in \mathbb{R}^{n_i}$ of quantities of product to be delivered to each of the n_i markets, bounded by the local constraints $\mathbf{0}_{n_i} \le x_i \le X_i$. Let $A_i \in \mathbb{R}^{m\times n_i}$ such that $[A_i]_{k,j} = 1$ if $[x_i]_j$ is the amount of commodity sent to the k-th market by agent i, $[A_i]_{k,j} = 0$ otherwise, for all $j = 1,\dots,n_i$, $k = 1,...,m$. Hence, $Ax = \sum_{i=1}^N A_i x_i \in \mathbb{R}^m$, where $A := [A_1 \dots A_N]$, are the quantities of product delivered to each market. Firm i aims at maximizing its profit, i.e., minimizing the cost $J_i(x_i, x_{-i}) = c_i(x_i) - p(Ax)^\top A_i x_i$. Here, $c_i(x_i) = x_i^\top Q_i x_i + q_i^\top x_i$ is firm i's production cost, with $Q_i \succ 0$; $p : \mathbb{R}^m \to \mathbb{R}^m$ associates to each market a price that depends on the amount of product delivered to that market. Specifically, for $k = 1,\dots,m$, $[p(x)]_k = \bar{P}_k - \chi_k [Ax]_k$, where $\bar{P}_k, \chi_k > 0$. We set $N = 20$, $m = 7$. The market structure (i.e., which firms participate in each market) is defined as in [109, Fig. 1]. Therefore, $x = \mathrm{col}((x_i)_{i\in\mathcal{I}}) \in \mathbb{R}^n$ and $n = 32$. We select randomly with uniform distribution r_k in $[1,2]$, Q_i diagonal with diagonal entries in $[14,16]$, q_i with elements in $[1,2]$, $\bar{P}_k$ in $[10,20]$, χ_k in $[1,3]$, X_i in $[5,10]$, for all $i \in \mathcal{I}$, $k = 1,\dots,m$. This setup satisfies Standing Assumptions 6.1-6.2 [109, §6]. The firms communicate over a randomly generated strongly connected row stochastic directed network, but cannot access the production of all the competitors. We set $\alpha \approx 3 \times 10^{-5}$ to satisfy the condition in (6.13). We compare the performance of Algorithms 6.1 and Algorithm 6.2, the latter both with a fixed ($\alpha^k = \alpha$) and vanishing step size ($\alpha^k = \frac{1}{k+1}$), in figure 6.1. Due to the small α, the schemes with fixed step are almost indistinguishable, and diminishing step sizes result in faster convergence. The good performance obtained with vanishing step suggests that the choice of α is quite conservative. Indeed, Algorithms 6.1-6.2 still converge, and much faster, with a fixed step size 400 times larger than its theoretical upper bound (dashed lines).

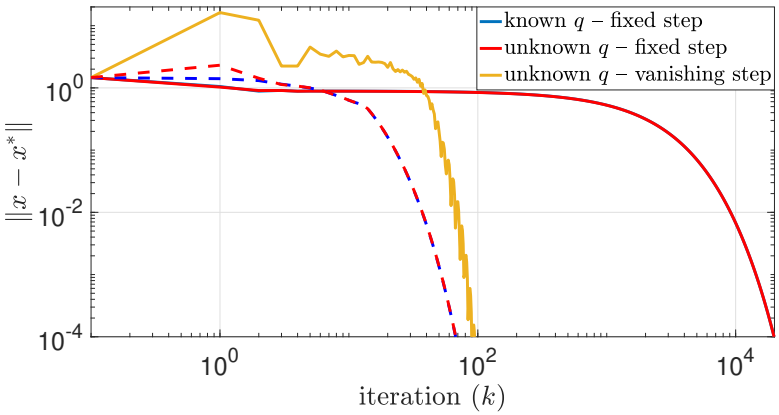

Figure 6.1: Distance from the Nash equilibrium, when the PF eigenvector is known (Algorithm 6.1) or computed online (Algorithm 6.2), with the step sizes chosen to satisfy the theoretical bounds (solid lines) or with a fixed step size chosen 400 times larger than the theoretical upper bound (dashed lines).

6.5. CONCLUSION

CERTAIN properties of doubly stochastic matrices carry on to row stochastic matrices, but in a different Hilbert space, weighted by their left Perron-Frobenius eigenvector. We exploited one such contractivity property to solve, in a fully-distributed way, Nash equilibrium problems over directed networks. Any requirement for global knowledge of the graph and of the game mapping can be avoided in the case of vanishing step sizes.

The extension of our results to generalized games, where the agents share some common constraints, is left as future research. It would be also valuable to relax our connectivity and monotonicity assumptions, namely allowing for jointly connected networks and (strictly) monotone game mappings.

6.6. APPENDIX

6.6.1. PROOF OF LEMMA 6.1

Note that, since $q^\top \mathbf{1}_N = 1$ and $W\mathbf{1}_N = \mathbf{1}_N$, it holds that $W(y - \mathbf{1}_N q^\top y) = W(y - \mathbf{1}_N q^\top y) - \mathbf{1}_N q^\top y + \mathbf{1}_N (q^\top \mathbf{1}_N) q^\top y = (W - \mathbf{1}_N q^\top)(y - \mathbf{1}_N q^\top y)$, and hence

$$\|W(y - \mathbf{1}_N q^\top y)\|_Q \leq \|W - \mathbf{1}_N q^\top\|_Q \|y - \mathbf{1}_N q^\top y\|_Q.$$

Therefore, it suffices to show that $\|W - \mathbf{1}_N q^\top\|_Q = \bar{\sigma} < 1$. Let $p := \mathrm{col}((\sqrt{q_i})_{i \in \mathcal{I}})$. Then,

$$\begin{aligned}
&\|W - \mathbf{1}_N q^\top\|_Q^2 \\
&= \|Q^{\frac{1}{2}}(W - \mathbf{1}_N q^\top) Q^{-\frac{1}{2}}\|^2 \\
&= \lambda_{\max}((Q^{\frac{1}{2}} W Q^{-\frac{1}{2}} - pp^\top)^\top (Q^{\frac{1}{2}} W Q^{-\frac{1}{2}} - pp^\top)) \\
&\overset{(a)}{=} \lambda_{\max}(Q^{-\frac{1}{2}} W^\top Q W Q^{-\frac{1}{2}} - pp^\top) \\
&:= \lambda_{\max}(M - pp^\top),
\end{aligned}$$

where in (a) we used $p^\top p = 1$, and $M = Q^{-\frac{1}{2}} W^\top Q W Q^{-\frac{1}{2}}$. Since M is symmetric and $Mp = p$, M has a basis of eigenvectors, say $\{v_1, \dots, v_{N-1}, p\}$, with associate eigenvalues $\{s_1, \dots, s_{N-1}, 1\}$. By orthogonality and $p^\top p = 1$, it follows that the eigenvalues of $M - pp^\top$ are $\{s_1, \dots, s_{N-1}, 0\}$, with associate eigenvectors $\{v_1, \dots, v_{N-1}, p\}$. Since $M \succeq 0$, it suffices to show that $s_i < 1$, for $i = 1, \dots, N-1$. Seeking a contradiction, let $j \in \{1, \dots, N-1\}$ such that $s_j \geq 1$, and $\bar{v} := Q^{-\frac{1}{2}} v_j$. Thus, we have $\|W\bar{v}\|_Q^2 = v_j^\top Q^{-\frac{1}{2}} W^\top Q W Q^{-\frac{1}{2}} v_j = v_j^\top M v_j \geq v_j^\top v_j = \bar{v} Q \bar{v} = \|\bar{v}\|_Q^2$. By [36, Lemma 1], it also holds, for some $\gamma > 0$, for any $y \in \mathbb{R}^N$, that $\|Wy\|_Q \leq \|y\|_Q - \gamma \|(I_N - W)y\|_Q$. Hence, by Standing Assumption 6.3, it must hold that $\bar{v} = \beta 1_N$, for some $\beta \neq 0$. Equivalently, $v_j = \beta p$. This is a contradiction, since p and v_j must be orthogonal. The conclusion follows with $\bar{\sigma} = \sqrt{\lambda_{N-1}(M)}$. ■

6.6.2. PROOF OF LEMMA 6.3

For any $x, y \in \mathbb{R}^n$ it holds that $\langle \bar{Q}^{-1}(F(x) - F(y)), x - y \rangle_{\bar{Q}} = \langle F(x) - F(y), x - y \rangle \geq \mu \|x - y\|^2 \geq \frac{\mu}{\lambda_{\max}(\bar{Q})} \|x - y\|_{\bar{Q}}^2$, and that $\|\bar{Q}^{-1}(F(x) - F(y))\|_{\bar{Q}}^2 = \|F(x) - F(y)\|_{\bar{Q}^{-1}}^2 \leq \frac{\lambda_{\max}(\bar{Q}^{-1})}{\lambda_{\min}(\bar{Q})} \theta_0^2 \|x - y\|_{\bar{Q}}^2$. Analogously, by Lemma 6.2, it holds that, for any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^{Nn}$, $\|\bar{\boldsymbol{Q}}^{-1}(\boldsymbol{F}(\boldsymbol{x}) - \boldsymbol{F}(\boldsymbol{y}))\|_{\bar{\boldsymbol{Q}}}^2 \leq \frac{\lambda_{\max}(\bar{Q}^{-1})}{\lambda_{\min}(\boldsymbol{Q})} \theta^2 \|\boldsymbol{x} - \boldsymbol{y}\|_{\boldsymbol{Q}}^2$. ■

6.6.3. PROOF OF LEMMA 6.4

We use the shorthand notation $\bar{\boldsymbol{F}}\boldsymbol{x}$ and $\bar{F}x$ in place of $\bar{\boldsymbol{F}}(\boldsymbol{x})$ and $\bar{F}(x)$. Let $\boldsymbol{x} \in \mathbb{R}^{Nn}$, $\boldsymbol{y} = \mathbf{1} \otimes y \in \boldsymbol{E}$, and $\hat{\boldsymbol{x}} := \boldsymbol{W}\boldsymbol{x} = \hat{\boldsymbol{x}}_\parallel + \hat{\boldsymbol{x}}_\perp = \mathbf{1}_N \otimes \hat{x}_\parallel + \hat{\boldsymbol{x}}_\perp \in \mathbb{R}^{Nn}$, with $\hat{\boldsymbol{x}}_\perp \in E_\perp^Q$. Thus, we have

$$
\begin{aligned}
&\|\mathcal{F}(\boldsymbol{x}) - \mathcal{F}(\boldsymbol{y})\|_{\boldsymbol{Q}}^2 \\
&= \|(\hat{\boldsymbol{x}} - \alpha \mathcal{R}^\top \bar{\boldsymbol{F}}\hat{\boldsymbol{x}}) - (\boldsymbol{y} - \alpha \mathcal{R}^\top \bar{\boldsymbol{F}}\boldsymbol{y})\|_{\boldsymbol{Q}}^2 \\
&= \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}}^2 + \|\hat{\boldsymbol{x}}_\perp\|_{\boldsymbol{Q}}^2 + \alpha^2 \|\mathcal{R}^\top(\bar{\boldsymbol{F}}\hat{\boldsymbol{x}} - \bar{\boldsymbol{F}}\boldsymbol{y})\|_{\boldsymbol{Q}}^2 \\
&\quad - 2\alpha \langle \hat{\boldsymbol{x}}_\perp, \mathcal{R}^\top(\bar{\boldsymbol{F}}\hat{\boldsymbol{x}} - \bar{\boldsymbol{F}}\boldsymbol{y}) \rangle_{\boldsymbol{Q}} \\
&\quad - 2\alpha \langle \hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}, \mathcal{R}^\top(\bar{\boldsymbol{F}}\hat{\boldsymbol{x}} - \bar{\boldsymbol{F}}\hat{\boldsymbol{x}}_\parallel) \rangle_{\boldsymbol{Q}} \\
&\quad - 2\alpha \langle \hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}, \mathcal{R}^\top(\bar{\boldsymbol{F}}\hat{\boldsymbol{x}}_\parallel - \bar{\boldsymbol{F}}\boldsymbol{y}) \rangle_{\boldsymbol{Q}} \\
&\leq \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}}^2 + \|\hat{\boldsymbol{x}}_\perp\|_{\boldsymbol{Q}}^2 + \alpha^2 \bar{\ell}^2 (\|\hat{\boldsymbol{x}}_\perp\|_{\boldsymbol{Q}}^2 + \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}}^2) \\
&\quad + 2\alpha\bar{\ell} \|\hat{\boldsymbol{x}}_\perp\| (\|\hat{\boldsymbol{x}}_\perp\|_{\boldsymbol{Q}} + \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}}) \\
&\quad + 2\alpha\bar{\ell} \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}} \|\hat{\boldsymbol{x}}_\perp\|_{\boldsymbol{Q}} - 2\alpha\bar{\mu}\lambda_{\min}(Q) \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}}^2,
\end{aligned}
\tag{6.15}
$$

and to bound the addends in (6.15) we used:

- 3^{rd}, 4^{th}, 5^{th} terms: Lipschitz continuity of $\bar{\boldsymbol{F}}$, the Cauchy-Schwartz inequality, $\|\mathcal{R}^\top v\|_{\boldsymbol{Q}} = \|v\|_{\bar{Q}}$ for any $v \in \mathbb{R}^n$, $\|\hat{\boldsymbol{x}} - \boldsymbol{y}\|_{\boldsymbol{Q}}^2 = \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}}^2 + \|\hat{\boldsymbol{x}}_\perp\|_{\boldsymbol{Q}}^2$ (by orthogonality);
- 7^{th} term: $\langle \hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}, \mathcal{R}^\top(\bar{\boldsymbol{F}}\hat{\boldsymbol{x}}_\parallel - \bar{\boldsymbol{F}}\boldsymbol{y}) \rangle_{\boldsymbol{Q}} = \langle \hat{x}_\parallel - y, \bar{F}\hat{x}_\parallel - \bar{F}y \rangle_{\bar{Q}} \geq \bar{\mu} \|\hat{x}_\parallel - y\|_{\bar{Q}}^2 \geq \bar{\mu}\lambda_{\min}(\bar{Q}) \|\hat{x}_\parallel - y\|^2 = \bar{\mu}\lambda_{\min}(\bar{Q}) \|\hat{\boldsymbol{x}}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}}^2$, and the last equality follows since $\hat{\boldsymbol{x}}_\parallel, \boldsymbol{y} \in \boldsymbol{E}$, $\boldsymbol{Q} = Q \otimes I_n$ and $\mathbf{1}_N^\top q = 1$.

Besides, for every $\boldsymbol{x} = \boldsymbol{x}_\parallel + \boldsymbol{x}_\perp \in \mathbb{R}^{Nn}$, with $\boldsymbol{x}_\parallel \in \boldsymbol{E}$ and $\boldsymbol{x}_\perp \in \boldsymbol{E}_\perp^Q$, it holds that $\hat{\boldsymbol{x}} = \boldsymbol{W}\boldsymbol{x} = \boldsymbol{x}_\parallel + \boldsymbol{W}\boldsymbol{x}_\perp$, where $\boldsymbol{W}\boldsymbol{x}_\perp \in \boldsymbol{E}_\perp^Q$ (since $(q \otimes I_n)^\top \boldsymbol{W}\boldsymbol{x}_\perp = (q \otimes I_n)^\top \boldsymbol{x}_\perp = \boldsymbol{0}_n$, by definition of $\boldsymbol{W}$ and q). Consequently, by Lemma 6.1 and by $\boldsymbol{x}_\perp = (I_{Nn} - \boldsymbol{1}_N q^\top \otimes I_n)\boldsymbol{x}$, we have $\|\hat{\boldsymbol{x}}_\perp\|_{\boldsymbol{Q}} = \|\boldsymbol{W}\boldsymbol{x}_\perp\|_{\boldsymbol{Q}} \le \bar{\sigma}\|\boldsymbol{x}_\perp\|_{\boldsymbol{Q}}$. Therefore, we can finally write

$$\begin{aligned}\|\mathcal{F}(\boldsymbol{x}) - \mathcal{F}(\boldsymbol{y}))\|_{\boldsymbol{Q}}^2 &\le \begin{bmatrix}\|\boldsymbol{x}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}} \\ \|\boldsymbol{x}_\perp\|_{\boldsymbol{Q}}\end{bmatrix}^\top M_\alpha \begin{bmatrix}\|\boldsymbol{x}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}} \\ \|\boldsymbol{x}_\perp\|_{\boldsymbol{Q}}\end{bmatrix} \\ &\le \lambda_{\max}(M_\alpha)(\|\boldsymbol{x}_\parallel - \boldsymbol{y}\|_{\boldsymbol{Q}}^2 + \|\boldsymbol{x}_\perp\|_{\boldsymbol{Q}}^2) \\ &= \lambda_{\max}(M_\alpha)\|\boldsymbol{x} - \boldsymbol{y}\|_{\boldsymbol{Q}}^2. \qquad \blacksquare\end{aligned}$$

6.6.4. PROOF OF THEOREM 6.2

We recast Algorithm 6.2 as

$$\boldsymbol{x}^{k+1} = \mathrm{proj}_{\boldsymbol{\Omega}}(\hat{\mathcal{F}}^k(\boldsymbol{x}^k)),$$

where $\hat{\mathcal{F}}^k(\boldsymbol{x}^k) := \boldsymbol{W}\boldsymbol{x}^k - \alpha\mathcal{R}^\top(\bar{Q} + \tilde{Q}^k)^{-1}\boldsymbol{F}(\boldsymbol{W}\boldsymbol{x}^k)$, and $\tilde{Q}^k = \mathrm{diag}(((\hat{q}_{i,i}^k - q_i)I_{n_i})_{i\in\mathcal{I}})$. We noted in §6.3.2 that $(\bar{Q} + \tilde{Q}^k) = \mathrm{diag}((\hat{q}_{i,i}^k I_{n_i})_{i\in\mathcal{I}}) \succ 0$, for all k; also, $\hat{q}_{i,i}^k - q_i \to 0$, for all $i \in \mathcal{I}$. Intuitively, Theorem 6.2 is based on the fact that $\hat{\mathcal{F}}^k$ approaches $\mathcal{F}$ in (6.11) asymptotically (i.e., when $\tilde{Q}^k \approx \boldsymbol{0}$), hence a contractivity property similar to Lemma 6.4 can be ensured for any big-enough k. Specifically, we note that $(\bar{Q} + \tilde{Q}^k)^{-1} = \bar{Q}^{-1} - (\bar{Q}(\bar{Q} + \tilde{Q}^k))^{-1}\tilde{Q}^k =: \bar{Q}^{-1} - P^k$, since the matrices involved are diagonal. Therefore $\hat{\mathcal{F}}^k(\boldsymbol{x}) = \mathcal{F}(\boldsymbol{x}) + \alpha\tilde{\mathcal{F}}^k(\boldsymbol{W}\boldsymbol{x})$, with $\mathcal{F}$ as in (6.11) and $\tilde{\mathcal{F}}^k(\boldsymbol{x}) := \mathcal{R}^\top P^k \boldsymbol{F}(\boldsymbol{x})$. Analogously to Lemma 6.3, it can be shown that $\tilde{\mathcal{F}}^k$ is $\tilde{\ell}^k$-Lipschitz in $\mathcal{H}_{\boldsymbol{Q}}$, with $\tilde{\ell}^k := \lambda_{\max}(P^k)\theta\sqrt{\lambda_{\max}(Q)/\lambda_{\min}(Q)}$. Then, by Lemma 6.4, $\hat{\mathcal{F}}$ is $(\sqrt{\rho_\alpha} + \alpha\tilde{\ell}^k)$-restricted Lipschitz in $\mathcal{H}_{\boldsymbol{Q}}$ with respect to $\boldsymbol{E}$ (cf. Lemma 6.4). Then, analogously to Theorem 6.1, it holds, for all $k \in \mathbb{N}$, that

$$\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\|_{\boldsymbol{Q}} \le (\sqrt{\rho_\alpha} + \alpha\tilde{\theta}^k)\|\boldsymbol{x}^k - \boldsymbol{x}^*\|_{\boldsymbol{Q}}.$$

We remark that $\tilde{\ell}^k \to 0$, since $\tilde{Q}^k \to \boldsymbol{0}$. Hence, for any $\epsilon > 0$, the conclusion follows with $K = (\prod_{k=1}^{\bar{k}} \max\{\sqrt{\rho_\alpha} + \alpha\tilde{\theta}^k, 1\})(\sqrt{\rho_\alpha} + \epsilon)^{-\bar{k}}$, where $\bar{k} := \max\{k \mid \alpha\tilde{\theta}^k > \epsilon\}$. $\blacksquare$

6.6.5. PROOF OF THEOREM 6.3

Analogously to the proof of Theorem 6.2, for all $k \in \mathbb{N}$, it holds that $\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\|_{\boldsymbol{Q}} \le \delta^k\|\boldsymbol{x}^k - \boldsymbol{x}^*\|_{\boldsymbol{Q}}$, $\delta^k := (\sqrt{\rho_{\alpha^k}} + \alpha^k\tilde{\theta}^k)$, with ρ_{α^k} as in (6.13) and $(\tilde{\theta}^k)_{k\in\mathbb{N}}$ a vanishing nonnegative sequence. The conclusion follows because $\prod_{k=0}^{\infty}\delta^k = 0$, as we show next. By explicit computation of the quantity in (6.13) and Taylor expansion at $\alpha = 0$, it holds, in a neighborhood V_0 of $\alpha = 0$, that $\sqrt{\rho_\alpha} = 1 - \bar{\mu}\lambda_{\min}(Q)\alpha + o(\alpha)$, where $o(\alpha)$ is a series of monomial terms at least quadratic in α. Take $\bar{k}$ such that, for all $k \ge \bar{k}$, $\tilde{\ell}^k \le \theta^* < \bar{\mu}\lambda_{\min}(Q)$ for some θ^*, $\alpha^k \in \boldsymbol{V}_0$ and $\delta^k < 1$ (which is always possible, because $\tilde{\ell}^k \to 0$, $\alpha^k \to 0$ and $\delta^k = 1 - (\bar{\mu}\lambda_{\min}(Q) - \tilde{\ell}^k)\alpha^k + o(\alpha^k)$ if $\alpha^k \in \boldsymbol{V}_0$). Then, $\prod_{k=\bar{k}}^{\infty}\delta^k = 0$ if and only if $\sum_{k=\bar{k}}^{\infty} -\log(\delta^k) = \infty$. In turn, by the asymptotic comparison theorem and by the Taylor expansion at $\alpha^k = 0$, the latter series diverges if the series $\sum_{k=\bar{k}}^{\infty}\alpha^k(\bar{\mu}\lambda_{\min}(Q) - \tilde{\ell}^*)$ diverges, which holds by the assumption on $(\alpha^k)_{k\in\mathbb{N}}$. $\blacksquare$

7

THE END: A FRAMEWORK FOR EFFICIENT DISTRIBUTED EQUILIBRIUM SEEKING

This is the end, my only friend, the end

The Doors

Continuous improvement is better than delayed perfection.

Mark Twain

Multi-agent decision problems are typically solved via distributed algorithms, where the computational burden is partitioned among a group of agents, only allowed to communicate on a peer-to-peer network. To cope with the limited information available, each processor is required to store a copy of certain variables, while agreement among the local copies is enforced via consensus protocols. This structure often leads to redundancy of the information, poor scalability with the network size, communication and memory overhead. In this chapter, we develop a framework for the design and analysis of distributed algorithms, named *Estimation Network Design* (END), to systematically assign local copies only to a subset of the agents, while still ensuring consistency. END unifies and generalizes several existing (application-specific) approaches, and leverages the original sparsity of the problem to improve efficiency and minimize redundancy. We illustrate the flexibility and potential of END for several methods in the context of consensus optimization and game equilibrium seeking.

Parts of this chapter are based on [28].

7.1. INTRODUCTION

LARGE-SCALE problems in machine learning [33], signal processing [57] and decentralized control [114] involve huge volumes of data, often spatially scattered. In this scenario, distributed multi-agent computation – featuring peer-to-peer communication only, without central gathering of the data – is emerging as a fundamental paradigm to enable scalability, privacy preservation and robustness. On the downside, distributed algorithms require storage and transmission of multiple local copies of some variables, a form of redundancy that is absent from centralized processing. For instance, in consensus optimization [103], NE seeking over networks [63] and common fixed point computation [62], each agent keeps and exchanges with its neighbors an estimate of the entire decision variable – and possibly of some other global quantities, as the cost gradient or dual variables. This may results in prohibitive memory and communication requirements, and hinders scalability when the dimension of the estimates grows with the network size.

It is natural to question if this complexity can be mitigated. In particular, is it possible to leverage the structure of a distributed problem to reduce the amount of copies allocated and exchanged in the network? We review the efforts made in this direction for two prominent applications: multi-agent optimization and games under partial-decision information.

Distributed optimization: Modern big data problems are typically *partially separable* [101], i.e., the objective function is the sum of local costs, each depending only on a *limited portion* of the overall optimization vector. Nevertheless, most consensus optimization algorithms entail the agents reaching agreement on the *whole* solution [102], [144] – even in applications where each agent finally discards most of the optimal vector, as in resource allocation and network control [104].

Part of the literature focuses on *partitioned* optimization, where the local cost of each agent only depends on its own "action" (local decision variable) and on the actions of its neighbors over the communication graph [104]. Then, approximating the actions of non-neighboring agents is superfluous [53], [104], [138]. Similarly, in constraint-coupled optimization, storing an estimate of the entire dual variable is not needed if each constraint only involves the actions of an agent and its neighbors [44]. Notably, both cases require that the communication graph matches the *interference* graph (describing the coupling among the agents in the cost or constraints), which allows the distributed implementation of many centralized methods [98]. Remarkably, general (non-partitioned) problems are addressed via dual methods by Mota et al. [98], and later by Alghunaim, Yuan and Sayed [2], [3]: in this approach, each component of the optimization variable is estimated by a suitably chosen cluster of agents, including *all* the agents whose cost depends on that component, but conceivably smaller than the entire network. As a drawback, the dual reformulation is only effective over undirected communication networks.

Nash equilibrium seeking: In NE problems under partial-decision information, each agent can only communicate with some neighbors, although its private cost function depends also on the action of non-neighboring agents. To cope with the lack of knowledge, it is typically assumed that each agent estimates all the competitors' actions [22],

[109], [146] – even if its cost might depend on a much smaller subset of them. This approach raises concerns in terms of efficiency and is impractical for large networks. Scalable solutions are known for the class of aggregative games, where the agents only need to reconstruct an aggregation value (e.g., the average of all the actions) to evaluate their objective functions [82], [108]. While this setup was extensively studied, the problem remains widely open for more general, partially coupled, costs. The only results we are aware of are the algorithms proposed by Salehisadaghiani and Pavel [121], [123]: assuming that the communication network can be freely designed, each agent only needs to keep proxies of the decisions that directly influence its cost. Moreover, memory and communication efficient methods for GNE problems (where the agents are also coupled via shared constraints) are not known in literature.

Contributions: Our work is motivated by the observation that, in distributed applications, the coupling among the agents often exhibits some sparsity. This sparsity *could* and *should* be exploited to design efficient algorithms and reduce the number of repeated variables in the network. While some particular scenarios are addressed via ad-hoc schemes [104], [121], what is missing is a systematic methodology to exploit the specific structure of a given problem – without resorting to a case-by-case convergence analysis.

To fill this gap, we introduce *Estimation Network Design* (END), a framework for the analysis and design of distributed iterations. Specifically, we provide a graph-theoretic language to describe how the estimates of the components of any variable of interest are allocated and combined among the agents in a generic distributed algorithm (in Section 7.2). The notation we develop allows one to seamlessly cope with complex network interaction and with the non-homogeneity of the agents' copies (e.g., the local vectors kept by distinct agents may have different dimensions). Our framework is:

- *Versatile and algorithm-free*: The variables of interest can include any quantity some agents need to reach consensus upon, for instance decision vectors or dual multipliers in variational problems, but also the gradient of a cost or an aggregative function. For this reason, END can be employed in virtually any networked decision problem;
- *General*: END unifies the convergence analysis of standard (sparsity-unaware) algorithms (e.g., [122]) with that of algorithms specifically devised for problems with unique sparsity structure (e.g., [121]);
- *Customizable*: END algorithms can be tailored for specific instances of a problem, by embedding efficiency criteria (e.g., minimal memory allocation, bandwidth constraints), while preserving consistency.

We showcase the flexibility of END by generalizing several distributed algorithms in the literature. In particular:

1. For NE problems over networks, we prove linear convergence of a pseudo-gradient algorithm over directed graphs. Our result improves on existing work by relaxing the assumptions on the communication network [27], and by allowing for much more ductile estimate assignment [121], [123]. Moreover, special cases of our

method recover both the full- and the partial-decision information setup (and a plethora of intermediate scenarios), for which a joint convergence analysis was not available. For GNE problems, we study a novel class of aggregative games, generalizing that considered in [54], and we demonstrate that END can reduce the amount of copies allocated on the network for both the aggregation function and the dual variables (Section 7.3);

2. We present the END version of several popular consensus optimization algorithms. In the case of dual methods, our setup coincides with that studied in [2], [3], [98]; compared to these works, we also consider primal methods and directed, time-varying graphs. In particular, we obtain a gradient-tracking method where each agent only has to approximate a portion of the whole gradient; and a push-sum *distributed gradient descent* (DGD) where each agent only estimates some components of the optimization variable (Section 7.4);

3. We numerically compare END algorithms against their sparsity-unaware counterparts. Our simulations suggests that not only communication and memory overhead are significantly reduced, but even that convergence speed can be improved (Section 7.5).

To improve readability, the proofs are in the chapter appendix. We refer to Appendices A, B, C for the basic notation and mathematical background. In addition, to avoid notation jargons, in this chapter we denote by n_x the dimension of any vector x, and by

$$\mathcal{N}(i) := \{j \mid (j,i) \in \mathcal{E}\}, \quad \overline{\mathcal{N}}(i) := \{j \mid (i,j) \in \mathcal{E}\}$$

the in-neighbors and out-neighbors of agent i over a graph $\mathcal{G} = (\mathcal{I}, \mathcal{E})$.

7

7.2. ESTIMATE NETWORK DESIGN FRAMEWORK

7.2.1. THE END SETUP

We start by introducing a generic information structure, useful both to describe existing distributed algorithms and to design new ones. It is characterized by:

- a set of agents $\mathcal{I} := \{1, 2, \dots, N\}$;
- a given (directed) *communication* network $\mathcal{G}^{\mathrm{C}} = (\mathcal{I}, \mathcal{E}^{\mathrm{C}})$, over which the agents can exchange information: agent i can receive information from agent j if and only if $j \in \mathcal{N}^{\mathrm{C}}(i)$;
- a variable of interest $y \in \mathbb{R}^{n_y}$ partitioned as $y = \mathrm{col}((y_p)_{p\in\mathcal{P}})$, where, for each $p \in \mathcal{P} := \{1, \dots, P\}$, $y_p \in \mathbb{R}^{n_{y_p}}$, and $n_y = \sum_{p\in\mathcal{P}} n_{y_p}$;
- a given bipartite directed *interference* graph $\mathcal{G}^{\mathrm{I}} = (\mathcal{P}, \mathcal{I}, \mathcal{E}^{\mathrm{I}})$, $\mathcal{E}^{\mathrm{I}} \subseteq \mathcal{P} \times \mathcal{I}$, that specifies which components of y are indispensable for each agent: $p \in \mathcal{N}^{\mathrm{I}}(i)$ means that agent i needs (an estimate of) y_p to perform some essential local computation.[1]

[1] For ease of notation, we assume that $\overline{\mathcal{N}}^{\mathrm{I}}(p) \neq \varnothing$ for all $p \in \mathcal{P}$ (i.e., each component of y is indispensable for some agent). We recall that we use the superscript to distinguish between different graphs and the related quantities, e.g., $\overline{\mathcal{N}}^{\mathrm{I}}(p)$ are the out-neighbors of node p over the graph $\mathcal{G}^{\mathrm{I}}$.

The agents may be unable to access the value of the variable of interest y. Instead, each agent keeps an estimate of some (possibly all) the components y_p's, and exchanges its estimates with some neighbors, as specified by:

- a bipartite directed *estimate* graph $\mathcal{G}^{\mathrm{E}} = (\mathcal{P}, \mathcal{I}, \mathcal{E}^{\mathrm{E}})$, $\mathcal{E}^{\mathrm{E}} \subseteq \mathcal{P} \times \mathcal{I}$, that specifies which components of y are estimated by each agent: agent i keeps an estimate $\boldsymbol{y}_{i,p} \in \mathbb{R}^{n_{y_p}}$ of y_p if and only if $p \in \mathcal{N}^{\mathrm{E}}(i)$;
- P directed *design* graphs $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$, with $\mathcal{G}_p^{\mathrm{D}} = (\overline{\mathcal{N}}^{\mathrm{E}}(p), \mathcal{E}_p^{\mathrm{D}})$, to describe how the agents exchange their estimates: agent i can receive $\boldsymbol{y}_{j,p}$ from agent j if and only if $i \in \mathcal{N}_p^{\mathrm{D}}(j)$.[2]

Example 7.1 (Partially separable optimization)**.** Consider the distributed optimization problem

$$\min_{y\in\mathbb{R}^{n_y}} \sum_{i\in\mathcal{I}} f_i(y), \tag{7.1}$$

where $f_i : \mathbb{R}^{n_y} \to \bar{\mathbb{R}}$ is a private cost function of agent i, and the optimization variable is partitioned as $y = \mathrm{col}((y_p)_{p\in\mathcal{P}})$. In several engineering applications, like network control and data ranking [101], each cost function f_i depends only on some of the components of y, as specified by an interference graph $\mathcal{G}^{\mathrm{I}}$: f_i depends on y_p if and only if $p \in \mathcal{N}^{\mathrm{I}}(i) \subseteq \mathcal{P}$. With some abuse of notation, we highlight this fact by writing

$$f_i(y) = f_i((y_p)_{p\in\mathcal{N}^{\mathrm{I}}(i)}). \tag{7.2}$$

The common approach to solve (7.1) over a communication network $\mathcal{G}^{\mathrm{C}}$ is to assign to each agent $i \in \mathcal{I}$ a copy $\tilde{\boldsymbol{y}}_i := \mathrm{col}((\boldsymbol{y}_{i,p})_{p\in\mathcal{P}}) \in \mathbb{R}^{n_y}$ of the whole decision variable and to let the agents exchange their estimates with every neighbor over $\mathcal{G}^{\mathrm{C}}$; in END notation, we write this as[3]

$$\mathcal{E}^{\mathrm{E}} = \mathcal{P} \times \mathcal{I}, \qquad \mathcal{G}_p^{\mathrm{D}} = \mathcal{G}^{\mathrm{C}} \ (\forall\, p \in \mathcal{P}). \tag{7.3}$$

This choice of graphs $\mathcal{G}^{\mathrm{E}}$ and $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$ does not take advantage of the structure in (7.2). In fact, agent i only needs $(y_p)_{p\in\mathcal{N}^{\mathrm{I}}(i)}$ to evaluate (the gradient of) its local cost f_i; storing a copy of the whole vector y could be unnecessary and inefficient – especially if $\mathcal{G}^{\mathrm{I}}$ is sparse and P is large. □

7.2.2. DESIGN

From an algorithm design perspective, the graphs $\mathcal{G}^{\mathrm{C}}$ and $\mathcal{G}^{\mathrm{I}}$ shall be considered fixed a priori and part of the problem formulation; in contrast, the graphs $\mathcal{G}^{\mathrm{E}}$ and $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$ are

[2] Note that the vertices of $\mathcal{G}_p^{\mathrm{D}}$ are $\overline{\mathcal{N}}^{\mathrm{E}}(p)$, namely all the agents that keep an estimate of y_p: in fact, agent i can only receive an estimate of y_p if it is keeping an estimate of y_p. Note that the graph $\mathcal{G}^{\mathrm{E}}$ is uniquely determined by $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$; for this reason, in the following, we often only refer to the design graphs $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$.

[3] In the following, we often refer to (7.3) as the "standard" setup, as it is the most widely studied scenario. With $\mathcal{G}^{\mathrm{c}} = \mathcal{G}_p^{\mathrm{D}}$ we also imply $W^{\mathrm{C}} = W_p^{\mathrm{D}}$.

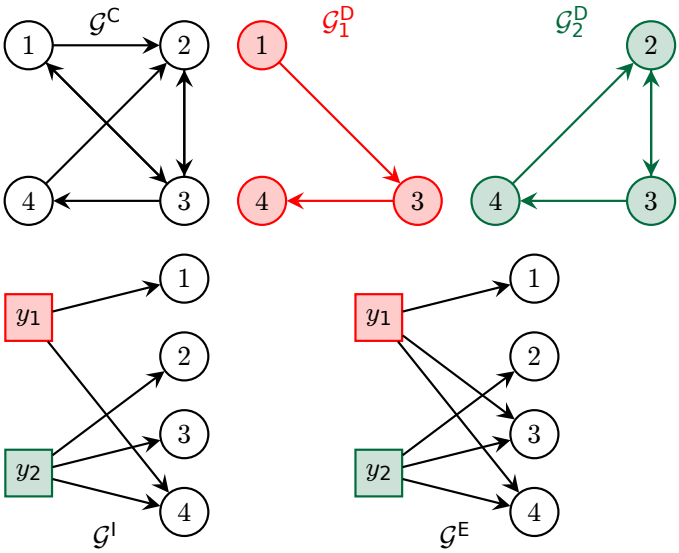

Figure 7.1: An example of END. On the left, the given communication and interference graphs, with $\mathcal{I} = \{1,2,3,4\}$ and $\mathcal{P} = \{1,2\}$. On the right, a possible choice for the estimation graphs $\mathcal{G}_1^{\mathrm{D}}$ and $\mathcal{G}_2^{\mathrm{D}}$, and the resulting estimation graph. $\mathcal{G}_1^{\mathrm{D}}$ is designed to have minimum number of edges, provided that is rooted at 1 (and the conditions in Problem 7.1*(i)* and 7.1*(ii)* are met); the graph is found by solving $\mathrm{UDST}(\mathcal{G}^{\mathrm{C}}, 1, \{1,4\})$ (please see Appendix B.3). Informally speaking, the goal is to minimize the transmission cost for one round of communication, but allowing the information on $\boldsymbol{y}_{1,1}$ to reach agent 4. Note that agent 3 estimates y_1, despite this variable is not indispensable for its local computation (i.e., $1 \notin \mathcal{N}^{\mathrm{I}}(3)$): otherwise, the information on y_1 could not reach agent 4. Instead, $\mathcal{G}_2^{\mathrm{D}}$ is designed to be strongly connected, with minimum number of nodes (i.e., solving $\mathrm{SCSS}(\mathcal{G}^{\mathrm{C}}, \{2,3,4\})$), to minimize the memory allocation for the copies of y_2.

7

design choices. Informally speaking, the goal is to design the graphs $\mathcal{G}^{\mathrm{E}}$ and $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$ so that it is possible to *distributedly* and (possibly) *efficiently* solve a given decision problem. In mathematical terms, this objective translates to imposing extra structure on the estimate and design graphs.

Problem 7.1. Given the communication graph $\mathcal{G}^{\mathrm{C}}$ and the interference graph $\mathcal{G}^{\mathrm{I}}$, design the estimate graph $\mathcal{G}^{\mathrm{E}}$ and the design graphs $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$ such that:

(i) $\mathcal{G}^{\mathrm{I}} \subseteq \mathcal{G}^{\mathrm{E}}$;

(ii) $\mathcal{G}_p^{\mathrm{D}} \subseteq \mathcal{G}^{\mathrm{C}}$, for all $p \in \mathcal{P}$;

(iii) "additional requirements" on $\mathcal{G}^{\mathrm{E}}$, $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$ are met. □

In particular, it must hold that $\mathcal{G}^{\mathrm{E}} \subseteq \mathcal{G}^{\mathrm{I}}$, namely each agent estimates at least the components of y which are indispensable for local computation (in fact, $\mathcal{G}^{\mathrm{I}}$ expresses the minimal information necessary for each agent). Moreover, since data transmission can only happen over the communication graph $\mathcal{G}^{\mathrm{C}}$, it must hold that $\mathcal{G}_p^{\mathrm{D}} \subseteq \mathcal{G}^{\mathrm{C}}$, for all $p \in \mathcal{P}$.

The "additional requirements" in Problem 7.1*(iii)* can encode *feasibility* conditions (for instance, we always need some type of connectedness for the graphs $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$, to en-

sure that the agents can reach consensus on their estimates), but also *efficiency* specifications (treated as soft constraints, e.g, we might aim at reducing the memory allocation by minimizing the overall number $|\mathcal{E}^{\mathrm{E}}|$ of copies employed).

For continuity of presentation, we discuss in Section 7.7.1 several instances of Problem 7.1, solution methods and design choices. A simple example is also illustrated in Figure 7.1.

7.2.3. UNIFIED ANALYSIS

All in all, the choice of the estimate and design graphs is vastly problem-dependent. Luckily, differently from most existing works, we do not need to consider a specific structure. Instead, we simply assume some level of connectedness for the design graphs.

Assumption 7.1 (*Connectedness*)**.** At least one of the following conditions holds:

(i) For each $p \in \mathcal{P}$, there exists $r_p \in \mathcal{I}$ such that $\mathcal{G}_p^{\mathrm{D}}$ is rooted at r_p.

(ii) For each $p \in \mathcal{P}$, $\mathcal{G}_p^{\mathrm{D}}$ is strongly connected.

(iii) For each $p \in \mathcal{P}$, $\mathcal{G}_p^{\mathrm{D}}$ is undirected and connected. □

Throughout the chapter, we also assume that the design graphs are chosen to satisfy the specifications in Problem 7.1*(i)* and 7.1*(ii)*, without further mention.

Standing Assumption 7.1 (*Consistency*)**.** It holds that $\mathcal{G}^{\mathrm{I}} \subseteq \mathcal{G}^{\mathrm{E}}$ and that $\mathcal{G}_p^{\mathrm{D}} \subseteq \mathcal{G}^{\mathrm{C}}$ for all $p \in \mathcal{P}$. □

The conditions above ensure some properties for the estimate exchange, akin to those exploited in the analysis of standard consensus-based algorithms (as exemplified in Lemmas 7.1 and 7.2 below). This simple observation allows us to generalize the convergence analysis of several distributed algorithms to the END framework, allowing for great freedom in the estimates exchange, as we show in Sections 7.3 and 7.4. However, before proceeding, we need to introduce the stacked notation used througout in our analysis.

7.2.4. END NOTATION

For all $p \in \mathcal{P}$, let $N_p := \left|\overline{\mathcal{N}}^{\mathrm{E}}(p)\right|$ be the number of copies of y_p. We define:

$$\boldsymbol{y}_p := \mathrm{col}((\boldsymbol{y}_{i,p})_{i \in \overline{\mathcal{N}}^{\mathrm{E}}(p)}) \in \mathbb{R}^{N_p n_{y_p}}, \quad \forall p \in \mathcal{P}; \tag{7.4}$$

$$\boldsymbol{y} := \mathrm{col}((\boldsymbol{y}_p)_{p \in \mathcal{P}}) \in \mathbb{R}^{n_{\boldsymbol{y}}}, \tag{7.5}$$

where we recall that $\boldsymbol{y}_{i,p}$ is the estimate of the quantity y_p kept by agent i; $n_{\boldsymbol{y}} := \sum_{p \in \mathcal{P}} N_p n_{y_p}$. Note that $\boldsymbol{y}_p$ collects all the copies of y_p, kept by different agents. For each $p \in \mathcal{P}$, for each $i \in \overline{\mathcal{N}}^{\mathrm{E}}(p)$, we denote by

$$i_p := \sum_{j \in \overline{\mathcal{N}}^{\mathrm{E}}(p),\, j \leq i} 1 \tag{7.6}$$

the position of i in the ordered set of nodes $\overline{\mathcal{N}}^{\mathrm{E}}(p)$. For all $i \in \mathcal{N}^{\mathrm{E}}(p)$, we denote by $\mathrm{R}_{i,p} \in \mathbb{R}^{n_{y_p} \times N_p n_{y_p}}$ the matrix that selects $\boldsymbol{y}_{i,p}$ from $\boldsymbol{y}_p$, i.e., $\boldsymbol{y}_{i,p} = \mathrm{R}_{i,p} \boldsymbol{y}_p$. We denote

$\boldsymbol{W}^{\mathrm{D}} := \mathrm{diag}((W_p^{\mathrm{D}} \otimes \mathrm{I}_{n_{yp}})_{p\in\mathcal{P}})$, $\boldsymbol{L}^{\mathrm{D}} := \mathrm{diag}((L_p^{\mathrm{D}} \otimes \mathrm{I}_{n_{yp}})_{p\in\mathcal{P}})$, where we recall that W_p^{D} is the weight matrix of $\mathcal{G}_p^{\mathrm{D}}$, and L_p^{D} is its Laplacian. For all $p \in \mathcal{P}$, let

$$\mathcal{C}_p := \{\boldsymbol{y}_p \in \mathbb{R}^{N_p n_{yp}} \mid \boldsymbol{y}_p = \mathbf{1}_{N_p} \otimes v, v \in \mathbb{R}^{n_{yp}}\}, \tag{7.7}$$

be the consensus space for $\boldsymbol{y}_p$, namely the subspace where all the estimates of y_p are equal. The overall consensus space for the estimates is denoted by $\mathcal{C} := \prod_{p\in\mathcal{P}} \mathcal{C}_p$. Given $y = \mathrm{col}((y_p)_{p\in\mathcal{P}}) \in \mathbb{R}^{n_y}$, we define $\mathcal{C}(y) := \mathrm{col}((\mathbf{1}_{N_p} \otimes y_p)_{p\in\mathcal{P}})$. We denote by $\mathcal{C}_\perp$ the complementary subspace of $\mathcal{C}$; by $\Pi_\parallel := \mathrm{diag}(((\mathbf{1}_{Np}\mathbf{1}_{N_p}^\top \otimes I_{n_{yp}})/N_p)_{p\in\mathcal{P}})$ and $\Pi_\perp := I - \Pi_\parallel$ the projection matrices onto $\mathcal{C}$ and $\mathcal{C}_\perp$, respectively.

Sometimes it is useful to define agent-wise quantities, which we indicate with a tilde. Let

$$\tilde{\boldsymbol{y}}_i := \mathrm{col}((\boldsymbol{y}_{i,p})_{p\in\mathcal{N}^{\mathrm{E}}(i)}), \quad \forall i \in \mathcal{I}; \tag{7.8}$$

$$\tilde{\boldsymbol{y}} := \mathrm{col}((\tilde{\boldsymbol{y}}_i)_{i\in\mathcal{I}}) \in \mathbb{R}^{n_y}, \tag{7.9}$$

where $\tilde{\boldsymbol{y}}_i$ collects all the estimates kept by agent i. Let $\mathrm{P} \in \mathbb{R}^{n_y \times n_y}$ be the permutation matrix such that $\mathrm{P}\boldsymbol{y} = \tilde{\boldsymbol{y}}$; the graph structures corresponding to $\tilde{\boldsymbol{y}}$ can be defined via permutations, e.g., $\tilde{\boldsymbol{W}}^{\mathrm{D}} := \mathrm{P}\boldsymbol{W}^{\mathrm{D}}\mathrm{P}^\top$, $\tilde{\boldsymbol{L}}^{\mathrm{D}} := \mathrm{P}\boldsymbol{L}^{\mathrm{D}}\mathrm{P}^\top$, $\tilde{\mathcal{C}} := \mathrm{P}\mathcal{C}$.

The following lemmas follow by stacking over $p \in \mathcal{P}$ well-known graph-theoretic properties (see Appendix B.3).

Lemma 7.1. Let Assumption 7.1*(i)* hold. Then, $\mathrm{null}(\boldsymbol{L}^{\mathrm{D}}) = \mathcal{C}$. Moreover, $\mathrm{null}(\tilde{\boldsymbol{L}}^{\mathrm{D}}) = \tilde{\mathcal{C}}$. □

Lemma 7.2. Let Assumption 7.1*(ii)* hold and assume that W_p^{D} is balanced, for all $p \in \mathcal{P}$. Then, for any $\boldsymbol{y} \in \mathbb{R}^{n_y}$, $\langle \boldsymbol{y}, \boldsymbol{L}^{\mathrm{D}}\boldsymbol{y}\rangle \geq \frac{\bar{\lambda}}{2}\|(I-\Pi_\parallel)\boldsymbol{y}\|^2$, where $\bar{\lambda} := \min_{p\in\mathcal{P}}\{\lambda_2(L_p^{\mathrm{D}\top} + L_p^{\mathrm{D}})\} > 0$. □

7.3. GENERALIZED NASH EQUILIBRIUM SEEKING

In this section we consider GNE problems. In particular, each agent $i \in \mathcal{I}$ is equipped with a private cost function $J_i(x_i, x_{-i})$, $J_i : \mathbb{R}^{n_{x_i}} \times \mathbb{R}^{n_{x_{-i}}} \to \mathbb{R}$, which depends both on its local action (decision variable) $x_i \in \mathbb{R}^{n_{x_i}}$ and on the actions of the other agents $x_{-i} := \mathrm{col}((x_j)_{j\in\mathcal{I}\setminus\{i\}}) \in \mathbb{R}^{n_{x_{-i}}}$. Each agent chooses its action in a local feasible set $\Omega_i \subseteq \mathbb{R}^{n_{x_i}}$; let $x := \mathrm{col}((x_i)_{i\in\mathcal{I}}) \in \Omega$ be the overall action, with $\Omega := \prod_{i\in\mathcal{I}} \Omega_i \subseteq \mathbb{R}^{n_x}$. The agents' decisions are also coupled via shared constraints: specifically, the overall feasible set is $\mathcal{X} := \Omega \cap \{x \in \mathbb{R}^{n_x} \mid Ax = a\}$, where $A \in \mathbb{R}^{n_\lambda \times n_x}$, $a \in \mathbb{R}^{n_\lambda}$. We call *generalized game* the following set of interdependent optimization problems:

$$(\forall i \in \mathcal{I}) \ \underset{x_i\in\mathbb{R}^{n_{x_i}}}{\text{minimize}} \ J_i(x_i, x_{-i}) \quad \text{s.t.} \quad (x_i, x_{-i}) \in \mathcal{X}. \tag{7.10}$$

The goal is to distributedly compute a GNE, a set of decisions simultaneously solving all the problems in (7.10). Formally, a GNE is an I-tuple $x^\star = \mathrm{col}((x_i^\star)_{i\in\mathcal{I}}) \in \mathcal{X}$ such that, for all $i \in \mathcal{I}$, $J_i(x_i^\star, x_{-i}^\star) \leq \inf_{x_i}\{J_i(x_i, x_{-i}^\star) \mid (x_i, x_{-i}^\star) \in \mathcal{X}\}$. Let us define the *pseudo-gradient* operator $F : \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$,

$$F(x) := \mathrm{col}((\nabla_{x_i} J_i(x_i, x_{-i}))_{i\in\mathcal{I}}). \tag{7.11}$$

We restrict our attention to convex and strongly monotone games; the following are standard conditions for GNE seeking over graphs [82, Asm. 1], [136, Asm. 1-2], [109, Asm. 1-2].

Assumption 7.2. For all $i \in \mathcal{I}$, Ω_i is closed and convex, J_i is continuous and $J_i(\cdot, x_{-i})$ is convex and differentiable for every x_{-i}; $\mathcal{X}$ is non-empty and satisfies Slater's constraint qualification. □

Assumption 7.3. The pseudo-gradient F in (7.11) is μ-strongly monotone and θ-Lipschitz continuous, for some $\mu, \theta > 0$. □

We focus on the partial-decision information scenario, where each agent i only relies on the data received locally from some neighbors over a communication network $\mathcal{G}^{\mathrm{C}}$. To cope with the limited information, the solution usually explored in the literature is to embed each agent with an estimate of the whole vector x [146], [136] (and possibly a copy of a dual variable [109]). Critically, this approach fails to exploit the possible sparsity in the cost and constraint coupling. We remedy in the remainder of this section.

7.3.1. END PSEUDO-GRADIENT DYNAMICS FOR NE SEEKING

We start by considering games without coupling constraints (i.e., $\mathcal{X} = \Omega$): then, the notion of GNE boils down to that of a NE. Under Assumptions 7.2 and 7.3, the game in (7.10) has a unique NE. We describe the cost coupling via an interference graph $\mathcal{G}^{\mathrm{I}} = (\mathcal{I}, \mathcal{I}, \mathcal{E}^{\mathrm{I}})$, where $(p, i) \in \mathcal{E}^{\mathrm{I}}$ if and only if J_i explicitly depends on x_p, for all $i \neq p$, and $(i, i) \in \mathcal{E}^{\mathrm{I}}$ for all $i \in \mathcal{I}$; we also write, with some abuse of notation,

$$J_i((x_p)_{p \in \mathcal{N}^{\mathrm{I}}(i)}) := J_i(x_i, x_{-i}).$$

Hence, we choose the variable of interest for the END framework to be the overall action, i.e. $y = x$; $\mathcal{P} = \mathcal{I}$ and $y_i = x_i$ for all $i \in \mathcal{I}$ (finer partitions are also possible). Each agent i keeps and sends the copies $\{\boldsymbol{y}_{i,p}, p \in \mathcal{N}^{\mathrm{E}}(i)\}$, estimating the actions of a *subset* of the other agents. Since the action x_i is actually a local variable, under the control of agent i, we formally define $\boldsymbol{y}_{i,i} := x_i$ (i.e., agent i's estimate of its own action coincides with the *real* value). We study the following iteration (we recall the notation in Section 7.2.4): each agent $i \in \mathcal{I}$ performs

$$(\forall\, p \in \mathcal{N}^{\mathrm{E}}(i)) \qquad \hat{\boldsymbol{y}}^k_{i,p} := \textstyle\sum_{j \in \mathcal{N}^{\mathrm{D}}_p(i)} [W^{\mathrm{D}}_p]_{i_p, j_p} \boldsymbol{y}^k_{i,p} \tag{7.12a}$$

$$(\forall\, p \in \mathcal{N}^{\mathrm{E}}(i) \backslash \{i\}) \qquad \boldsymbol{y}^{k+1}_{i,p} = \hat{\boldsymbol{y}}^k_{i,p} \tag{7.12b}$$

$$\boldsymbol{y}^{k+1}_{i,i} = \mathrm{proj}_{\Omega_i}\left(\hat{\boldsymbol{y}}^k_{i,i} - \alpha \nabla_{x_i} J_i((\hat{\boldsymbol{y}}^k_{i,p})_{p \in \mathcal{N}^{\mathrm{I}}(i)})\right). \tag{7.12c}$$

In (7.12a), the estimates of the agents are updated according to a consensus protocol, with an extra (projected) gradient step for the own estimate $\boldsymbol{y}_{i,i}$. The algorithm retrieves [25, Alg. 1] if $W^{\mathrm{D}}_i = W^{\mathrm{C}}$ for all $i \in \mathcal{I}$, (i.e., the standard setup in (7.3)). Since $\mathcal{G}^{\mathrm{D}}_i \subseteq \mathcal{G}^{\mathrm{C}}$, the algorithm is distributed. Note that the local gradient $\nabla_{x_i} J_i$ is computed on the local estimates kept by agent i, not on the real action $x^k = \mathrm{col}((\boldsymbol{y}^k_{i,i})_{i \in \mathcal{I}})$. We define the *extended pseudo-gradient mapping*

$$\boldsymbol{F}(\boldsymbol{y}) := \mathrm{col}((\nabla_{x_i} J_i((\boldsymbol{y}_{i,p})_{p \in \mathcal{N}^{\mathrm{I}}(i)}))_{i \in \mathcal{I}}), \tag{7.13}$$

$\mathrm{R} := \mathrm{diag}((\mathrm{R}_{i,i})_{i\in\mathcal{I}})$, with $\mathrm{R}_{i,i}$ as in Section 7.2.4, and $\boldsymbol{\Omega} := \{\boldsymbol{y} \mid \mathrm{R}\boldsymbol{y} \in \Omega\}$. Then, (7.12a) can be written in stacked form in one line:

$$\boldsymbol{y}^{k+1} = \mathrm{proj}_{\boldsymbol{\Omega}}\left(\boldsymbol{W}^{\mathrm{D}}\boldsymbol{y}^k - \alpha \mathrm{R}^\top \boldsymbol{F}(\boldsymbol{W}^{\mathrm{D}}\boldsymbol{y}^k)\right). \tag{7.14}$$

Assumption 7.4. For each $i \in \mathcal{I}$, $\mathcal{G}_i^{\mathrm{D}}$ is rooted at i and $W_i^{\mathrm{D}}\mathbf{1}_{N_i} = \mathbf{1}_{N_i}$; we denote by $q_i^{\mathrm{D}} \in \mathbb{R}^{N_i}$ the unique nonnegative vector such that ${q_i^{\mathrm{D}}}^\top W_i^{\mathrm{D}} = {q_i^{\mathrm{D}}}^\top$, $\mathbf{1}_{N_i}^\top q_i^{\mathrm{D}} = 1$. □

Assumption 7.4 is very mild: rootedness is necessary for the consensus of the estimates; row-stochasticity can be immediately satisfied whenever the agents have access to their own in-degree. One major technical complication – with respect to the usual, more restrictive strongly connectedness assumption – is that the (Perron) eigenvectors q_i^{D}'s might have zero elements. In addition, we require one technical condition.

Assumption 7.5. For all $i \in \mathcal{I}$, there is a matrix $Q_i \succ 0$ such that $\sigma_i := \|W_i^{\mathrm{D}} - \mathbf{1}_{N_i}{q_i^{\mathrm{D}}}^\top\|_{Q_i} < 1$, $[\mathbf{1}_{N_i}^\top Q_i]_{i_i} = 1$, and $\mathbf{1}_{N_i}^\top Q_i W_i^{\mathrm{D}}(I_{N_i} - \mathbf{1}_{N_i}{q_i^{\mathrm{D}}}^\top) = \mathbf{0}_{N_i}^\top$, and either *(i)* Q_i is diagonal, or *(ii)* $\Omega_i = \mathbb{R}^{n_{x_i}}$. □

Remark 7.1. Assumption 7.5*(i)* alone is general enough to comprise all the cases considered in the existing literature:

i. if $\mathcal{G}_i^{\mathrm{D}}$ is strongly connected with self-loops, then Assumption 7.5*(i)* holds with $Q_i = \mathrm{diag}(q_i/[q_i]_{i_i})$ [27, Lem. 1]; in particular, if W_i^{D} is doubly stochastic, $Q_i = I$;

ii. if $\mathcal{G}_i^{\mathrm{D}}$ is the directed star graph (namely, there are all and only the edges from node i to every node in $\overline{\mathcal{N}}^{\mathrm{E}}(i)$), then Assumption 7.5*(i)* holds with $Q_i = I$ (and $\sigma_i = 0$, q_i with only one nonzero element $[q_i]_{i_i} = 1$); note that having this structure for all $i \in \mathcal{I}$ correspond to the classical full-information scenario (i.e., estimates and true values coincide), as detailed below.

Other relevant cases, satisfying Assumption 7.5 but never addressed in literature, are discussed in the remainder of the section. □

Theorem 7.1. Let Assumptions 7.2 to 7.5 hold, and let

$$\Xi := \mathrm{diag}((Q_i \otimes I_{n_{x_i}})_{i\in\mathcal{I}}) \qquad \bar{\sigma} := \max_{i\in\mathcal{I}}\{\sigma_i\}$$
$$\bar{\theta} := \theta\sqrt{\max_{i\in\mathcal{I}}\{[Q_i]_{i_i,i_i}\}/\lambda_{\min}(\Xi)}$$
$$\underline{\gamma} := \sqrt{1/\max_{i\in\mathcal{I}}\{\mathbf{1}^\top Q_i \mathbf{1}\}}, \quad \bar{\gamma} := \sqrt{1/\min_{i\in\mathcal{I}}\{\mathbf{1}^\top Q_i \mathbf{1}\}}$$
$$M_\alpha := \begin{bmatrix} 1 - 2\alpha\mu\underline{\gamma}^2 + \alpha^2\theta^2\bar{\gamma}^2 & \bar{\sigma}(\alpha(\bar{\theta} + \theta\bar{\gamma}) + \alpha^2\bar{\theta}\theta\bar{\gamma}) \\ \bar{\sigma}(\alpha(\bar{\theta} + \theta\bar{\gamma}) + \alpha^2\bar{\theta}\theta\bar{\gamma}) & \bar{\sigma}^2(1 + 2\alpha\bar{\theta} + \alpha^2\bar{\theta}^2) \end{bmatrix}$$

Let $\alpha > 0$ be chosen such that

$$\rho_\alpha = \lambda_{\max}(M_\alpha) < 1. \tag{7.15}$$

Then the sequence $(\boldsymbol{y}^k)_{k\in\mathcal{N}}$ generated by (7.14) converges linearly to $\boldsymbol{y}^\star := \mathcal{C}(x^\star)$, where $x^\star$ is the NE of the game (7.10): for all $k \in \mathcal{N}$,

$$\|\boldsymbol{y}^{k+1} - \boldsymbol{y}^\star\|_\Xi^2 \leq \rho_\alpha \|\boldsymbol{y}^k - \boldsymbol{y}^\star\|_\Xi^2. \qquad \square$$

The condition (7.15) always holds for α small enough (explicit bounds are obtained as in [25]). Let us now highlight some of the novelties of Theorem 7.1:

(a) Consider the standard scenario in (7.3), where agents store and exchange an estimate of the whole x. If $\mathcal{G}^{\mathrm{C}}$ is strongly connected and W^{C} is doubly stochastic, Theorem 7.1 retrieves exactly [25, Th. 1]. If W^{C} is only row stochastic, Theorem 7.1 improves on the results in [27] since (7.14) does not require the knowledge of any Perron eigenvector, but just a small-enough step; this is achieved by using the weight matrix Ξ in the analysis.

(b) To our knowledge, the only other works that consider partial coupling are [121], [123]. The authors propose gossip algorithms where, assuming a lower bound for the strongly connected graph $\mathcal{G}^{\mathrm{C}}$ related to $\mathcal{G}^{\mathrm{I}}$, each agent must only estimate the actions that directly affect its cost.[4] However, this setup requires that the cost of each agent only depends on the actions of its communication neighbors and neighbors' neighbors [123, Lem. 3]. By allowing some agents to estimate a larger subset of actions (if needed), Theorem 7.1 avoids this limitation.

(c) Theorem 7.1 also allows for graphs $\mathcal{G}_i^{\mathrm{D}}$'s that are not strongly connected. For instance, if $\{\mathcal{G}_i^{\mathrm{D}}\}_{i\in\mathcal{I}}$ are all star graphs (i.e., Remark 7.1.*ii*), the action update in (7.14) is

$$x_i^{k+1} = \mathrm{proj}_{\Omega_i}(x_i^k - \alpha \nabla_{x_i} J_i(x^k)) \tag{7.16}$$

which is the standard pseudo-gradient method for the full-information scenario. In particular, when $\mathcal{G}^{\mathrm{E}}$ is complete, Theorem 7.1 retrieves the well-known bound $\alpha < 2\mu/\theta^2$ (since $\bar{\sigma} = 0$, $\underline{\gamma}^2 = \bar{\gamma}^2 = \frac{1}{N}$) [8, Prop. 26.16].

(d) Another (not strongly-connected) case not addressed before is that of a matrix $W_1^{\mathrm{D}} = \begin{bmatrix} 1 & \mathbf{0}_{N_i-1}^\top \\ c & \bar{W} \end{bmatrix}$, $c \in \mathbb{R}_{\geq 0}^{N_i-1}$, representing a leader-follower protocol (with agent 1 as the leader for ease of notation). If $\Omega_1 = \mathbb{R}^{n_{x_i}}$ and Assumption 7.4 holds, it can be checked that Assumption 7.5 is verified with $q_1^{\mathrm{D}} = \mathbf{e}_1$, $Q_1 = \begin{bmatrix} 1+\mathbf{1}^\top X_{22}\mathbf{1} & -\mathbf{1}^\top X_{2,2} \\ -X_{22}\mathbf{1}_{N_i-1} & X_{22} \end{bmatrix}$, where $X = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{1,2}^\top & X_{2,2} \end{bmatrix} \succ 0$ is any matrix such that $\|W_1^{\mathrm{D}} - \mathbf{1}q_1^{\mathrm{D}\top}\|_X < 1$. As a special case, if $[c]_{j_1} = 1$, we have $\hat{y}_{j,1}^k = x_1^k$ and agent j can use the real action x_1 when evaluating its cost. We can also model a scenario in which the exact information on x_1 propagates over $\mathcal{G}_1^{\mathrm{D}}$ but with some delay, by choosing $\mathcal{G}_1^{\mathrm{D}}$ as a directed tree.

[4] This is also achieved in Theorem 7.1 by the choice $\mathcal{G}_i^{\mathrm{D}} = (\overline{\mathcal{N}}^{\mathrm{I}}(i), \mathcal{E}^{\mathrm{C}} \cap (\overline{\mathcal{N}}^{\mathrm{I}}(i) \times \overline{\mathcal{N}}^{\mathrm{I}}(i)))$ if all the resulting $\{\mathcal{G}_i^{\mathrm{D}}\}_{i\in\mathcal{I}}$ are strongly connected, which is a much weaker assumption than [123, Asm. 6].

(e) Each graph $\mathcal{G}_i^{\mathrm{D}}$ can be chosen independently. For instance, one variable x_i might be publicly available (choose $\mathcal{G}_i^{\mathrm{D}}$ as a star graph), while other actions can only be reconstructed via consensus. The convergence of these cases would otherwise require ad-hoc analysis (e.g., to determine bounds on the step sizes).

7.3.2. GNE SEEKING IN AGGREGATIVE GAMES

In this section we also address the presence of coupling constraints. As per standard practice, we only focus on v-GNEs, namely GNEs with identical dual variables, which are computationally tractable and economically more justifiable [55]. Under Assumptions 7.2 and 7.3, there is a unique v-GNE; moreover $x^\star$ is the v-GNE of (7.10) if and only if there exists a dual variables $\lambda^\star \in \mathbb{R}^{n_\lambda}$ satisfying the KKT conditions [57, Th. 4.8]:

$$\mathbf{0}_{n_x} \in F(x^\star) + \mathrm{N}_\Omega(x^\star) + A^\top \lambda^\star, \quad \mathbf{0}_{n_\lambda} = Ax^\star - a. \tag{7.17}$$

The coupling constraints are partitioned in M blocks, i.e., $A = [A_{m,i}]_{m\in\mathcal{M},i\in\mathcal{I}}$, $a = \mathrm{col}((\sum_{i\in\mathcal{I}} a_{m,i})_{m\in\mathcal{M}})$, where, for all $m \in \mathcal{M} \coloneqq \{1,2,\dots,M\}$, $A_{m,i} \in \mathbb{R}^{n_{\lambda_m}\times n_{x_i}}$ and $a_{m,i} \in \mathbb{R}^{n_{\lambda_m}}$ are local data kept by agent i. The coupling constraints sparsity pattern is described by the interference graph $\mathcal{G}^{\mathrm{I},\lambda} = (\mathcal{M},\mathcal{I},\mathcal{E}^{\mathrm{I},\lambda})$, where $(m,i) \in \mathcal{E}^{\mathrm{I},\lambda}$ if agent i is involved in the constraints block indexed by m; in other terms,

$$(\forall (m,i) \notin \mathcal{E}^{\mathrm{I},\lambda}) \quad A_{m,i} = \mathbf{0},\ a_{m,i} = \mathbf{0}, \tag{7.18}$$

and the m-th block constraint can be written as $\sum_{i\in\overline{\mathcal{N}}^{\mathrm{I},\lambda}(m)} A_{m,i}x_i - a_{m,i} = \mathbf{0}$. Correspondingly, we partition the dual variable as $\lambda = \mathrm{col}((\lambda_m)_{m\in\mathcal{M}})$, $\lambda_m \in \mathbb{R}^{n_{\lambda_m}}$.

We further study aggregative games [14], [71], where the cost coupling arises via an aggregation mapping $\sigma : \mathbb{R}^{n_x} \to \mathbb{R}^{n_\sigma}$, so that for all $i \in \mathcal{I}$, $J_i(x_i, x_{-i}) = \bar{J}_i(x_i, \sigma(x))$ for some function $\bar{J}_i$. Let σ be partitioned as $\sigma = \mathrm{col}((\sigma_q)_{q\in\mathcal{Q}})$, $\mathcal{Q} \coloneqq \{1,2,\dots,Q\}$, and let $\mathcal{G}^{\mathrm{I},\sigma} = (\mathcal{Q},\mathcal{I},\mathcal{E}^{\mathrm{I},\sigma})$ be an interference graph such that, for all x (and with the usual overloading)

$$\sigma_q(x) = \sigma_q((x_i)_{i\in\overline{\mathcal{N}}^{\mathrm{I},\sigma}(q)}) \tag{7.19}$$

$$\bar{J}_i(x_i,\sigma(x)) = \bar{J}_i(x_i,(\sigma_q(x))_{q\in\mathcal{N}^{\mathrm{I},\sigma}(i)}), \tag{7.20}$$

namely, $(q,i) \in \mathcal{E}^{\mathrm{I},\sigma}$ whenever *either* $\sigma_q(x)$ explicitly depends on x_i, or $\bar{J}_i(x_i,\sigma)$ explicitly depends on σ_q. Finally, we consider *affine* aggregation functions, so that

$$\sigma_q(x) \coloneqq \textstyle\sum_{i\in\overline{\mathcal{N}}^{\mathrm{I},\sigma}(q)} B_{q,i}x_i + b_{q,i}, \tag{7.21}$$

$$\sigma(x) \ = Bx + b, \tag{7.22}$$

$B_{q,i} \in \mathbb{R}^{n_{\sigma_q}\times n_{x_i}}$, $b_{q,i} \in \mathbb{R}^{n_{\sigma_q}}$ being local data of agent i, $B \coloneqq [B_{q,i}]_{q\in\mathcal{Q},i\in\mathcal{I}}$, $b = \sum_{i\in\mathcal{I}} b_i$, $b_i \coloneqq \mathrm{col}((b_{q,i})_{q\in\mathcal{Q}})$; $B_{q,i} \coloneqq \mathbf{0}$ and $b_{q,i} = \mathbf{0}$ if $q \notin \mathcal{N}^{\mathrm{I},\sigma}(i)$.[5]

When $M = Q = 1$, this setup boils down to standard generalized aggregative games [22], [64], where each agent must estimate the whole dual variable $\lambda \in \mathbb{R}^{n_\lambda}$ and the whole aggregative value $\sigma(x) \in \mathbb{R}^{n_\sigma}$. Instead, our idea is to leverage the possible problem sparsity by assigning to each agent copies of only some of the components of the

[5] This setup includes non-aggregative games as well, via $\sigma(x) = x$.

dual variable and of the aggregation function, as specified by two[6] estimate graphs $\mathcal{G}^{\mathrm{E},\lambda} = (\mathcal{M}, \mathcal{I}, \mathcal{E}^{\mathrm{E},\lambda})$, $\mathcal{G}^{\mathrm{E},\sigma} = (\mathcal{Q}, \mathcal{I}, \mathcal{E}^{\mathrm{E},\sigma})$. We postpone the discussion of a motivating application to Section 7.5. To the best of our knowledge, the partial-coupling in the constraints or aggregation has not been considered in the literature. The only exception is the (non-generalized) game studied in [54], where the cost of each agent is only affected by some of the components of a (specific) aggregation function: nevertheless, this algorithm requires strong conditions on the (undirected) communication network – necessary to allow the choice $\mathcal{G}^{\mathrm{I}} = \mathcal{G}^{\mathrm{E}}$, see Example 7.6 in Section 7.7.1.

Let us define the *extended pseudo-gradient mapping*

$$\tilde{\boldsymbol{F}}(x, \boldsymbol{\sigma}) := \mathrm{col}((\nabla_{x_i} \bar{J}_i(x_i, \tilde{\boldsymbol{\sigma}}_i) + \tilde{B}_i^\top \nabla_{\tilde{\boldsymbol{\sigma}}_i} \bar{J}_i(x_i, \tilde{\boldsymbol{\sigma}}_i))_{i \in \mathcal{I}}), \tag{7.23}$$

where (with the customary overloading) $\bar{J}_i(x_i, \tilde{\boldsymbol{\sigma}}_i) := \bar{J}_i(x_i, (\boldsymbol{\sigma}_{i,q})_{q \in \mathcal{N}^{\mathrm{I},\sigma}(i)})$, $\tilde{B}_i := \mathrm{col}((B_{q,i})_{q \in \mathcal{N}^{\mathrm{E},\sigma}(i)})$. Note that $\tilde{\boldsymbol{F}}$ coincides with the pseudo-gradient mapping when the estimates are exact and at consensus, i.e., $\tilde{\boldsymbol{F}}(x, \mathcal{C}^\sigma(\sigma(x))) = F(x)$. We study the following distributed iteration:

$$x^{k+1} = \mathrm{proj}_\Omega(x^k - \beta(\alpha \tilde{\boldsymbol{F}}(x^k, \boldsymbol{\sigma}^k) + \boldsymbol{B}^\top \boldsymbol{L}^{\mathrm{D},\sigma} \boldsymbol{\sigma}^k + \boldsymbol{A}^\top \boldsymbol{\lambda}^k)) \tag{7.24a}$$

$$\boldsymbol{\sigma}^{k+1} = \boldsymbol{\sigma}^k - \beta \boldsymbol{L}^{\mathrm{D},\sigma} \boldsymbol{\sigma}^k + \boldsymbol{B}(x^{k+1} - x^k) \tag{7.24b}$$

$$\boldsymbol{z}^{k+1} = \boldsymbol{z}^k + \beta \boldsymbol{L}^{\mathrm{D},\lambda} \boldsymbol{\lambda}^k \tag{7.24c}$$

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k - \beta(\boldsymbol{L}^{\mathrm{D},\lambda}(2\boldsymbol{z}^{k+1} - \boldsymbol{z}^k) - \boldsymbol{A}(2x^{k+1} - x^k) + \boldsymbol{a}), \tag{7.24d}$$

where $\alpha > 0$, $\beta > 0$ are step sizes; $\boldsymbol{z} = \mathrm{col}((\boldsymbol{z}_m)_{m \in \mathcal{M}}) \in \mathbb{R}^{n_{\boldsymbol{\lambda}}}$, $\boldsymbol{z}_m = \mathrm{col}((\boldsymbol{z}_{i,m})_{i \in \overline{\mathcal{N}}^{\mathrm{E},\lambda}(m)})$, where, for all $m \in \mathcal{N}^{\mathrm{E},\lambda}(i)$, $\boldsymbol{z}_{i,m} \in \mathbb{R}^{n_{\lambda m}}$ is an auxiliary dual variable kept by agent i; $\boldsymbol{A} := P^{\lambda\top} \mathrm{diag}((\tilde{A}_i)_{i \in \mathcal{I}})$, where $\tilde{A}_i := \mathrm{col}((A_{m,i})_{m \in \mathcal{N}^{\mathrm{E},\sigma}(i)})$ (we recall that P^λ is the permutation matrix such that $P^{\lambda\top} \tilde{\boldsymbol{\lambda}} = \boldsymbol{\lambda}$); $\boldsymbol{a} := P^{\lambda\top} \mathrm{col}((\tilde{a}_i)_{i \in \mathcal{I}})$ with $\tilde{a}_i = \mathrm{col}((a_{m,i})_{m \in \mathcal{N}^{\mathrm{E},\lambda}(i)})$; $\boldsymbol{B} := \mathrm{diag}((N_q^\sigma I_{n_{\boldsymbol{\sigma}_q}})_{q \in \mathcal{Q}}) P^{\sigma\top} \mathrm{diag}((\tilde{B}_i)_{i \in \mathcal{I}})$ (we recall that $N_q^\sigma = |\overline{\mathcal{N}}^{\mathrm{E},\sigma}(q)|$). Moreover, let $\boldsymbol{b} := \mathrm{diag}((N_q^\sigma I_{n_{\boldsymbol{\sigma}_q}})_{q \in \mathcal{Q}}) P^{\sigma\top} \mathrm{col}((\tilde{b}_i)_{i \in \mathcal{I}})$ with $\tilde{b}_i := \mathrm{col}((b_{q,i})_{q \in \mathcal{N}^{\mathrm{E},\sigma}(i)})$; we impose

$$\boldsymbol{\sigma}^0 = \boldsymbol{B} x^0 + \boldsymbol{b}, \tag{7.25}$$

or, agent-wise, $\tilde{\boldsymbol{\sigma}}_i^0 = \mathrm{col}((N_q^\sigma B_{q,i} x_i^0 + N_q^\sigma b_{q,i})_{q \in \overline{\mathcal{N}}^{\mathrm{E},\sigma}(i)})$. The algorithm is based on primal-dual pseudo-gradient dynamics, where the update in (7.24b) represents a dynamic tracking of the aggregation function σ, over the graphs $\{\mathcal{G}_q^{\mathrm{D},\sigma}\}_{q \in \mathcal{Q}}$. It is inspired by the methods in [64], [109], and in fact as in these works it is derived as a FB method [8, §26.5].

Assumption 7.6. For each $m \in \mathcal{M}$, $\mathcal{G}_m^{\mathrm{D},\lambda}$ is undirected and connected and $W_m^{\mathrm{D},\lambda}$ is symmetric. For each $q \in \mathcal{Q}$, $\mathcal{G}_q^{\mathrm{D},\sigma}$ is strongly connected and $W_q^{\mathrm{D},\sigma}$ is balanced. □

[6] Here we consider END problem for two variables of interest λ and σ. The notation in Section 7.2 is recovered by setting $y = \mathrm{col}(\lambda, \sigma(x))$, with $P = M + Q$. However, for brevity, here we treat the two variables separately, as two independent instances of END, where the corresponding quantities in Section 7.4.1 are distinguished via the superscripts λ and σ (e.g., $\mathcal{G}_m^{\mathrm{D},\lambda}$ describes how the agents exchange the estimates of λ_m). We also indicate the estimates of λ_m and $\sigma_q(x)$ kept by agent i with $\boldsymbol{\lambda}_{i,m}$ and $\boldsymbol{\sigma}_{i,q}$ (in place of $\boldsymbol{y}_{i,m}^\lambda$ and $\boldsymbol{y}_{i,q}^\sigma$), and analogously for the stacked vectors — e.g., $\boldsymbol{\sigma}_q = \mathrm{col}((\boldsymbol{\sigma}_{i,q})_{i \in \overline{\mathcal{N}}^{\mathrm{E},\sigma}(q)})$, $\boldsymbol{\sigma} = \mathrm{col}((\boldsymbol{\sigma}_q)_{q \in \mathcal{Q}})$.

Theorem 7.2. Let Assumptions 7.2, 7.3 and 7.6 hold, and assume that $\tilde{\boldsymbol{F}}(x,\boldsymbol{\sigma})$ is θ-Lipschitz continuous. Then, for any small-enough $\alpha > 0$ there is a small-enough $\beta > 0$ such that the sequence $(x^k,\boldsymbol{\sigma}^k,\boldsymbol{z}^k,\boldsymbol{y}^k)_{k\in\mathcal{N}}$ generated by (7.24) converges to a point $(x^\star,\mathcal{C}^\sigma(\sigma(x^\star)),\boldsymbol{z}^\star,\mathcal{C}^\lambda(\lambda^\star))$, where $(x^\star,\lambda^\star)$ satisfies the KKT conditions in (7.17), hence $x^\star$ is the v-GNE of the game in (7.10). □

7.4. DISTRIBUTED OPTIMIZATION ALGORITHMS

IN this section, we leverage the END framework to extend several distributed optimization algorithms and deal with partial coupling. In particular, we consider again the cost-coupled problem in Example 7.1, hereby recalled:

$$\min_{y\in\mathbb{R}^{n_y}} f(y) := \textstyle\sum_{i\in\mathcal{I}} f_i((y_p)_{p\in\mathcal{N}^{\mathrm{I}}(i)}), \tag{7.26}$$

where f_i is a private cost function of agent i. We denote by $\mathcal{Y}^\star$ the solution set of (7.26), assumed to be nonempty. We take the optimization variable $y = \mathrm{col}((y_p)_{p\in\mathcal{P}})$ as the variable of interest in the END; with the usual abuse of notation, let us write $f_i((\boldsymbol{y}_{i,p})_{p\in\mathcal{N}^{\mathrm{E}}(i)}) = f_i((\boldsymbol{y}_{i,p})_{p\in\mathcal{N}^{\mathrm{I}}(i)})$.

7.4.1. END DUAL METHODS

Under Standing Assumption 7.1, we can recast (7.26) by introducing local estimates and consensus constraints.The reformulation in the following proposition is not novel, and in fact it was employed for the dual methods in [2], [3], [98].

Proposition 7.1. Let Assumption 7.1*(iii)* hold. Then, problem (7.26) is equivalent to:

$$\begin{cases} \min\limits_{\tilde{\boldsymbol{y}}\in\mathbb{R}^{n_y}} \ \textstyle\sum_{i\in\mathcal{I}} f_i(\tilde{\boldsymbol{y}}_i) \\ \text{s.t.} \quad \boldsymbol{y}_{i,p} = \boldsymbol{y}_{j,p} \quad \forall p\in\mathcal{P}, \forall (i,j)\in\mathcal{E}_p^{\mathrm{D}}. \end{cases} \tag{7.27}$$

□

If $\mathcal{G}_p^{\mathrm{D}} = \mathcal{G}^{\mathrm{C}}$ for all $p\in\mathcal{P}$, then (7.27) recovers the formulation used in the derivation of standard dual methods [7], [33], [140]: these algorithms require each agent to store a copy of the whole optimization vector. Instead, choosing a sparse $\mathcal{G}^{\mathrm{E}}$ can conveniently reduces the number of variables and constraints in (7.27). Regardless, due to its structure (i.e., separable costs and coupling constraints compliant with the communication graph), the problem in (7.27) can be immediately solved via several established Lagrangian-based algorithms (provided that the functions f_i's are sufficiently well-behaved). In practice, this allows one to extend most (virtually all) the existing dual methods to the END framework.

Example 7.2 (*END ADMM*)**.** Let Assumption 7.1*(iii)* hold, and assume that f_i is proper closed convex, for all $i\in\mathcal{I}$. Then, applying the *alternating direction method of multipliers* (ADMM) in [7] to the problem in (7.27)[7] results in the iteration

$$\tilde{\boldsymbol{y}}_i^{k+1} = \operatorname*{argmin}_{\tilde{\boldsymbol{y}}_i} \Big\{ f_i(\tilde{\boldsymbol{y}}_i) + \textstyle\sum_{p\in\mathcal{N}^{\mathrm{E}}(i)} \sum_{j\in\mathcal{N}_p^{\mathrm{D}}(i)} \Big(\|\boldsymbol{y}_{i,p}\|^2 - \langle z_{i,j,p}, \boldsymbol{y}_{i,p}\rangle \Big) \Big\} \tag{7.28a}$$

[7] After decoupling the constraints in (7.27) by introducing auxiliary bridge variables as $\{\boldsymbol{y}_{i,p} = h_{(i,j),p}, h_{(i,j),p} = h_{(j,i),p}, h_{(j,i),p} = \boldsymbol{y}_{j,p}\}$; the approach is standard and we refer to [7] for a complete derivation.

$$z_{i,j,p}^{k+1} = (1-\alpha) z_{i,j,p}^{k} - \alpha z_{j,i,p}^{k} + 2\alpha \boldsymbol{y}_{j,p}^{k+1}, \tag{7.28b}$$

where $z_{i,j,p}$ is an auxiliary variable kept by agent i, for each $i \in \mathcal{I}$, $p \in \mathcal{N}_i^{\mathrm{E}}$, $j \in \mathcal{N}_p^{\mathrm{D}}(i)$. Then, for any $\alpha \in (0,1)$, $\boldsymbol{y}_{i,p}$ converges to $y_p^\star$, where $y^\star = \mathrm{col}((y_p^\star)_{p\in\mathcal{P}})$ is a solution of (7.26), for all $i \in \mathcal{I}$ and $p \in \mathcal{N}_i^{\mathrm{E}}$. Note that performing the update (7.28b) requires agent i to receive data from its neighbor $j \in \mathcal{N}_p^{\mathrm{D}}(i)$ (while (7.28b) requires no communication). If $\mathcal{G}_p^{\mathrm{D}} = \mathcal{G}^{\mathrm{C}}$ for all $p \in \mathcal{P}$, then the method retrieves the standard ADMM for consensus optimization [7, Eq. (13)]. However, in general (7.28) requires the agents to store and exchange with their neighbors less (auxiliary) variables. □

The edge constraints in (7.27) can be replaced by equivalent node conditions (as $\boldsymbol{L}^{\mathrm{D}}\boldsymbol{y} = \boldsymbol{0}$). Furthermore, Proposition 7.1 would hold under the weaker Assumption 7.1(*i*), but distributed algorithms to efficiently solve (7.27) typically require undirected communication.

7.4.2. END ABC ALGORITHM

In this subsection, we propose an END version of the ABC algorithm, recently developed in [144]. For differentiable costs f_i's, let us consider the iteration: $(\forall\, i \in \mathcal{I})(\forall\, p \in \mathcal{N}_i^{\mathrm{E}})$

$$\boldsymbol{y}_{i,p}^{k+1} = -\boldsymbol{z}_{i,p}^{k} + \textstyle\sum_{j\in\overline{\mathcal{N}}_p^{\mathrm{E}}} \left([A_p]_{i_p,j_p} \boldsymbol{y}_{j,p}^{k} - \gamma [B_p]_{i_p,j_p} \nabla_{y_p} f_j(\tilde{\boldsymbol{y}}_j^k) \right) \tag{7.29a}$$

$$\boldsymbol{z}_{i,p}^{k+1} = \boldsymbol{z}_{i,p}^{k} + \textstyle\sum_{j\in\overline{\mathcal{N}}_p^{\mathrm{E}}} [C_p]_{i_p,j_p} \boldsymbol{y}_{j,p}^{k+1}, \tag{7.29b}$$

where $\boldsymbol{z}_{i,p} \in \mathbb{R}^{n_{yp}}$ is a local variable kept by agent i; for all $p \in \mathcal{P}$, A_p, B_p, C_p are matrices in $\mathbb{R}^{N_p \times N_p}$; $\gamma > 0$ is a step size (and we recall the definitions of i_p, j_p in Section 7.2.4). Note that if the matrices A_p, B_p, C_p's are compliant with the corresponding graphs $\mathcal{G}_p^{\mathrm{D}}$'s (e.g., $A_p = B_p = C_p = W_p^{\mathrm{D}}$), then the iteration (7.29) is distributed. We can rewrite (7.29) in stacked form as

$$\boldsymbol{y}^{k+1} = A\boldsymbol{y}^k - \gamma B \nabla_{\boldsymbol{y}} \boldsymbol{f}(\boldsymbol{y}^k) - \boldsymbol{z}^k \tag{7.30a}$$

$$\boldsymbol{z}^{k+1} = \boldsymbol{z}^k + C\boldsymbol{y}^{k+1}, \tag{7.30b}$$

where $A := \mathrm{diag}((A_p \otimes I_{n_{yp}})_{p\in\mathcal{P}})$, $B := \mathrm{diag}((B_p \otimes I_{n_{yp}})_{p\in\mathcal{P}})$ and $C := \mathrm{diag}((C_p \otimes I_{n_{yp}})_{p\in\mathcal{P}})$ belong to $\mathbb{R}^{n_y \times n_y}$, $\boldsymbol{z} := \mathrm{col}((\boldsymbol{z}_p)_{p\in\mathcal{P}})$ with $\boldsymbol{z}_p := \mathrm{col}((\boldsymbol{z}_{i,p})_{i\in\overline{\mathcal{N}}_p^{\mathrm{E}}})$, and $\boldsymbol{f}(\boldsymbol{y}) := \sum_{i\in\mathcal{I}} f_i(\tilde{\boldsymbol{y}}_i)$.

Theorem 7.3. Let $D := \mathrm{diag}((D_p \otimes I_{n_{yp}})_{p\in\mathcal{P}})$, for some $\{D_p \in \mathbb{R}^{N_p \times N_p}\}_{p\in\mathcal{P}}$. Assume that f_i is L-smooth and convex for each $i \in \mathcal{I}$, and that:

(i) $A = BD$ and $B \succcurlyeq 0$, $D \succ 0$;

(ii) $(\forall\, \boldsymbol{y} \in \mathcal{C})\ D\boldsymbol{y} = \boldsymbol{y}$, $B\boldsymbol{y} = \boldsymbol{y}$;

(iii) $C \succcurlyeq 0$, $\mathrm{null}(C) = \mathcal{C}$;

(iv) B and C commute: $BC = CB$;

(v) $I - \frac{1}{2}C - \sqrt{B}D\sqrt{B} \succcurlyeq 0$.

Let $y^\star \in \mathcal{Y}^\star$, $\boldsymbol{y}^\star := \mathcal{C}(y^\star)$, and consider the merit function

$$\mathfrak{M}(\boldsymbol{y}) := \max\{\|\Pi_\perp \boldsymbol{y}\|\|\nabla_{\boldsymbol{y}} \boldsymbol{f}(\boldsymbol{y}^\star)\|, |\boldsymbol{f}(\boldsymbol{y}) - \boldsymbol{f}(\boldsymbol{y}^\star)|\}. \tag{7.31}$$

Then, for any $\boldsymbol{y}^0 \in \mathbb{R}^{n_y}$, $\boldsymbol{z}^0 = \mathbf{0}_{n_y}$, $\gamma \in (\frac{\lambda_{\min}(D)}{L})$, the sequence $(\boldsymbol{y}^k)_{k\in\mathcal{N}}$ generated by (7.30) satisfies

$$\mathfrak{M}\left(\boldsymbol{y}^k_{\text{avg}}\right) = O\left(\tfrac{1}{k}\right), \tag{7.32}$$

for all $k \in \mathcal{N}$, where $\boldsymbol{y}^k_{\text{avg}} := \frac{1}{k}\sum_{t=1}^k \boldsymbol{y}^t$. □

It is shown in [144] that many celebrated schemes for consensus optimization can be retrieved as particular instances of the ABC algorithm, by suitably choosing the matrices A, B, C [144, Tab. 2]: EXTRA [127], NEXT [94], DIGing [103], NIDS [92], and others. Theorem 7.3 permits the extension of each of these methods to the END framework. We only discuss an example below; for the other schemes, the analysis can be carried out analogously, see also [144, §III.A].

Example 7.3 (*END AugDGM*)**.** The following gradient-tracking algorithm is the END version of [145, Alg. 1]: $(\forall i \in \mathcal{I})(p \in \mathcal{N}^{\text{E}}(i))$

$$\begin{aligned}
\boldsymbol{y}^{k+1}_{i,p} &= \sum_{j\in\mathcal{N}^{\text{D}}_p(i)} [W^{\text{D}}_p]_{i_p,j_p}(\boldsymbol{y}^k_{j,p} - \gamma \boldsymbol{v}^k_{j,p}) \\
\boldsymbol{v}^{k+1}_{i,p} &= \sum_{j\in\mathcal{N}^{\text{D}}_p(i)} [W^{\text{D}}_p]_{i_p,j_p}(\boldsymbol{v}^k_{j,p} + \nabla_{y_p} f_j(\tilde{\boldsymbol{y}}^{k+1}) - \nabla_{y_p} f_j(\tilde{\boldsymbol{y}}^k)),
\end{aligned}$$

or, in stacked form,

$$\boldsymbol{y}^{k+1} = \boldsymbol{W}^{\text{D}}(\boldsymbol{y}^k - \gamma \boldsymbol{v}^k) \tag{7.34a}$$

$$\boldsymbol{v}^{k+1} = \boldsymbol{W}^{\text{D}}(\boldsymbol{v}^k + \nabla_{\boldsymbol{y}} \boldsymbol{f}(\boldsymbol{y}^{k+1}) - \nabla_{\boldsymbol{y}} \boldsymbol{f}(\boldsymbol{y}^k)); \tag{7.34b}$$

we impose $\boldsymbol{y}(0) = \mathbf{0}$, $\boldsymbol{v}(0) = \boldsymbol{W}^{\text{D}}\nabla_{\boldsymbol{y}} \boldsymbol{f}(\boldsymbol{y}^0)$. Here, $\boldsymbol{v}_{i,p}$ represents an estimate of $\nabla_{y_p}\sum_{j\in\mathcal{I}} f_j(y)/N_p$ kept by agent i. Note that agent i only estimates and exchanges the components of the cost gradient (and of of the optimization variable) specified by $\mathcal{N}^{\text{E}}(i)$, instead of the whole vector as in [145, Alg. 1] – the two algorithms coincide only if $W^{\text{D}}_p = W^{\text{C}}$ for all $p \in \mathcal{P}$. By eliminating the $\boldsymbol{v}$ variable in (7.34), we obtain

$$\begin{aligned}
\boldsymbol{y}^{k+2} = {} & 2\boldsymbol{W}^{\text{D}}\boldsymbol{y}^{k+1} - (\boldsymbol{W}^{\text{D}})^2\boldsymbol{y}^k \\
& - \gamma(\boldsymbol{W}^{\text{D}})^2(\nabla_{\boldsymbol{y}} \boldsymbol{f}(\boldsymbol{y}^{k+1}) - \nabla_{\boldsymbol{y}} \boldsymbol{f}(\boldsymbol{y}^k)).
\end{aligned} \tag{7.35}$$

Instead, eliminating $\boldsymbol{z}$ from (7.30) we get

$$\begin{aligned}
\boldsymbol{y}^{k+2} = {} & (I - C + A)\boldsymbol{y}^{k+1} - A\boldsymbol{y}^k \\
& - \gamma B(\nabla_{\boldsymbol{y}} \boldsymbol{f}(\boldsymbol{y}^{k+1}) - \nabla_{\boldsymbol{y}} \boldsymbol{f}(\boldsymbol{y}^k)).
\end{aligned} \tag{7.36}$$

which retrieves (7.35) for $A = B = (\boldsymbol{W}^{\text{D}})^2$, $C = (I - \boldsymbol{W}^{\text{D}})^2$.[8] This choice satisfies the conditions in Theorem 7.3, with $D = I$, if Assumption 7.1*(iii)* holds and if each W^{D}_p is symmetric and doubly stochastic.

[8] In fact, the sequence $(\boldsymbol{y}^k)$ generated by (7.30) coincide with that generated by (7.34) for the given initialization.

Corollary 7.1. Let Assumption 7.1*(iii)* hold; assume that $W_p^{\mathrm{D}}\mathbf{1}_{N_p} = \mathbf{1}_{N_p}$, $W_p^{\mathrm{D}} = W_p^{\mathrm{D}\top}$, for all $p \in \mathcal{P}$, and that f_i is L-smooth and convex, for all $i \in \mathcal{I}$. Then, for any $\gamma \in (0, \frac{1}{L})$ the rate (7.32) holds for (7.34). □

Theorem 7.3 requires a recovery procedure (i.e., (7.32) holds for the running average only), as e.g., in [112], but pointwise convergence could be shown for several special cases of (7.30), see e.g., [145]. We also note that Theorem 7.3 enhances customizability with respect to [144, Th.24], even in the standard scenario (7.3) ($\mathcal{G}_p^{\mathrm{D}} = \mathcal{G}^{\mathrm{C}}$, for all $p \in \mathcal{P}$), by allowing for non-identical blocks A_p's (or B_p's, C_p's).

Remark 7.2 (*Strong convexity and linear convergence*)**.** For certain design choices, the ABC algorithm achieves linear convergence when each function f_i is strongly convex in y [94, Th. 15]. Interestingly, it can be analogously shown that the END ABC (7.30) converges linearly under a weaker assumption, namely strong convexity of each f_i with respect to $\tilde{\boldsymbol{y}}_i$ only. This condition requires $\mathcal{G}^{\mathrm{E}} = \mathcal{G}^{\mathrm{I}}$, which is only a viable choice in some cases (see in Examples 7.5 and 7.6 in Section 7.7.1); future work should investigate the convergence rate of (7.30) when only strong convexity of f_i with respect to $(y_p)_{p\in\mathcal{N}^{\mathrm{I}}(i)}$ (or of f in y, as in [127]) is postulated. □

7.4.3. END PUSH-SUM DGD

Techniques to solve optimization problems over switching or directed graphs also find their counterpart in the END framework. As an example, here we generalize the push-sum subgradient algorithm in [102, Eq. (1)].

Let the agents communicate over a time-varying network $(\mathcal{G}^{\mathrm{C},k})_{k\in\mathcal{N}}$, where $\mathcal{G}^{\mathrm{C},k} = (\mathcal{I}, \mathcal{E}^{\mathrm{C},k})$ represents the communication topology at time-step k. Given a fixed estimate graph $\mathcal{G}^{\mathrm{E}} \supseteq \mathcal{G}^{\mathrm{I}}$, for each $p \in \mathcal{P}$ we consider a time-dependent design graph $(\mathcal{G}_p^{\mathrm{D},k})_{k\in\mathcal{N}}$, $\mathcal{G}_p^{\mathrm{D},k} = (\overline{\mathcal{N}}^{\mathrm{E}}(p), \mathcal{E}_p^{\mathrm{D},k}) \subseteq \mathcal{G}^{\mathrm{C},k}$ (note that the set of nodes is fixed in $\mathcal{G}_p^{\mathrm{D},k}$). For all $i \in \mathcal{I}$ and $p \in \mathcal{N}^{\mathrm{E}}(i)$, agent i performs the following updates:

$$q_{i,p}^{k+1} = \textstyle\sum_{j\in\overline{\mathcal{N}}^{\mathrm{E}}(p)} [W_p^{\mathrm{D},k}]_{i_p,j_p} q_{j,p}^k \tag{7.37a}$$

$$\boldsymbol{w}_{i,p}^{k+1} := \textstyle\sum_{j\in\overline{\mathcal{N}}^{\mathrm{E}}(p)} [W_p^{\mathrm{D},k}]_{i_p,j_p} \boldsymbol{z}_{j,p}^k \tag{7.37b}$$

$$\boldsymbol{g}_{i,p}^{k+1} \in \partial_{y_p} f_i(\tilde{\boldsymbol{y}}_i^{k+1}), \quad \boldsymbol{y}_{i,p}^{k+1} := \frac{\boldsymbol{w}_{i,p}^{k+1}}{q_{i,p}^{k+1}} \tag{7.37c}$$

$$\boldsymbol{z}_{i,p}^{k+1} = \boldsymbol{w}_{i,p}^{k+1} - \gamma^k \boldsymbol{g}_{i,p}^{k+1}, \tag{7.37d}$$

initialized at $\boldsymbol{z}_{i,p}^0 \in \mathbb{R}^{n_{y_p}}$, $q_{i,p}^0 = 1$. With respect to [102, Eq. (1)], agent i keeps one scalar $q_{i,p}$ for each $p \in \mathcal{N}^{\mathrm{E}}(i)$ (instead of one overall), but does not store and exchange the variables $\boldsymbol{z}_{i,p} \in \mathbb{R}^{n_{y_p}}$ for $p \notin \mathcal{N}^{\mathrm{E}}(i)$.

Assumption 7.7. For all $k \in \mathcal{N}$ and $p \in \mathcal{P}$, it holds that:

(i) *Self-loops:* for all $i \in \overline{\mathcal{N}}^{\mathrm{E}}(p)$, $(i,i) \in \mathcal{E}_p^{\mathrm{D},k}$;

(ii) *Column-stochasticity*: $\mathbf{1}_{N_p}^\top W_p^{\mathrm{D},k} = \mathbf{1}_{N_p}^\top$;

(iii) *Finite weights*: $[W_p^{\mathrm{D},k}]_{i_p,j_p} \geq \nu > 0,\ \forall (i,j) \in \mathcal{E}_p^{\mathrm{D},k}$. □

Assumption 7.8. There exists an integer $Q > 0$ such that, for all $p \in \mathcal{P}$, $(\mathcal{G}_p^{\mathrm{D},k})_{k\in\mathcal{N}}$ is Q-strongly connected. □

Example 7.4 (*Design of* $\mathcal{G}_p^{\mathrm{D},k}$)**.** Assume that $\mathcal{G}^{\mathrm{C}} \subseteq \bigcup_{t=kQ}^{(k+1)Q-1} \mathcal{G}^{\mathrm{C},k}$ for all $k \in \mathcal{N}$, where $\mathcal{G}^{\mathrm{C}}$ is a strongly connected graph. Let $(\mathcal{G}_p^{\mathrm{D}})_{p\in\mathcal{P}}$ satisfy Standing Assumption 7.1 and Assumption 7.1(*ii*). Then, Assumption 7.8 holds by choosing $\mathcal{G}_p^{\mathrm{D},k} = \mathcal{G}_p^{\mathrm{D}} \cap \mathcal{G}^{\mathrm{C},k}$, for all $p \in \mathcal{P}$ for all $k \in \mathcal{N}$. □

Theorem 7.4. Let Assumptions 7.7 and 7.8 hold. Assume that, for all $i \in \mathcal{I}$, f_i is convex, and there is $L > 0$ such that, for all $y \in \mathbb{R}^{n_y}$ and $g_i \in \partial_y f_i(y)$, it holds that $\|g_i\| \leq L$. Let $(\gamma^k)_{k\in\mathcal{N}}$ be a positive non-increasing sequence such that $\sum_{k=0}^{\infty} \gamma^k = \infty$, $\sum_{k=0}^{\infty} (\gamma^k)^2 < \infty$. Then, the sequence $(\boldsymbol{y}^k)_{k\in\mathcal{N}}$ generated by (7.37) converges to $\mathcal{C}(y^\star)$, for some $y^\star \in \mathcal{Y}^\star$. □

7.4.4. CONSTRAINT-COUPLED DISTRIBUTED OPTIMIZATION

We consider now a different, constraint-coupled problem:

$$\begin{cases} \min\limits_{x_i\in\mathbb{R}^{n_{x_i}},i\in\mathcal{I}} \ \sum_{i\in\mathcal{I}} f_i(x_i) & \text{(7.38a)} \\ \text{s.t.} \quad \sum_{i\in\overline{\mathcal{N}}^{\mathrm{I}}(p)} A_{p,i}x_i - a_{p,i} = \mathbf{0}, \quad \forall p \in \mathcal{P} & \text{(7.38b)} \end{cases}$$

for a given interference graph $\mathcal{G}^{\mathrm{I}} = (\mathcal{P},\mathcal{I},\mathcal{E}^{\mathrm{I}})$, where f_i and $\{A_{p,i} \in \mathbb{R}^{n_{y_p}\times n_{x_i}}, a_{p,i} \in \mathbb{R}^{n_{y_p}}\}_{p\in\mathcal{N}^{\mathrm{I}}(i)}$ are private data kept by agent i (and the constraints (7.38b) are *not* compliant with the communication graph $\mathcal{G}^{\mathrm{C}}$, namely $\overline{\mathcal{N}}^{\mathrm{I}}(p) \not\subseteq \mathcal{N}^{\mathrm{C}}(i)$ for any i). Existing distributed methods to address (7.38) typically require each agent to store a copy of the entire dual variable (here, also our variable of interest) $y = \mathrm{col}((y_p)_{p\in\mathcal{P}}) \in \mathbb{R}^{n_y}$ (and possibly of other variables in $\mathbb{R}^{n_y}$, e.g., an estimate of the constraint violation) [58], [89]. END primal-dual or dual methods can improve efficiency by exploiting the sparsity of $\mathcal{G}^{\mathrm{I}}$. For instance, (a simplified version of) (7.24) can be directly used to solve (7.38) over undirected graphs.[9] Alternatively, let us consider the dual of (7.38):

$$\max_{y\in\mathbb{R}^{n_y}} \ \sum_{i\in\mathcal{I}} \varphi_i((y_p)_{p\in\mathcal{N}^{\mathrm{I}}(i)}) \tag{7.39}$$

$\varphi_i(y) := \min_{x_i\in\mathbb{R}^{n_{x_i}}} f_i(x_i) + \sum_{p\in\mathcal{N}^{\mathrm{I}}(i)} \langle y_p, A_{p,i}x_i - a_{p,i}\rangle$; note that (7.39) is in the form (7.26). In fact, (7.39) was solved in [3] via the reformulation (7.27); this approach has the disadvantage of requiring undirected communication. Nonetheless, (7.39) can also be solved over directed (time-varying) networks, e.g., via the iteration in (7.37).[10]

[9] In fact, under convexity of the functions f_i's, (7.17) coincides with the optimality conditions for (7.38).

[10] If each f_i is convex with compact domain, where the subgradients of the local dual function φ_i can be computed as $\boldsymbol{g}_{i,p}^k = A_{p,i}x_i^\star(\tilde{\boldsymbol{y}}_i^k) - a_{p,i}$, with $x_i^\star(\tilde{\boldsymbol{y}}_i) \in \operatorname{argmin}_{x_i\in\mathbb{R}^{n_{x_i}}} f_i(x_i) + \sum_{p\in\mathcal{N}^{\mathrm{I}}(i)} \langle \boldsymbol{y}_{i,p}, A_{p,i}x_i - a_{p,i}\rangle$.

7.5. ILLUSTRATIVE APPLICATIONS

IN this section, we compare numerically some END algorithms and their sparsity-unaware counterparts, investigating how performance scales with the problem dimension, connectivity and sparsity.

7.5.1. UNICAST RATE ALLOCATION

We study a bandwidth allocation problem with fixed routing [4], [54], here modeled as a GNE problem – see Figure 7.2 for an illustration. Consider an undirected connected communication network $\mathcal{G}^{\mathrm{C}} = (\mathcal{I}, \mathcal{E}^{\mathrm{C}})$. From each node $i \in \mathcal{I}$, a user sends data with rate $x_i \in [0,1]$, over a path $\mathcal{L}_i$ – i.e., a sequence of consecutive edges over the graph $\mathcal{G}^{\mathrm{C}}$. User i aims to choose x_i to minimize the cost function

$$\begin{aligned} J_i(x_i, x_{-i}) &= \bar{J}_i(x_i, \{\sigma_l\}_{l\in\mathcal{L}_i}) = -U(x_i) + \textstyle\sum_{l\in\mathcal{L}_i} c_l(x_i, \sigma_l) \\ \sigma_l &:= \textstyle\sum_{j|l\in\mathcal{L}_j} x_j, \end{aligned}$$

where U_i is an utility function, σ_l represents the aggregative rate over the link (i.e., edge) l, and c_l is a penalty related to link l (e.g., quantifying the loss of service quality due to congestion [4] or a tax imposed by a network manager [54]). Furthermore, the capacity of each link l is bounded by the coupling constraints $\sigma_l \le a_l$, for some $a_l > 0$. Our objective is to seek a v-GNE of the resulting generalized game, when the users can only communicate over the graph $\mathcal{G}^{\mathrm{C}}$.

Let us relabel the "active" edges $\bigcup_{i\in\mathcal{I}}\{l \in \mathcal{L}_i\}$ (i.e., the edges of $\mathcal{G}^{\mathrm{C}}$ hosting at least one path) as $\{1,2,\dots,P\} =: \mathcal{P}$. Let us define for all $p \in \mathcal{P}$ and $i \in \mathcal{I}$,

$$A_{p,i} = B_{p,i} = \begin{cases} 1, & \text{if } p \in \mathcal{L}_i \text{ (user } i \text{ uses link } p) \\ 0, & \text{otherwise,} \end{cases}$$

the so-called *routing* matrix [4]. With these definitions, the problem retrieves an aggregative game of the type described in Section 7.3.2 (although with *inequality* coupling constraints), with $\mathcal{M} = \mathcal{Q} = \mathcal{P}$. In particular, only the aggregative variables and constraints relative to the links in $\mathcal{L}_i$ directly affect agent i, i.e., $\mathcal{N}^{\mathrm{I},\sigma}(i) = \mathcal{N}^{\mathrm{I},\lambda}(i) = \mathcal{L}_i$.[11]

A similar setup was considered in [54], [121] but modeled as a (non-aggregative) non-generalized game, and further assuming that the graphs $\{\mathcal{G}^{\mathrm{C}}|_{\overline{\mathcal{N}}^{\mathrm{I}}(p)}\}_{p\in\mathcal{P}}$ are connected, cf. Figure 7.2 (see also Example 7.6 in Section 7.7.1). Here, to compute a v-GNE, we employ a modification of Algorithm (7.24), where the right-hand side of (7.24d) is replaced by its positive part, to deal with inequalities coupling constraints.[12] In particular, we fix $W_p^{\mathrm{D},\sigma} = W_p^{\mathrm{D},\lambda} =: W_p^{\mathrm{D}}$ for all $p \in \mathcal{P}$, and we compare the performance of the algorithm for two different choices of the design graphs:

[11] The setup also fits more general cases where from a node i multiple distinct flows $x_{i,1},\dots,x_{1,n_{x_i}}$ are sent over distinct paths $\mathcal{L}_{i,1},\dots,\mathcal{L}_{i,n_{x_i}}$, by either a single user (adjusting the cost function) or by multiple competitive users (considering an augmented communication graph $\mathcal{G}^{\mathrm{C}\prime} \supset \mathcal{G}^{\mathrm{C}}$).

[12] Correctness and convergence of the resulting algorithm can be proven analogously to Section 7.7.2: the argument is standard [22], [109] and omitted.

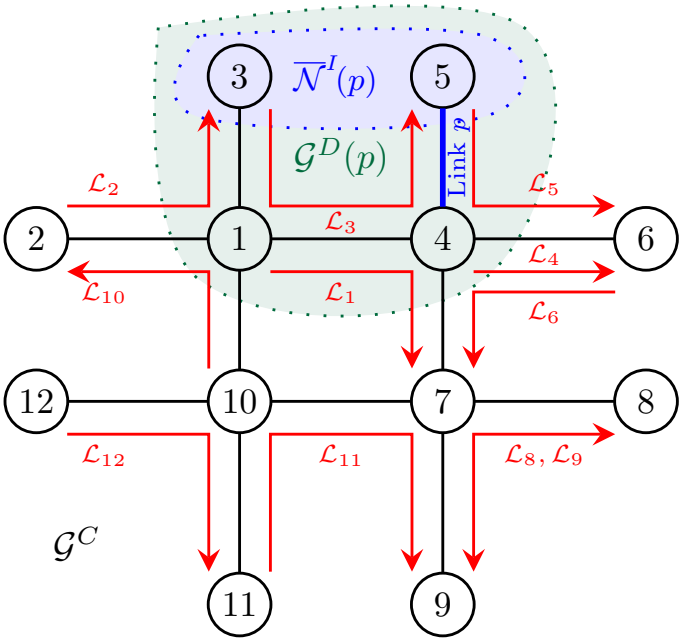

Figure 7.2: Unicast rate allocation game. No flow is sent from node 7, which works only as a communication node (formally, $\mathcal{L}_7$ has length zero). Link (4,5) – labeled p – is used by users 3 and 5 (i.e., $\overline{\mathcal{N}}^{\mathrm{I}}(p) = \{3,5\}$), that are not communication neighbors. Therefore, the graph $\mathcal{G}^{\mathrm{C}}|_{\overline{\mathcal{N}}^{\mathrm{I}}(p)}$ is not connected.

7

- *Standard*: W_p^{D}'s are chosen as in (7.3), i.e., each agent estimates the whole dual variable and aggregative value: the problem sparsity is ignored;
- *Customized*: W_p^{D}'s are chosen to minimize the per-iteration communication cost, by solving Problem 7.1*(i)* with: "*(iii)* $\mathcal{G}_p^{\mathrm{D}}$ is connected; $\mathcal{G}_p^{\mathrm{D}}$ has minimal number of edges" (this corresponds to a Steiner tree problem, see Section 7.7.1 and Appendix B).

We choose $U_i(x_i) = 10 * \log(x_i + 1)$, $c_l(x_i, \sigma_l) = \psi_l \frac{x_i}{1+e^{-\sigma_l}}$ with ψ_l sampled randomly uniformly in the interval $[0,1]$ (Assumption 7.3 is satisfied in the feasible set, invariant for (7.24)), Metropolis-Hastings weights for every graph, $\alpha = 0.1$, $\beta = 10^{-3}$. The results are illustrated in Figure 7.3. In our first experiment, we consider the scenario in Figure 7.2, with $N = P = 12$. The customized algorithm converges to the unique v-GNE $x^\star$ over 10 times faster than the standard version; the communication burden at each step is also reduced, as the mean size of the (aggregative/dual) estimates kept and transmitted by each agent is 2.6, instead of $P = 12$. Since the latter quantity grows with the problem dimension, we expect the gap between the two methods to increase for larger networks. Simulations with $N = 7, 12, 16, 20$ ($P = 7, 16, 18, 26$) confirm this intuition: for the case $N = 20$, the customized algorithm saves 99% of the communication cost (where sending one scalar value to *one* neighbor on $\mathcal{G}^{\mathrm{C}}$ costs 1, in unicast fashion). Finally, for the selected parameters and $N \geq 24$, the standard algorithm fails to converge in our simulations, while the customized algorithm still converges (at least) up to $N = 50$ – suggesting

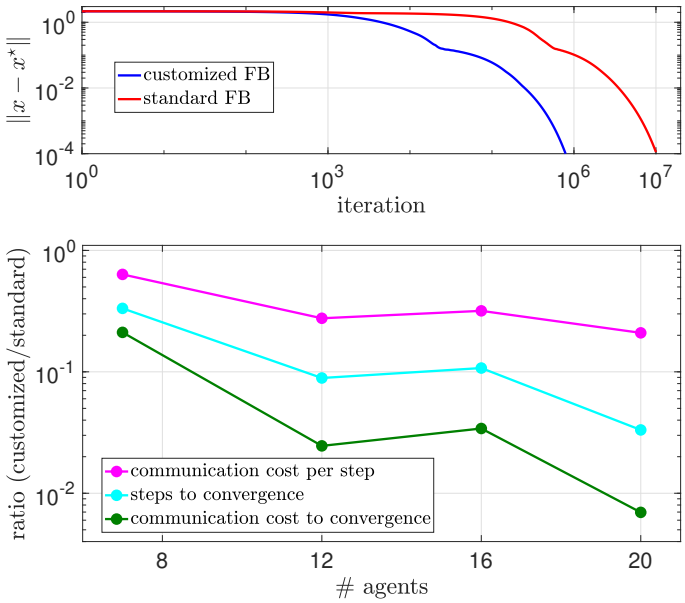

Figure 7.3: Unicast rate allocationi via algorithm (7.24), for the scheme in Figure 7.2 (top) and for different randomly generated networks, with maximal path length 4 and stopping criterion $\|x - x^\star\| <= 10^{-2}$ (bottom).

tolerance to larger step sizes, with upper bounds less affected by the problem dimension.

In conclusion, while requiring some initial design effort, the customized method can result in substantial efficiency improvement, especially if the v-GNE problem is solved multiple times due to time-varying cost parameters or link capacities [4].

7

7.5.2. OPTIMAL ESTIMATION IN WIRELESS SENSOR NETWORKS

We study a regression problem with sparse measurements [2], [101], arising from distributed estimation in wireless and ad-hoc sensor networks. Let us consider some sensors $\{1,2,\dots,N\} =: \mathcal{I}$ and some sources $\{1,2,\dots,P\} =: \mathcal{P}$, spatially distributed on a plane in the square $[0,1] \times [0,1]$. Each source p emits a signal $\bar{y}_p \in \mathbb{R}$, sensed by all the sensors in a radius $r_s > 0$; in turn, each sensor i measures

$$h_i := H_i \operatorname{col}((\bar{y}_p)_{p \in \mathcal{N}^I(i)}) + w_i, \tag{7.40}$$

where $h_i \in \mathbb{R}^{n_{h_i}}$, H_i is a known output matrix, w_i is the measurement noise, and $\mathcal{N}^I(i)$ is the set of sources positioned less than r_s away from sensor i. Sensor i can send information to all the peers in a radius r_c^i (e.g., proportional to the sensor specific power); this induces a directed communication network $\mathcal{G}^C = (\mathcal{I}, \mathcal{E}^C)$ among the sensors, which we assume to be *strongly connected* (see Figure 7.4 for an illustration).

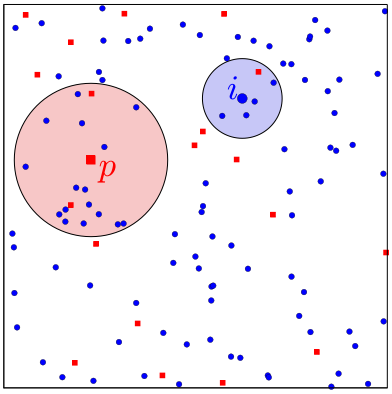

Figure 7.4: Distribution of sources (red) and sensors (blue): the sensors in the red circle receive signal from source p, while all those in the blue circle can receive by (but not necessarily sent to) sensor i.

LINEAR REGRESSION

In our first simulation, the sensors' goal is to collaboratively solve the least square problem

$$\min_{y\in\mathbb{R}^P} \sum_{i\in\mathcal{I}} \left\| h_i - H_i \mathrm{col}((y_p)_{p\in\mathcal{N}^{\mathrm{I}}(i)}) \right\|^2, \tag{7.41}$$

which is in the form (7.26). We seek a solution via algorithm (7.37), comparing the performance for two choices of the design graphs:

- *Standard*: W_p^{D}'s are chosen as in (7.3);
- *Customized*: W_p^{D}'s are chosen by (approximately) solving Problem 7.1*(i)* with: "(iii) $\mathcal{G}_p^{\mathrm{D}} = \mathcal{G}^{\mathrm{C}}|_{\overline{\mathcal{N}}^{\mathrm{E}}(p)}$ is strongly connected and has minimal number of nodes" (solved as a SCSS problem, see Section 7.7.1).

We set $N = 100$, $P = 20$, and randomly generate sensor/sources positions as in Figure 7.4. We choose $r_{\mathrm{s}} = 0.2$, and draw each r_{c}^i uniformly in $[r_{\mathrm{c}}^{\min}, r_{\mathrm{c}}^{\min} + 0.1]$. For all $i \in \mathcal{I}$, we fix $n_{h_i} = 10$, we generate H_i by first uniformly drawing entries in $[0,1]$ and then normalizing the rows to unitary norm, we draw each element of w_i from an unbiased normal distribution with variance 0.1; each signal $\bar{y}_p$ is uniformly randomly chosen in $[0,1]$; the step size is set as $\gamma^k = k^{-0.51}$ in (7.37).[13] The advancement is evaluated via the merit function $\mathfrak{V}(\boldsymbol{y}) := \max\{\|\mathrm{diag}((\frac{1}{N_p} I)_{p\in\mathcal{P}})\Pi_{\perp}\boldsymbol{y}\|\|\nabla_{\boldsymbol{y}}\boldsymbol{f}(\boldsymbol{y}^\star)\|, |\boldsymbol{f}(\Pi_{\parallel}\boldsymbol{y}) - \boldsymbol{f}(\boldsymbol{y}^\star)|\}$, where $\boldsymbol{y}^\star = \mathcal{C}(y^\star)$ and $y^\star$ solves (7.41). Figure 7.5 shows the results for different values of $r_{\mathrm{c}}^{\min}$, where a higher value induces a denser graph $\mathcal{G}^{\mathrm{C}}$. For $r_{\mathrm{c}}^{\min} = 0.1$, the customized method is 15 times faster then the standard one. Increasing $r_{\mathrm{c}}^{\min}$ only marginally reduces the per-iteration communication cost of the customized method. In fact already for $r_{\mathrm{c}}^{\min} = 0.25$, the graph $\mathcal{G}^{\mathrm{C}}|_{\overline{\mathcal{N}}^{\mathrm{I}}(p)}$ is strongly connected for all $p \in \mathcal{P}$, so $\mathcal{G}^{\mathrm{E}} = \mathcal{G}^{\mathrm{I}}$ can be

[13] Although the bounded subgradient assumption in Theorem 7.4 fails, boundedness of the sequences generated by (7.37), and hence convergence, can be established based on coercivity of the cost function.

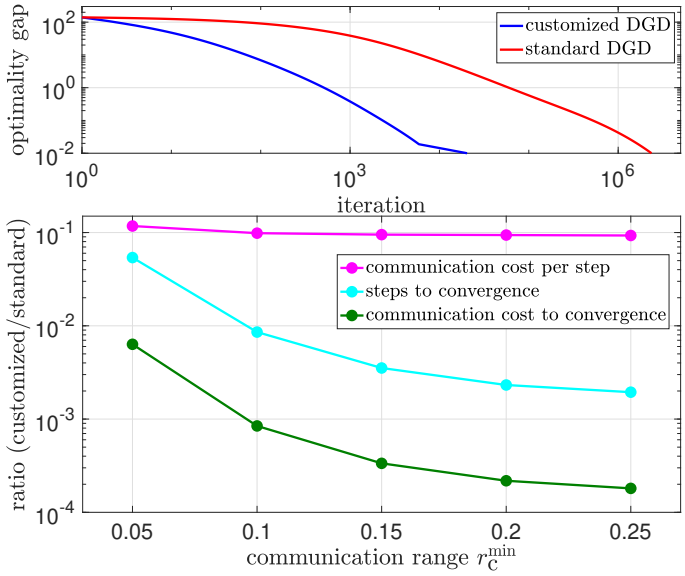

Figure 7.5: Linear regression via algorithm (7.37), for different values of the minimum sensor communication radius $r_c^{\min}$ and stopping criterion $\mathfrak{V}(\boldsymbol{y}) \leq 10^{-2}$ (bottom), and the trajectories obtained with $r_c^{\min} = 0.1$ (top).

chosen: in other terms, the condition in Example 7.6, (Section 7.7.1) are satisfied, similarly to the situation assumed – for *undirected* graphs – in the numerical examples in [2], [3]. In this situation, the customized method achieves a reduction of the communication cost (where sending a variable to *all* the neighbors on $\mathcal{G}^{\mathrm{C}}$ has a cost of 1, in a broadcast fashion) of over 99.9%.

7

LASSO

We assume that only 30% of the sources emits a signal at a given instant (the vector $\bar{y}$ is sparse). The agents collaboratively solve the following problem, regularized to promote sparsity,

$$\min_{y \in \mathbb{R}^P} \|y\|_1 + \sum_{i \in \mathcal{I}} \left\| h_i - H_i \mathrm{col}(y_p)_{p \in \mathcal{N}^{\mathrm{I}}(i)} \right\|^2,$$

where $\|\cdot\|_1$ is the ℓ_1 norm. By defining $f_i((y_p)_{p \in \mathcal{N}^{\mathrm{I}}(i)}) = \|h_i - H_i \mathrm{col}((y_p)_{p \in \mathcal{N}^{\mathrm{I}}(i)})\|^2 + \sum_{p \in \mathcal{N}^{\mathrm{I}}(i)} \frac{1}{|\mathcal{N}^{\mathrm{I}}(p)|} |y_p|$, we retrieve the form (7.26). We set $I = 10$, $P = 20$, $r_c^{\min} = 0.1$, $n_{h_i} = 1$ for all i, generate random positions for the sensors and sources, and choose the other parameters as above, for both the standard and customized methods. Figure 7.6 compares the results for different values of r_s. For larger r_s, the interference graph $\mathcal{G}^{\mathrm{I}}$ is denser, and the gap between customized and standard method decreases: in fact, for $r_s = 0.8$ the two algorithms coincide, as $\mathcal{G}^{\mathrm{I}}$ is complete.

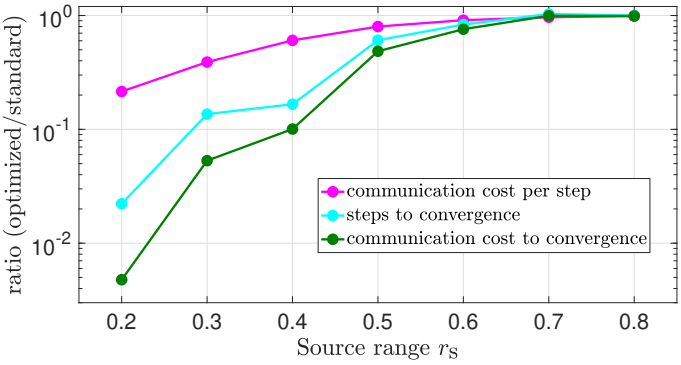

Figure 7.6: LASSO via algorithm (7.37), and different source ranges r_S.

7.6. CONCLUSION

WE presented END, a graph-theoretic language of consensus in distributed iterations. END algorithms allow for unprecedented flexibility in the exchange of variables, while introducing little technical complication in the analysis with respect to their standard counterparts. From a design perspective, END enables exploiting the intrinsic sparsity of a specific problem to improve scalability and reduce communication and memory bottlenecks. Besides the equilibrium seeking and optimization instances we considered, END can find application in virtually any distributed decision problem, for example common fixed point computation [62], multi-cluster games, aggregative optimization [90].

In principle, END algorithms could also be combined with other communication-reduction techniques, such as data sparsification or compression [81], to further enhance efficiency. Future work should focus on computationally efficient and distributed methods to perform the allocation of the estimates; in particular, it would be highly valuable to dynamically assign the estimates online, thus avoiding the need for any a priori design.

7.7. APPENDIX

7.7.1. EXAMPLES OF THE END DESIGN PHASE

In this section, we present some examples of Problem 7.1(*i*) and discuss choices for the design and estimate graphs.

MINIMAL MEMORY ALLOCATION

We consider the problem of minimizing, for each $\mathcal{P}$, the number of copies of y_p, provided that the conditions in Problem 7.1(*i*) and 7.1(*ii*) are satisfied and the graphs $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$ enjoy some connectivity properties. In particular, consider Problem 7.1 with

(iii) for each $p \in \mathcal{P}$, $\mathcal{G}_p^{\mathrm{D}}$ is rooted at $r_p \in \mathcal{I}$ [respectively, $\mathcal{G}_p^{\mathrm{D}}$ is strongly connected / $\mathcal{G}_p^{\mathrm{D}}$ is undirected and connected]; the number $|\overline{\mathcal{N}}^{\mathrm{E}}(p)|$ of nodes in $\mathcal{G}_p^{\mathrm{D}}$ is minimal (provided that all the other specifications are satisfied).

If the problem is feasible, then by definition a solution is given by choosing each[14] $\mathcal{G}_p^{\mathrm{D}}$ as a solution of UDST($\mathcal{G}^{\mathrm{C}}, r_p, \overline{\mathcal{N}}^{\mathrm{I}}(p)$) [respectively, a solution of SCSS($\mathcal{G}^{\mathrm{C}}, \overline{\mathcal{N}}^{\mathrm{I}}(p)$) / a solution of UST($\mathcal{G}^{\mathrm{C}}, \overline{\mathcal{N}}^{\mathrm{I}}(p)$)] (see Appendix B.3 for the definition of these Steiner problems). In fact, this is the design choice suggested in [2], [98] (for the case of undirected graphs). A sufficient condition for the existence of a solution is that the graph $\mathcal{G}^{\mathrm{C}}$ is strongly connected [respectively, the graph $\mathcal{G}^{\mathrm{C}}$ is strongly connected / the graph $\mathcal{G}^{\mathrm{C}}$ is undirected and connected]. A simple example is also illustrated in Figure 7.1.

Note that the solution is generally not unique. For example, let $\mathcal{G}^{\mathrm{UDST}} = (\mathcal{V}^{\mathrm{UDST}}, \mathcal{E}^{\mathrm{UDST}})$ be a solution of UDST($\mathcal{G}^{\mathrm{C}}, r_p, \overline{\mathcal{N}}^{\mathrm{I}}(p)$); then any graph $\mathcal{G}'$ such that $\mathcal{G}^{\mathrm{UDST}} \subseteq \mathcal{G}' \subseteq \mathcal{G}^{\mathrm{C}}|_{\mathcal{V}^{\mathrm{UDST}}}$ is also a solution for Problem 7.1. In simple terms, we can add edges to $\mathcal{G}^{\mathrm{UDST}}$, an extra degree of freedom that can be employed to improve connectivity or robustness to link failures (possibly at the cost of extra communication).

OTHER CRITERIA

One can impose a different connectedness/efficiency specification on each graphs $\mathcal{G}_p^{\mathrm{D}}$ (see Figure 7.1). Concerning the efficiency specifications, instead of minimizing the overall dimension of $\tilde{\boldsymbol{y}}$, it can be convenient to promote allocations where the memory occupation (or the communication requirements) are partitioned equally among the agents: for example, assuming that undirected connected graphs are required, this could be achieved by sequentially designing the graphs $\mathcal{G}_p^{\mathrm{D}}$'s as solutions of a Steiner tree problem, but penalizing unbalanced allocations by opportunely choosing the edge weights for each p. Bandwidth constraints can be addressed similarly, to avoid overloading some channels of the communication network $\mathcal{G}^{\mathrm{C}}$.

DESIGNING THE COMMUNICATION GRAPH

Here, we consider the graph $\mathcal{G}^{\mathrm{C}}$ as given, which is natural for ad-hoc networks or when relying on existing infrastructures. Yet, other works [54], [121], [123] assume that the communication network can be freely designed. In the END framework, this case is addressed by formally assuming that $\mathcal{G}^{\mathrm{C}}$ is complete; then the graphs $\{\mathcal{G}_p^{\mathrm{D}}\}_{p\in\mathcal{P}}$ can be chosen to fulfill some specifications (e.g., minimize the number of active edges in $\mathcal{G}^{\mathrm{C}}$ – which determines the physical channels/edges actually needed).

COMPUTATIONAL COST

Many of the design problems considered in this section can be cast as Steiner problems (see Appendix B), for which off-the-shelf algorithms are available [37], sometimes even distributed [107]. While this is good news, it also reveals that solving Problem 7.1 can be, by itself, computationally demanding[15]. Let us emphasize that Problem 7.1 is still part of the development of an algorithm, to be subsequently used to solve an underlying decision problem. Are the benefits of an efficient estimate allocation – in terms of algorithm execution – worth the additional initial design effort?

[14] Note that in this case Problem 7.1 splits into P independent problems, as Problem 7.1(*ii*) is equivalent to "*(ii)* $\overline{\mathcal{N}}^{\mathrm{I}}(p) \subseteq \overline{\mathcal{N}}^{\mathrm{E}}(p)$, for all $p \in \mathcal{P}$.

[15] The Steiner problems considered are NP-hard, but polynomial algorithms are available to approximate a solution. Note that generally it is not necessary to solve Problem 7.1 to optimality, as long as the approximated solution ensures the feasibility conditions (e.g., connectedness).

If $\mathcal{G}^{\mathrm{I}}$ is dense, it may be convenient to settle for a suboptimal,[16] but readily available, choice – for instance, ignoring the sparsity of $\mathcal{G}^{\mathrm{I}}$, as in (7.3) (which is the standard solution in literature [102], [109]). On the contrary, for *repeated* problems like distributed optimal estimation or model predictive control [98] (where the same function is minimized multiple times, but for different values of some parameters/measurements), a careful a priori design is advantageous, despite the initial (one-time) computational cost of solving Problem 7.1. Finally, in many relevant applications, the specific problem structure renders the choice of optimal graphs $\mathcal{G}_p^{\mathrm{D}}$ straightforward, as exemplified next.

Example 7.5 (Partitioned optimization)**.** Motivated by distributed estimation and resource allocation applications, the works [53], [104], [138] solve optimization problems of the form

$$\min_{y_i \in \mathbb{R}^{n_{y_i}}, i \in \mathcal{I}} \sum_{i \in \mathcal{I}} f_i(y_i, (y_j)_{j \in \mathcal{N}^{\mathrm{C}}(i)}), \tag{7.42}$$

where the cost f_i of agent i depends on its local action y_i and on the actions of its neighbors over the undirected communication network $\mathcal{G}^{\mathrm{C}}$: this is a special case of Example 7.1, with $\mathcal{P} = \mathcal{I}$, $\overline{\mathcal{N}}^{\mathrm{I}}(i) = \mathcal{N}^{\mathrm{C}}(i) \cup \{i\}$. Consider Problem 7.1, with "*(iii)* $\forall i \in \mathcal{I}$, $\mathcal{G}_i^{\mathrm{D}}$ is connected; $\mathcal{G}_i^{\mathrm{D}}$ has minimum number of nodes (provided that all other specifications are met)". A solution is to fix each $\mathcal{G}_i^{\mathrm{D}}$ as the undirected star graph centered in i with vertices $\mathcal{V}_i^{\mathrm{D}} = \overline{\mathcal{N}}^{\mathrm{I}}(i)$: then agent i keeps all and only proxies of the actions that affect its cost. In fact, this is the solution employed in [53], [104], [138]. □

Example 7.6 ($\mathcal{G}^{\mathrm{E}} = \mathcal{G}^{\mathrm{E}}$)**.** With the goal of minimizing the overall memory allocation, consider the choice

$$\mathcal{G}_p^{\mathrm{D}} = \mathcal{G}^{\mathrm{C}}|_{\mathcal{N}^{\mathrm{I}}(p)}, \qquad \forall p \in \mathcal{P}. \tag{7.43}$$

In this case, $\mathcal{G}^{\mathrm{E}} = \mathcal{G}^{\mathrm{I}}$, i.e., each agent only estimates the minimum amount of variables needed for local computation. This is only a viable option if the resulting graphs $\{\mathcal{G}_i^{\mathrm{D}}\}_{i \in \mathcal{I}}$ ensure the desired connectedness properties in Problem 7.1*(iii)*, which cannot be expected in general (see $\mathcal{G}_1^{\mathrm{D}}$ in Figure 7.1), but holds in some cases (e.g., Example 7.5, $\mathcal{G}_2^{\mathrm{D}}$ in Figure 7.1; see also [121, Asm. 5], [123, Asm. 6] for sufficient conditions in the context of NE seeking). □

7.7.2. PROOFS

PROOF OF THEOREM 7.1

We study convergence of (7.14) in the space weighted by $\Xi \succ 0$. Let $\boldsymbol{y}^\star := \mathcal{C}(x^\star)$, where $x^\star$ is the NE of (7.10). Our proof is based on the following lemma.

Lemma 7.3. Let $\mathcal{F}(\boldsymbol{y}) := \boldsymbol{W}^{\mathrm{D}}\boldsymbol{y} - \alpha \mathrm{R}^\top \boldsymbol{F}(\boldsymbol{W}^{\mathrm{D}}\boldsymbol{y})$. Then, for any $\boldsymbol{y} \in \mathbb{R}^{n_y}$, it holds that

$$\|\mathcal{F}(\boldsymbol{y}) - \mathcal{F}(\boldsymbol{y}^\star)\|_\Xi \leq \sqrt{\rho_\alpha}\|\boldsymbol{y} - \boldsymbol{y}^\star\|_\Xi. \qquad \square$$

[16] Namely, ensuring the *feasibility* conditions in Problem 7.1*(iii)* (hard constraints), but not efficient in terms of memory/communication requirements.

Proof. Let $\boldsymbol{y} = \boldsymbol{y}_\parallel + \boldsymbol{y}_\perp$, where $\boldsymbol{y}_\parallel := \operatorname{diag}(((\mathbf{1}_{N_i} q_i^{\mathrm{D}\top}) \otimes I_{n_{x_i}})_{i\in\mathcal{I}})\boldsymbol{y} \in \mathcal{C}$, and thus $\boldsymbol{y}_\parallel = \mathcal{C}(y_\parallel)$ for some $y_\parallel \in \mathbb{R}^{n_y}$. Let $\hat{\boldsymbol{y}} := \boldsymbol{W}^{\mathrm{D}}\boldsymbol{y} = \boldsymbol{y}_\parallel + \hat{\boldsymbol{y}}_\perp$, where $\hat{\boldsymbol{y}}_\perp := \boldsymbol{W}^{\mathrm{D}}\boldsymbol{y}_\perp = \boldsymbol{W}^{\mathrm{D}}\operatorname{diag}(((I - \mathbf{1}_{N_i} q_i^{\mathrm{D}\top}) \otimes I_{n_{x_i}})_{i\in\mathcal{I}})\boldsymbol{y}$ and we used that $\boldsymbol{W}^{\mathrm{D}}\boldsymbol{y}_\parallel = \boldsymbol{y}_\parallel = \mathcal{C}(y_\parallel)$ (by row stochasticity). By Assumption 7.5, we have $q_i^{\mathrm{D}}\mathbf{1}_{N_i}^\top Q_i W_i^{\mathrm{D}}(I - \mathbf{1}_{N_i} q_i^{\mathrm{D}\top}) = \mathbf{0}$, and hence $\langle \boldsymbol{y}_\parallel, \hat{\boldsymbol{y}}_\perp\rangle_\Xi = 0$. Therefore

$$
\begin{aligned}
&\|(\hat{\boldsymbol{y}} - \alpha \mathrm{R}^\top \boldsymbol{F}(\hat{\boldsymbol{y}})) - (\boldsymbol{y}^\star - \alpha \mathrm{R}^\top \boldsymbol{F}(\boldsymbol{y}^\star))\|_\Xi^2 \\
&= \|\boldsymbol{y}_\parallel - \boldsymbol{y}^\star\|_\Xi^2 + \|\hat{\boldsymbol{y}}_\perp\|_\Xi^2 \\
&\quad + \alpha^2 \|\mathrm{R}^\top(\boldsymbol{F}(\hat{\boldsymbol{y}}) - \boldsymbol{F}(\boldsymbol{y}_\parallel) + \boldsymbol{F}(\boldsymbol{y}_\parallel) - \boldsymbol{F}(\boldsymbol{y}^\star))\|_\Xi^2 \\
&\quad - 2\alpha \langle \hat{\boldsymbol{y}}_\perp, \mathrm{R}^\top(\boldsymbol{F}(\hat{\boldsymbol{y}}) - \boldsymbol{F}(\boldsymbol{y}^\star))\rangle_\Xi \\
&\quad - 2\alpha \langle \boldsymbol{y}_\parallel - \boldsymbol{y}^\star, \mathrm{R}^\top(\boldsymbol{F}(\hat{\boldsymbol{y}}) - \boldsymbol{F}(\boldsymbol{y}_\parallel))\rangle_\Xi \\
&\quad - 2\alpha \langle \boldsymbol{y}_\parallel - \boldsymbol{y}^\star, \mathrm{R}^\top(\boldsymbol{F}(\boldsymbol{y}_\parallel) - \boldsymbol{F}(\boldsymbol{y}^\star))\rangle_\Xi. \\
&\leq \|\boldsymbol{y}_\parallel - \boldsymbol{y}^\star\|_\Xi^2 + \|\hat{\boldsymbol{y}}_\perp\|_\Xi^2 + \alpha^2(\bar{\theta}\|\hat{\boldsymbol{y}}_\perp\|_\Xi + \theta\bar{\gamma}\|\boldsymbol{y}_\parallel - \boldsymbol{y}_\parallel\|_\Xi)^2 \\
&\quad + 2\alpha\bar{\theta}\|\hat{\boldsymbol{y}}_\perp\|_\Xi(\|\boldsymbol{y}_\parallel - \boldsymbol{y}^\star\|_\Xi + \|\hat{\boldsymbol{y}}_\perp\|_\Xi) \\
&\quad + 2\alpha\theta\bar{\gamma}\|\boldsymbol{y}_\parallel - \boldsymbol{y}^\star\|_\Xi\|\hat{\boldsymbol{y}}_\perp\|_\Xi - 2\alpha\mu\underline{\gamma}^2\|\boldsymbol{y}_\parallel - \boldsymbol{y}^\star\|_\Xi^2
\end{aligned}
\tag{7.44}
$$

where the last inequality follows by the Cauchy–Schwartz inequality and using that $\|\mathrm{R}^\top v\|_\Xi^2 \leq \max_{i\in\mathcal{I}}\{[Q_i]_{i_i,i_i}\}\|v\|^2$ for all $v \in \mathbb{R}^{n_x}$, because $\mathrm{R}\Xi\mathrm{R}^\top = \operatorname{diag}(([Q_i]_{i_i,i_i} I)_{i\in\mathcal{I}})$; that $\boldsymbol{F}$ is θ-Lipschitz continuous if F is (see [23, Lem. 3]); finally, that $\|\boldsymbol{F}(\boldsymbol{y}_\parallel) - \boldsymbol{F}(\boldsymbol{y}^\star)\|^2 = \|F(y_\parallel) - F(x^\star)\|^2 \leq \theta^2\|y_\parallel - x^\star\| = \theta^2\|\boldsymbol{y}_\parallel - \boldsymbol{y}^\star\|_{\operatorname{diag}((Q_i \otimes I/(\mathbf{1}^\top Q_i\mathbf{1}))_{i\in\mathcal{I}})}$ (the last equality due to $(\boldsymbol{y}_\parallel - \boldsymbol{y}^\star) \in \mathcal{C}$) and similarly that $\langle \boldsymbol{y}_\parallel - \boldsymbol{y}^\star, \mathrm{R}^\top(\boldsymbol{F}(\boldsymbol{y}_\parallel) - \boldsymbol{F}(\boldsymbol{y}^\star))\rangle_\Xi = \langle y_\parallel - x^\star, F(y_\parallel) - F(x^\star)\rangle \geq \mu\|y_\parallel - x^\star\|^2 = \mu\|\boldsymbol{y}_\parallel - \boldsymbol{y}^\star\|^2_{\operatorname{diag}((Q_i \otimes I/(\mathbf{1}^\top Q_i\mathbf{1}))_{i\in\mathcal{I}})}$, (by using the normalization $[\mathbf{1}^\top Q_i]_{i_i} = 1$ in Assumption 7.5). In addition, by Assumption 7.4, we have $\|\hat{\boldsymbol{y}}_\perp\|_\Xi = \|\boldsymbol{W}^{\mathrm{D}}\boldsymbol{y}_\perp\|_\Xi \leq \bar{\sigma}\|\boldsymbol{y}_\perp\|_\Xi$; together with (7.44), this yields

$$
\begin{aligned}
&\|\mathcal{F}(\boldsymbol{y}) - \mathcal{F}(\boldsymbol{y}^\star))\|_\Xi \\
&\leq \begin{bmatrix} \|\boldsymbol{y}_\parallel - \boldsymbol{y}^\star\|_\Xi \\ \|\boldsymbol{y}_\perp\|_\Xi \end{bmatrix}^\top \mathrm{M}_\alpha \begin{bmatrix} \|\boldsymbol{y}_\parallel - \boldsymbol{y}^\star\|_\Xi \\ \|\boldsymbol{y}_\perp\|_\Xi \end{bmatrix} \\
&\leq \lambda_{\max}(\mathrm{M}_\alpha)(\|\boldsymbol{y}_\parallel - \boldsymbol{y}^\star\|_\Xi^2 + \|\boldsymbol{y}_\perp\|_\Xi^2) \\
&= \lambda_{\max}(\mathrm{M}_\alpha)\|\boldsymbol{y} - \boldsymbol{y}^\star\|_\Xi^2.
\end{aligned}
$$

■

To conclude the proof of Theorem 7.1, we note that (7.14) is equivalently written as $\boldsymbol{y}^{k+1} = \operatorname{proj}_{\boldsymbol{\Omega}}^{\Xi}\left(\boldsymbol{W}^{\mathrm{D}}\boldsymbol{y}^k - \alpha\mathrm{R}^\top\boldsymbol{F}(\boldsymbol{W}^{\mathrm{D}}\boldsymbol{y}^k)\right)$, where $\operatorname{proj}_{\boldsymbol{\Omega}}^{\Xi}$ is the projection in the space weighted by Ξ. In fact, $\operatorname{proj}_{\boldsymbol{\Omega}} = \operatorname{proj}_{\boldsymbol{\Omega}}^{\Xi}$ block-wise under either Assumption 7.5*(i)* (due to block diagonality of Q_i and the rectangular structure of $\boldsymbol{\Omega}$) or Assumption 7.5*(ii)* (trivially). Moreover, $\boldsymbol{y}^\star$ is a fixed point for (7.14). By nonexpansiveness of the projection operator [8, Prop. 12.28], we can finally write

$$
\begin{aligned}
\|\boldsymbol{y}^{k+1} - \boldsymbol{y}^\star\|_\Xi &= \|\operatorname{prox}_{\boldsymbol{g}}^{\Xi}(\mathcal{F}(\boldsymbol{y})) - \operatorname{prox}_{\boldsymbol{g}}^{\Xi}(\mathcal{F}(\boldsymbol{y}^\star))\|_\Xi \\
&\leq \|\mathcal{F}(\boldsymbol{y}) - \mathcal{F}(\boldsymbol{y}^\star)\|_\Xi,
\end{aligned}
$$

and the conclusion follows by Lemma 7.3. ■

PROOF OF THEOREM 7.2 (SKETCH)

We can rewrite (7.24) in terms of the variable $\boldsymbol{s} := \boldsymbol{\sigma} - \boldsymbol{B}x - \boldsymbol{b}$ by replacing $\boldsymbol{\sigma}^k$ with $\boldsymbol{s}^k + \boldsymbol{B}x^k + \boldsymbol{b}$ in (7.24a), and (7.24b) with

$$\boldsymbol{s}^{k+1} = \boldsymbol{s}^k - \beta \boldsymbol{L}^{\mathrm{D},\sigma}(\boldsymbol{s}^k + \boldsymbol{B}x^k + \boldsymbol{b}), \tag{7.45}$$

and $\boldsymbol{s}^0 = \boldsymbol{0}$. Let us define $\boldsymbol{\omega} := \mathrm{col}(x, \boldsymbol{s}, \boldsymbol{z}, \boldsymbol{\lambda})$,

$$\mathfrak{A}(\boldsymbol{\omega}) := \underbrace{\begin{bmatrix} \alpha \tilde{\boldsymbol{F}}(x, \boldsymbol{\sigma}) + \boldsymbol{B}^\top \boldsymbol{L}^{\mathrm{D},\sigma} \boldsymbol{\sigma} \\ \boldsymbol{L}^{\mathrm{D},\sigma} \boldsymbol{\sigma} \\ \boldsymbol{0} \\ \boldsymbol{b} \end{bmatrix}}_{:=\mathfrak{A}_1} + \underbrace{\begin{bmatrix} \boldsymbol{A}^\top \boldsymbol{\lambda} \\ \boldsymbol{0} \\ -\boldsymbol{L}^{\mathrm{D},\lambda} \boldsymbol{\lambda} \\ \boldsymbol{L}^{\mathrm{D},\lambda} \boldsymbol{z} - \boldsymbol{A}x \end{bmatrix}}_{:=\mathfrak{A}_2} + \underbrace{\begin{bmatrix} \mathrm{N}_\Omega \\ \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix}}_{:=\mathfrak{A}_3} \tag{7.46}$$

$$\Phi := \begin{bmatrix} \beta^{-1} I & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{A}^\top \\ \boldsymbol{0} & \beta^{-1} I & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \beta^{-1} I & \boldsymbol{L}^{\mathrm{D},\lambda} \\ -\boldsymbol{A} & \boldsymbol{0} & \boldsymbol{L}^{\mathrm{D},\lambda} & \beta^{-1} I \end{bmatrix}, \tag{7.47}$$

where $\boldsymbol{\sigma}$ is not not a variable but just shorthand notation for $\boldsymbol{\sigma} = \boldsymbol{s} + \boldsymbol{B}x + \boldsymbol{b}$. We assume that $\beta > 0$ is chosen small enough, such that $\Phi \succ 0$. The proof is based on the following auxiliary results.

Lemma 7.4 (*Invariance*)**.** For all $k \in \mathcal{N}$, $\Pi_{\|}^{\sigma} \boldsymbol{s}^k = \boldsymbol{0}$. □

Proof. It follows by induction, by using (7.25) and (7.45). ■

Lemma 7.5 (*Algorithm derivation*)**.** The iteration in (7.24), with (7.24b) replaced by (7.45), can be written as

$$\mathfrak{A}_2(\boldsymbol{\omega}^{k+1}) + \mathfrak{A}_3(\boldsymbol{\omega}^{k+1}) = \mathfrak{A}_1(\boldsymbol{\omega}^k) + \Phi(\boldsymbol{\omega}^{k+1} - \boldsymbol{\omega}^k), \tag{7.48}$$

with $\mathfrak{A}_1$, $\mathfrak{A}_2$, $\mathfrak{A}_3$ as in (7.46). □

Proof. It follows by expanding the terms, and by recalling that $(\mathrm{Id} + \mathrm{N}_\Omega)^{-1} = \mathrm{proj}_\Omega$. ■

Lemma 7.6 (*Fixed points*)**.** The fixed points of (7.48) coincide with $\mathrm{zer}(\mathfrak{A})$. The set $\mathrm{zer}(\mathfrak{A}) \cap \Sigma := \{\boldsymbol{\omega} \mid \Pi_{\|}^{\sigma} \boldsymbol{s} = \boldsymbol{0}\}$ is nonempty. Moreover, for any $\boldsymbol{\omega}^\star = (x^\star, \boldsymbol{s}^\star, \boldsymbol{z}^\star, \boldsymbol{\lambda}^\star) \in \mathrm{zer}(\mathfrak{A}) \cap \Sigma$, we have that $\sigma^\star := \boldsymbol{s}^\star + \boldsymbol{B}x^\star + \boldsymbol{b} = \mathcal{C}(\sigma(x^\star))$, $\boldsymbol{\lambda}^\star = \mathcal{C}(\lambda^\star)$, where $(x^\star, \lambda^\star)$ solve (7.17), hence $x^\star$ is the v-GNE of the game in (7.10). □

Proof. (7.48) is the FB algorithm [8, §26.5] applied to the operator $\Phi^{-1}\mathfrak{A}$; it is known that its fixed points are the zeros of $\mathfrak{A}$ [109, Lem. 2]. Consider any $\boldsymbol{\omega}^\star \in \mathrm{zer}\mathfrak{A} \cap \Sigma$. Note that $\boldsymbol{\sigma}^\star = \Pi_{\|}^{\sigma} \boldsymbol{\sigma}^\star = \Pi_{\|}^{\sigma} \boldsymbol{s}^\star + \Pi_{\|}^{\sigma} \boldsymbol{B}x^\star + \Pi_{\|}^{\sigma} \boldsymbol{b} = \mathcal{C}^\sigma(\sigma(x^\star))$, where the first equality follows by the second row in (7.46) and Assumption 7.6 (and Lemma 7.1), and the second by definition of $\boldsymbol{B}$, $\boldsymbol{b}$ and $\Pi_{\|}^{\sigma}$. By the third row in (7.46), $\boldsymbol{\lambda}^\star \in \mathcal{C}^\lambda$; in turn, the first row retrieves the first KKT condition in (7.17). The second condition in (7.17) is obtained by left-multiplying the last row in (7.46) by $\Pi_{\|}^{\lambda}$. Conversely, $\mathrm{zer}(\mathfrak{A}) \cap \Sigma \neq \varnothing$ can be shown similarly to [22, Lem. 10]. ■

Lemma 7.7 (*Monotonicity properties*)**.** The operator $\mathfrak{A}_2+\mathfrak{A}_3$ is monotone. For any α small-enough, the operator $\mathfrak{A}_1$ is η-restricted cocoercive, for some $\eta>0$: for all $\boldsymbol{\omega}\in\Sigma$ and all $\boldsymbol{\omega}^\star\in\operatorname{zer}(\mathfrak{A})\cap\Sigma$, $\langle\mathfrak{A}_1(\boldsymbol{\omega})-\mathfrak{A}_1(\boldsymbol{\omega}^\star),\boldsymbol{\omega}-\boldsymbol{\omega}^\star\rangle\geq\eta\|\mathfrak{A}_1(\boldsymbol{\omega})-\mathfrak{A}_1(\boldsymbol{\omega}^\star)\|^2$. □

Proof. For the first part, $\mathfrak{A}_1$ is monotone because it is a skew-symmetric linear operator [8, Ex 20.35], $\mathfrak{A}_2$ is monotone because N_Ω is [8]. The second statement follows as in [64, Lemma 4] by using Lemma 7.2, that $\tilde{\boldsymbol{F}}$ is Lipschitz, that $\tilde{\boldsymbol{F}}(x,\Pi_\parallel^\sigma\boldsymbol{\sigma})=F(x)$ and $F(x)$ is strongly monotone. ■

Lemma 7.5 recasts (7.24) as a preconditioned FB algorithm [109], applied to the operators $\mathfrak{A}_1$, $\mathfrak{A}_2+\mathfrak{A}_3$, with preconditioning matrix Φ. Then, with Lemma 7.7 in place and restricting the analysis to the invariant subspace Σ, the convergence of (7.48) to a fixed point can be shown in the space $\mathcal{H}_\Phi$ analogously to [109, Th. 2]; finally, Lemmas 7.4 and 7.6 characterize such a fixed point as per statement. ■

PROOF OF THEOREM 7.3

We adapt the proof of [144, Th. 24]. We note that $\boldsymbol{z}^0=\mathbf{0}_{n_y}\in\operatorname{range}(B)$; by the conditions 7.3(*i*) and 7.3(*iv*), the update in (7.30), and an induction argument, we have $\boldsymbol{y}^k,\boldsymbol{z}^k\in\operatorname{range}(B)$, for all $k\geq1$. Hence, we rewrite (7.30) as

$$\boldsymbol{y}^k=B\underline{\boldsymbol{y}}^k,\quad \boldsymbol{z}^k=\gamma B\underline{\boldsymbol{z}}^k \tag{7.49a}$$

$$\underline{\boldsymbol{y}}^{k+1}=D\boldsymbol{y}^k-\gamma(\nabla_{\boldsymbol{y}}\boldsymbol{f}(\boldsymbol{y}^k)+\underline{\boldsymbol{z}}^k) \tag{7.49b}$$

$$\underline{\boldsymbol{z}}^{k+1}=\underline{\boldsymbol{z}}^k+\tfrac{1}{\gamma}C\underline{\boldsymbol{y}}^{k+1} \tag{7.49c}$$

for all $k\geq1$. Let $\Phi(\boldsymbol{y},\boldsymbol{z}):=\boldsymbol{f}(\boldsymbol{y})+\langle\boldsymbol{y},\boldsymbol{z}\rangle$; the form in (7.49) can be exploited to prove the following lemma.

Lemma 7.8. Let $(\boldsymbol{y}^k,\underline{\boldsymbol{y}}^k,\underline{\boldsymbol{z}}^k)$ be a sequence generated by (7.49). Then, for all $\boldsymbol{y}\in\mathcal{C}$, $\boldsymbol{z}\in\mathcal{C}_\perp$, it holds that:

$$\Phi(\boldsymbol{y}_{\mathrm{avg}}^{k+1},\boldsymbol{z})-\Phi(\boldsymbol{y},\boldsymbol{z})\leq\tfrac{1}{2k}h(\boldsymbol{y},\boldsymbol{z}),$$

where $h(\boldsymbol{y},\boldsymbol{z}):=\frac{1}{\gamma}\|\boldsymbol{y}^0-\boldsymbol{y}\|_D^2+\gamma\frac{\|B-\Pi_\parallel\|}{\underline{\lambda}}\|\boldsymbol{z}\|^2$ and $\underline{\lambda}:=\min\{(\lambda_2(C_p))_{p\in\mathcal{P}}\}$. □

Proof. The proof is analogous to that of [144, Lemma 23], and omitted here. Note that [144] uses a matrix notation (i.e., $\boldsymbol{y}\in\mathbb{R}^{I\times n}$), while we need a stacked notation (as the vectors $(\boldsymbol{y}_i)_{i\in\mathcal{I}}$ are not homogeneous in size). Nonetheless, (7.49) matches [144, Eq. (33)], which allows us to repeat all the steps in [144, Lem. 23] (with the only precaution of replacing J, $\operatorname{span}(\mathbf{1}_m)$, $\lambda_2(C)$ in [144] with $\Pi_\parallel$, $\mathcal{C}$, $\underline{\lambda}$). ■

For all $\boldsymbol{z}\in\mathcal{C}_\perp$ (so that $\langle\boldsymbol{z},\boldsymbol{y}^\star\rangle=0$), setting $\boldsymbol{y}=\boldsymbol{y}^\star$ in Lemma 7.8, together with the definition of Φ, yields $\boldsymbol{f}(\boldsymbol{y}_{\mathrm{avg}}^k)-\boldsymbol{f}(\boldsymbol{y}^\star)+\langle\boldsymbol{y}_{\mathrm{avg}}^k,\boldsymbol{z}\rangle\leq\frac{1}{2k}h(\boldsymbol{y}^\star,\boldsymbol{z})$. Further choosing $\boldsymbol{z}=2\frac{\Pi_\perp\boldsymbol{y}_{\mathrm{avg}}^k}{\|\Pi_\perp\boldsymbol{y}_{\mathrm{avg}}^k\|}\|\boldsymbol{z}^\star\|$, with $\boldsymbol{z}^\star:=-\nabla_{\boldsymbol{y}}\boldsymbol{f}(\boldsymbol{y}^\star)$, leads to

$$\boldsymbol{f}(\boldsymbol{y}_{\mathrm{avg}}^k)-\boldsymbol{f}(\boldsymbol{y}^\star)+2\|\boldsymbol{z}^\star\|\left\|\Pi_\perp\boldsymbol{y}_{\mathrm{avg}}^k\right\|\leq\tfrac{1}{2k}h(\boldsymbol{y}^\star,2\boldsymbol{z}^\star). \tag{7.50}$$

By convexity and since $\boldsymbol{z}^\star\in\mathcal{C}_\perp$ (by optimality conditions), it holds that $f(\boldsymbol{y}_{\mathrm{avg}}^k)-f(\boldsymbol{y}^\star)\geq-\langle\boldsymbol{y}_{\mathrm{avg}}^k-\boldsymbol{y}^\star,\boldsymbol{z}^\star\rangle=-\langle\Pi_\perp\boldsymbol{y}_{\mathrm{avg}}^k,\boldsymbol{z}^\star\rangle\geq-\|\Pi_\perp\boldsymbol{y}_{\mathrm{avg}}^k\|\|\boldsymbol{z}^\star\|$; the latter inequality and (7.50) imply $\mathfrak{M}(\boldsymbol{y}_{\mathrm{avg}}^k)\leq\frac{1}{2k}h(\boldsymbol{y}^\star,2\boldsymbol{z}^\star)$. ■

PROOF OF THEOREM 7.4

Note that, for each $p \in \mathcal{P}$, (7.37) is the standard perturbed push-sum protocol [102, Eq. (4)], with perturbation term $-\gamma^k \boldsymbol{g}_{i,p}^{k+1}$. Therefore, since $\boldsymbol{g}_{i,p}^{k+1}$ is uniformly bounded by assumption and by the choice of $(\gamma^k)_{k\in\mathcal{N}}$, we can apply [102, Lem. 1] to infer that, for all $i \in \mathcal{I}$, $p \in \mathcal{N}^{\mathrm{E}}(i)$

$$\lim_{k\to\infty} \|\boldsymbol{y}_{i,p}^k - \bar{\boldsymbol{z}}_p^k\| = 0, \tag{7.51}$$

$$\textstyle\sum_{k=0}^{\infty} \gamma^k \|\boldsymbol{y}_{i,p}^k - \bar{\boldsymbol{z}}_p^k\| = 0, \tag{7.52}$$

where $\bar{\boldsymbol{z}}_p^k := \frac{1}{N_p} \sum_{i\in\overline{\mathcal{N}}^{\mathrm{E}}(p)} \boldsymbol{z}_{i,p}^k \in \mathbb{R}^{n_{y_p}}$, for all $k \in \mathcal{N}$. Let us also define $\bar{\boldsymbol{z}}^k := \mathrm{col}((\bar{\boldsymbol{z}}_p^k)_{p\in\mathcal{P}}) \in \mathbb{R}^{n_y}$. By (7.37) and Assumption 7.7(ii), it follows that

$$\bar{\boldsymbol{z}}_p^{k+1} = \bar{\boldsymbol{z}}_p^k - \gamma^k \tfrac{1}{N_p} \textstyle\sum_{i\in\overline{\mathcal{N}}^{\mathrm{E}}(p)} \boldsymbol{g}_{i,p}^{k+1}. \tag{7.53}$$

We next show that $\lim_{k\to\infty} \bar{\boldsymbol{z}}^k = y^\star \in \mathcal{Y}^\star$; then, the theorem follows by (7.51). The main complication with respect to the proof of [102, Th. 1] is that we need a modification of [102, Lem. 8] to cope with the non-homogeneity of the estimates.

Lemma 7.9. For all $y^\star \in \mathcal{Y}^\star$, for all $k \in \mathcal{N}$, it holds that

$$\begin{aligned}\|\bar{\boldsymbol{z}}^{k+1} - y^\star\|_D^2 \le \|\bar{\boldsymbol{z}}^k - y^\star\|_D^2 - 2\gamma^k (f(\bar{\boldsymbol{z}}^k) - f(y^\star)) \\ + 4L\gamma^k \sum_{i\in\mathcal{I}} \sum_{p\in\overline{\mathcal{N}}^{\mathrm{E}}(i)} \|\bar{\boldsymbol{z}}_p^k - \tilde{\boldsymbol{y}}_{i,p}^{k+1}\| + (\gamma^k)^2 NL^2,\end{aligned}$$

where $D := \mathrm{diag}((N_p I_{n_p})_{p\in\mathcal{P}})$. □

Proof. By (7.53), we have

$$\begin{aligned}\|\bar{\boldsymbol{z}}^{k+1} - y^\star\|_D^2 = \|\bar{\boldsymbol{z}}^k - y^\star\|_D^2 \\ - 2\gamma^k \sum_{p\in\mathcal{P}} \left\langle \bar{\boldsymbol{z}}_p^k - y_p^\star, \textstyle\sum_{i\in\overline{\mathcal{N}}^{\mathrm{E}}(p)} \boldsymbol{g}_{i,p}^{k+1} \right\rangle \\ + (\gamma^k)^2 \sum_{p\in\mathcal{P}} \tfrac{1}{N_p} \left\| \textstyle\sum_{i\in\overline{\mathcal{N}}^{\mathrm{E}}(p)} \boldsymbol{g}_{i,p}^{k+1} \right\|^2.\end{aligned} \tag{7.54}$$

The third addend on the right-hand side of (7.54) is bounded above by $(\gamma^k)^2 NL^2$. For the second addend, we have

$$\begin{aligned}&\sum_{p\in\mathcal{P}} \left\langle \bar{\boldsymbol{z}}_p^k - y_p^\star, \textstyle\sum_{i\in\overline{\mathcal{N}}^{\mathrm{E}}(p)} \boldsymbol{g}_{i,p}^{k+1} \right\rangle \\ &= \sum_{i\in\mathcal{I}} \sum_{p\in\overline{\mathcal{N}}^{\mathrm{E}}(i)} \left\langle (\bar{\boldsymbol{z}}_p^k - \boldsymbol{y}_{i,p}^{k+1}) + (\boldsymbol{y}_{i,p}^{k+1} - y_p^\star), \boldsymbol{g}_{i,p}^{k+1} \right\rangle \\ &\overset{(a)}{\ge} \sum_{i\in\mathcal{I}} -L\|\mathrm{col}((\bar{\boldsymbol{z}}_p^k)_{p\in\overline{\mathcal{N}}^{\mathrm{E}}(i)}) - \tilde{\boldsymbol{y}}_i^{k+1}\| + f_i(\tilde{\boldsymbol{y}}_i^{k+1}) - f_i(y^\star) \\ &\overset{(b)}{\ge} \sum_{i\in\mathcal{I}} -2L\|\mathrm{col}((\bar{\boldsymbol{z}}_p^k)_{p\in\overline{\mathcal{N}}^{\mathrm{E}}(i)}) - \tilde{\boldsymbol{y}}_i^{k+1}\| + f_i(\bar{\boldsymbol{z}}^{k+1}) - f_i(y^\star),\end{aligned}$$

where in (a) we used that $\boldsymbol{g}_i^{k+1} \in \partial_{\tilde{y}_i} f_i(\tilde{\boldsymbol{y}}_i^{k+1})$ and convexity of f_i, and (b) follows by adding and subtracting (inside the sum) $f_i((\bar{\boldsymbol{z}}_p^{k+1})_{p\in\mathcal{N}^{\mathrm{E}}(i)}) = f_i(\bar{\boldsymbol{z}}^{k+1})$ and by L-Lipschitz continuity of f_i. The result follows by substituting the bound back into (7.54). ■

We finally note that, due to (7.52) and the choice of $(\gamma_k)_{k\in\mathcal{N}}$, the inequality in Lemma 7.9 satisfies all the conditions of [102, Lem. 7], in the norm $\|\cdot\|_D$; hence we can conclude that $\bar{\boldsymbol{z}}^k \to y^\star$, for some $y^\star \in \mathcal{Y}^\star$. ■

IV

Conclusion

8

CONCLUDING REMARKS

Results? Why man, I have gotten a lot of results. I know several thousand things that won't work.

Thomas Edison

There will come a time when you believe everything is finished. That will be the beginning.

Louis L'Amour

In this dissertation, we have studied distributed algorithms for generalized Nash equilibrium seeking, fitted for coordination and control of modern complex network systems. In this last chapter, we summarize our contributions to the field of games in partial-decision information, and we assess to which extent our research goals have been achieved. Finally, we conclude the thesis by highlighting prominent open challenges and research prospects.

8.1. CONTRIBUTIONS

THIS PhD dissertation contributes to the advancement of game-theoretical methods for intelligent coordination of network systems, by developing distributed, center-free approaches for GNE seeking. We focused on the partial-decision information scenario, and designed algorithm integrating consensus dynamics within the local decision processes. In detail, the novelties of this thesis are summarized as follows:

- **Fully-distributed preconditioned proximal-point algorithms**
  We introduced the first single-layer fixed-step fully-distributed GNE seeking iterations based on proximal best-response. In Chapter 2, we showed, theoretically and numerically, that the proposed algorithms outperform the state of the art methods, in terms of both computation and communication requirements. Our operator-theoretic derivation also facilitates the convergence analysis in case of aggregative games, acceleration schemes and inexact updates. Furthermore, in Chapter 3, we proved that our proximal-point algorithms are suitable to address convex games in partial-decision information under a mild condition that does not require strong monotonicity nor smoothness of the game mapping.

- **Feedback GNE seeking for dynamical agents**
  In Chapter 4, we designed continuous-time fully-distributed feedback compensators to solve generalized (aggregative) games played by dynamical agents. We focused on networks of feedback linearizable systems, and devised dynamic controllers based on primal-dual pseudo-gradient algorithms. As a key contribution, we proved closed-loop global asymptotic stability of the v-GNE, in the presence of general nonlinear coupling constraints. Furthermore, our controllers can be tuned in a totally decentralized way and without requiring any global information – on the game or on the communication graph – thanks to the use of uncoordinated integral gains.

- **Games over switching and directed graphs**
  We addressed NE problems in partial decision information without requiring symmetric and time-invariant communication. In Chapter 5, we studied a pseudo-gradient iteration that converges Q-linearly over time-varying, doubly-stochastic networks. Thanks to an original convergence argument, we provided step sizes bounds independent of the number of agents. In Chapter 6, we extended the analysis to the case of a fixed, but row stochastic, networks, by using (for the updates) the PF eigenvector, that can be computed online. In Chapter 7, we further improved on this result and showed that single-iteration contractivity can be achieved (in a suitable norm) by means of a small-enough step and without knowledge of the PF eigenvector.

- **Exploiting the game sparsity**
  In Chapter 7, we proposed a graph-theoretic language for design of efficient distributed algorithms, where the sparsity in the agents' coupling is taken into account to reduce the amount of variables estimated and transmitted by each agent. We also leveraged our framework to unify the convergence analysis of pseudo-

gradient methods in full- and partial- decision information, and to develop a scalable solution method for a novel class of generalized aggregative games.

- **Fundamental theoretical results and methods**
  Our research on games in partial-decision information also led to several fundamental results of independent and general interest. Here we mention: (i) the analysis of the proximal-point algorithm under a mild restricted monotonicity assumption, in Chapter 2; (ii) a methodology to study the dynamic tracking of a variable in the context of operator-splitting methods (specifically, a forward step), by means of a change of coordinates, in Chapters 2 and 4; (iii) a linear convergence proof, based on contractivity and monotonicity properties over two complementary subspaces: this strategy – that immediately extends to different variational problems over graphs – exploits orthogonality to provide stronger results with respect to existing approaches (Chapter 5) and is suitable to cope with directed communication (Chapters 6 and 7).

ANSWERING THE RESEARCH QUESTIONS

We next summarize how the key research questions formulated in Section 1.4 are addressed in this thesis.

Q1. How to achieve fast, scalable, communication-efficient, fully-distributed GNE seeking?

Algorithms that are extremely fast and only require one communication per update can be designed by leveraging preconditioned proximal-point methods, based on consensus and proximal best-response dynamics, as shown in Chapter 2. Beside their efficiency, PPPAs scale well with the network dimension and their performance is guaranteed for a wide range of game parameters.

Alternatively, Chapter 5 proves that a a contractivity analysis can be exploited to improve on the known step sizes bounds for a gradient-based fully-distributed NE seeking iteration, thus achieving superior guaranteed convergence rate. This approach answers one key tuning problem of pseudo-gradient algorithms, where large step sizes can jeopardize convergence, but theoretical upper-bounds are usually impractically small.

Finally, independently of the methodology of choice, efficiency can (and should) be enhanced by adjusting the algorithms to the specific structure of the application at hand, for instance aggregative costs or partial coupling among the agents (in the objectives or constraints). Chapter 8 copes with the latter issue, and provides a means to boost communication-efficiency and scalability, in many practical scenarios, via judicious design of sparsity-aware consensus dynamics.

Q2. What methods can be employed to relax monotonicity and smoothness requirements in GNE seeking under partial-decision information?

While some literature focuses on pseudo-gradient based algorithms for non-strongly monotone games [66], [87], this dissertation puts forward the proximal-point method, as a natural candidate to relax both monotonicity and smoothness

assumptions for fully-distributed GNE seeking. Although the partial-decision information scenario introduces some technical difficulties, Chapter 3 shows that a preconditioned proximal-point algorithm can be used to seek a GNE under the restricted monotonicity of an augmented operator, a condition that can be difficult to verify but is very mild in practice, requiring neither monotonicity nor continuity of the game mapping.

Q3. How to design distributed controllers to solve GNE problems in the presence of dynamical agents?

A modular solution is to study primal-dual fully-distributed continuous-time algorithms for computing a GNE; then such algorithms can be employed as a building block to design feedback controllers, depending on the specific physical dynamics of the systems considered. For multi-integrator agents, this is achieved by means of a proportional compensator and by leveraging a change of coordinates; in turn, nonlinear agents with maximal relative degree can be addressed via a linearizing feedback (Chapter 4). The stability analysis in this thesis is based on monotonicity properties, and immediately extends to some networks of passive agents [118], or even passive networks of dynamically interconnected systems. The approach is complementary to methods based on singular perturbation (where exponential stability is required) [15], [111], all together providing a solid base to address GNE problems for a broad class of dynamical systems.

Q4. How to analyze convergence to an equilibrium in games played over directed, time-varying communication networks?

For strongly monotone NE problems in partial-decision information, the convergence of pseudo-gradient algorithms can be studied by looking at one-step contractivity properties of the iterates. In fact, for the case of doubly stochastic (balanced) communication, Q-linear convergence can be proven even if the network topology is time-varying, as shown in Chapter 5. While this thesis focused on networks that are (strongly) connected at every instant, the approach carries on to jointly connected graphs. The case of row stochastic (unbalanced) networks is more complex, because strong monotonicity of the game mapping and restricted contractivity of the network matrix holds for two different inner products. This problem is solved by using a modified algorithm (where the game mapping is modulated by the PF eigenvector, as in Chapter 6), or by performing the analysis in a suitably weighted space (Chapter 7). Unfortunately, this technique does not extend to generalized games over directed (or time-varying) networks, for which fixed-step solution methods are currently unknown.

8.2. FUTURE RESEARCH AND RECOMMENDATIONS

SOME prominent open problems specific to the field of GNE seeking under partial-decision information, directions for future research, as well as possible extensions of the results in this thesis, are discussed next:

- **Solving merely monotone games**
  In Chapter 3, we have proven convergence of our PPPA under the restricted mono-

tonicity of an augmented operator, depending both on the game and network parameters. While this condition is significantly weaker than the (usually postulated) strong monotonicity of the game mapping, it is often hard to verify: future work should provide sufficient conditions for its satisfaction, at least for some prevalent classes of games (e.g., linearly coupled). It would be also highly valuable to extend the applicability of our method by studying its convergence under (mere) monotonicity of the game mapping, a well-studied assumption for which analytical tests are available. In fact, the only works that address merely monotone GNEPs via fully-distributed algorithms require double-layer iterations [148] or vanishing step sizes, which compromise efficiency and prevent real-time implementation. How to solve merely monotone (possibly smooth) GNEPs, under partial-decision information, via single-layer and fixed-step methods? Working out this question in its full generality (with no further assumptions, if at all possible) will likely require a new methodological approach, that does not rely on the monotonicity of extended operators (cf. Chapters 3 and 4) nor on the contractivity of game-related mappings (cf. Chapter 5).

- **GNE seeking over directed and time-varying networks**
  The approach in Chapters 5 and 7 to solve strongly monotone NEPs over a directed graph could be extended, with some modifications, to prove linear convergence in the case of time-varying networks (not necessarily strongly connected at every iteration). More complex is the case of games with coupling constraints. If allowing for vanishing step-sizes, and having access to column-stochastic weights, fully-distributed iterations can be obtained by leveraging known consensus protocols (e.g.,, [125, §3]) and a Krasnosel'skii-Mann analysis as in [10]; nonetheless, it is not clear how to study fixed-step methods. The technical complications are similar to those encountered with merely monotone games (since the KKT operator, enclosing the primal-dual optimality conditions, is not strongly monotone, even if the game mapping is), although the problem exhibits more structure. For the case of asymmetric but time-invariant communication, a promising option is the choice of a non-self adjoint preconditioning matrix, for our PPPA (Chapter 2) or for preconditioned FB methods as in [109], an idea that was first proposed in [34].

- **Dealing with disturbed and constrained dynamical agents**
  Chapter 4 introduced feedback controllers to achieve a GNE when the agents' dynamics are perfectly modeled. Future work should consider the presence of unknown disturbances, measurement noise and parametric uncertainties. Current approaches for NEPs [116], [154] exploit ISS, which would not be ensured for non-strongly monotone games. One interesting alternative is to leverage the *integral* input-to-state stability properties guaranteed by averaged operators [41], when disturbances can be asymptotically compensated. Moreover, in safety critical applications (e.g., autonomous driving), it is paramount to enforce certain operating constraints at every instant, even during the system transient. This problem was addressed, for a particular class of passive systems with output constraints, in [117], [118] (based on penalty methods). Nonetheless, the problem remains largely unexplored for more general dynamics and for the case of input constraints.

- **Equilibrium seeking for passive networks and power systems**
  The fully-distributed feedback controllers developed in Chapter 4 are also suitable (under mild assumptions) for GNE seeking in dynamical networks with equilibrium-independent passivity properties [128]. In general, this further requires the design of a smart feedforward term or a suitable integral action, to render the unknown GNE a steady state. The most interesting case is that of systems with cyberphysical coupling, where the agents' physical dynamics are also intertwined. This scenario arises for instance in multi-agent power flow problems on passive energy networks [48], where each microgrid is regarded as a self-interested decision maker, that aims at maximizing its profit while obeying power balance coupling constraints. One of the main complications is that the power demand is typically unknown and acts as a disturbance on the dynamical network, making the design of fully-distributed GNE seeking controllers in this setup a challenging problem.

- **Pseudo-gradient methods for nonsmooth games**
  Nonsmooth GNEPs (where the part of the pseudo-gradient that couples the agents is not Lipschitz) were solved via resolvent based iterations, as the Douglas–Rachford [75] or the proximal point (Chapter 3) algorithms. It would be also interesting to study gradient-based alternatives, resting on either line search [152] or adaptive rules for the step sizes that can account for local smoothness [96]. Whether such procedures can be effectively implemented in a distributed way – and beat the performance of implicit methods – is the question to be answered in future research.

- **Design of adaptive step sizes and dynamic estimate assignment**
  In the literature of fully-distributed GNE seeking algorithms, the theoretical bounds for the step sizes are typically: (i) dependent on global parameters, not available locally; (ii) based on a worst-case scenario, hence overly conservative in practice (as demonstrated by our numerical studies). A solution to both issues is provided by adaptive step sizes (updated online), that can greatly improve convergence speed and allow for a fully uncoordinated tuning, without giving up on a-priori convergence guarantees. While we have successfully exploited this option for the continuous-time method (Chapter 4), it would be highly valuable to design adaptive rules for discrete-time algorithms as well. To further decentralize the algorithm deployment, future work should also focus on computationally efficient and distributed methods to dynamically assign the estimates kept by each agent (see Chapter 7), for instance in the spirit of [5]. This could reduce memory and communication overhead for sparse problems, without requiring preliminary design effort and even allowing for plug-and-play operation.

- **Performance guarantees**
  With regards to strongly monotone non-generalized games, in Chapters 2 and 5 we provided algorithms with faster guaranteed linear convergence rate, compared to the existing literature. One intriguing unresolved question is whether methods can be designed where: (i) the step sizes can be chosen independently of the communication topology; (ii) the convergence rate depends on the parameters of

the game and of the network in a decoupled way (and possibly matches the rate of centralized algorithms for sufficiently connected, but not complete, graphs). This *rate separation* was only recently achieved for consensus optimization [91], [144]. It would be also interesting to study assumptions that can ensure linear convergence to an NE, under partial-decision information, in the absence of strong monotonicity. For instance, can linear convergence be achieved when the (merely monotone) game mapping is inverse Lipschitz continuous (see Chapter 3)? Such property ensures uniqueness of the equilibrium, as well as the contractivity of certain resolvents, a fact that could be used in the analysis. Also challenging is the case of games with multiple NEs, where geometric convergence should be to the *set* of solutions.

- **Research beyond the scope of this thesis**
  We next mention some critical issues, often encountered in engineering systems and practical implementation, that were not addressed in this dissertation:
  - *Stochasticity:* In application like transportation systems and electricity markets, the uncertainty on some quantities is modeled via *stochastic* GNEP, with expected value cost functions. This setup was addressed in [59] via a fully-distributed pseudo-gradient method. To improve the sample efficiency, an attracting idea would be to instead exploit proximal-best response dynamics as in Chapter 2.
  - *Asynchronicity and imperfect communication:* Asynchronous algorithms can reduce idle time, transmission and memory-access congestion, and eliminate the need for synchronization, which is costly in large networks. Although some GNE seekinh methods that allow the use of delayed information have been introduced [35] – even in partial-decision information [150] – it would be valuable to analyze algorithms in the more general *partially asynchronous* scenario [18], to ensure convergence in face of lossy communication and uncoordinated updates as well. It would be also interesting to quantify the robustness of GNE seeking algorithms to noisy communication (in the absence of coupling constraints, this can be done via ISS arguments, see Chapter 5).
  - *Presence of adversarial agents:* As partial-decision information methods rely on peer-to-peer data exchange, they are inherently prone to communication attacks and susceptible to the presence of non-truthful – or even adversarial – decision makers. This problem was first addressed, via a resilient consensus protocol, in [65], [67], but for unconstrained games only and under some restrictive assumptions.
  - *Time-varying games:* The agents in a game often operate in a dynamic environment, for instance in GNEP for multiple autonomous vehicles, power system applications and cognitive radio networks. In such context, it might be necessary to explicitly account for the evolution of the environment, via non-constant objective functions (or constraints), with the goal of tracking a GNE of the time-varying game [16]. To the best of our knowledge, this problem was not addressed in games under partial-decision information.

8.3. A BIRD'S-EYE VIEW

WITH the increased complexity and liberalization in large-scale multi-agent engineering systems, game theory is emerging as a powerful framework to achieve robust and efficient operation of many critical infrastructures. The present dissertation focuses on games played in the partial-decision information scenario. Addressing such a setup is a fundamental building block for the development of coordination mechanisms that can improve dependability – and save money and energy – in multi-agent autonomous driving systems, cognitive radio systems, and virtually in any network application where the decision makers can only rely on peer-to-peer communication.

While network optimization has come a long way, the research in GNE seeking under partial-decision information is still in its infancy. The field has been increasingly recognized in the latest years, for aggregative games and beyond; the original technical challenges arising in this setup are even stimulating the development of novel mathematical tools, distributed learning and operator-theoretical methods. Nonetheless, many essential questions demand an answer, before a reliable practical implementation is possible. In this perspective, this thesis advances the theoretical understanding of games in the partial-decision information scenario, and provides a broad tool kit for designing efficient algorithmic solutions, suitable to cope with complex network interaction and dynamic coupling. We hope this work will bring the deployment of fully-distributed GNE seeking methods in industry one step closer.

APPENDICES

A

BASIC NOTATION

THE following notation is used throughout the disseration. For convenience, a list of symbols is also provided at the end of the thesis.

- $\mathbb{N}$ is the set of natural numbers, including 0.
- $\mathbb{R}$ is the set of real numbers, $\mathbb{R}_{\geq 0}$ is the set of nonnegative real numbers, $\mathbb{R}_{>0}$ is the set of positive real numbers, $\overline{\mathbb{R}} := \mathbb{R} \cup \{\infty\}$ is the extended real line.
- $\mathbb{R}^q$ is the set of real (column) vectors of dimension q, $\mathbb{R}^q_{\geq 0}$ is the set of nonnegative real vectors of dimension q.
- (a,b) and $[a,b]$ denote open and closed intervals, i.e., $(a,b) = \{x \in \mathbb{R} \mid a < x < b\}$, $[a,b] = \{x \in \mathbb{R} \mid a \leq x \leq b\}$;
- $\mathbb{R}^{p\times q}$ is the set of real matrices with p rows and q columns, $\mathbb{R}^{p\times q}_{\geq 0}$ is the set of real matrices, with p rows, q columns and nonnegative elements.
- ℓ^1 is the set of absolutely summable sequences.
- $\mathcal{X}\times\mathcal{Y}$, $\mathcal{X}\cup\mathcal{Y}$, $\mathcal{X}\cap\mathcal{Y}$, $\mathcal{X}\backslash\mathcal{Y}$, $\mathcal{X}+\mathcal{Y}$ denote the Cartesian product, union, intersection, difference and Minkowsi sum of two sets $\mathcal{X}$ and $\mathcal{Y}$, respectively.
- $\mathbf{0}_q \in \mathbb{R}^q$ is a vector with all elements equal to 0; $\mathbf{1}_q \in \mathbb{R}^q$ is a vector with all elements equal to 1; $I_q \in \mathbb{R}^{q\times q}$ is an identity matrix. The subscripts may be omitted when the dimension can be inferred from the context.
- $\mathbf{e}_i$ denotes a vector of appropriate dimension with i-th element equal to 1 and all other elements equal to 0.
- $\otimes$ denotes the Kronecker product.
- Given $f : \mathbb{R} \to \mathbb{R}$ and $g : \mathbb{R} \to \mathbb{R}$, the notation $f(x) = O(g(x))$ means that there exist $\epsilon, c > 0$ such that $f(x) \leq cg(x)$ for all x such that $\|x\| \leq \epsilon$.

- For a matrix $A \in \mathbb{R}^{p\times q}$:
  - $A^\top$ is its transpose;
  - $[A]_{i,j}$ is the element on row i and column j;
  - $\mathrm{null}(A) \coloneqq \{x \in \mathbb{R}^q \mid Ax = \mathbf{0}_n\}$ is the null space of A;
  - $\mathrm{range}(A) \coloneqq \{v \in \mathbb{R}^p \mid v = Ax, x \in \mathbb{R}^q\}$ is the range of A;
  - if A is square, then $\det(A)$ is its determinant, $\rho(A)$ is its spectral radius, $\Lambda(A)$ is its spectrum;
  - $\sigma_{\min}(A) =: \sigma_1(A) \leq \ldots \leq \sigma_q(A) =: \sigma_{\max}(A)$ denote the singular values of A;
  - $\|A\| = \sigma_{\max}(A)$ is the largest singular value of A;
  - $\|A\|_\infty$ is is the maximum of the absolute row sums of A.
- If $A = A^\top \in \mathbb{R}^{q\times q}$ is a symmetric matrix, then:
  - $\lambda_{\min}(A) =: \lambda_1(A) \leq \ldots \leq \lambda_q(A) =: \lambda_{\max}(A)$ denote its eigenvalues;
  - $A \succ 0$ stands for a positive definite matrix;
  - $A \succcurlyeq 0$ stands for a positive semidefinite matrix.
- Given N (column) vectors $x_1, x_2, \ldots, x_N$, we denote by $\mathrm{col}(x_1, x_2, \ldots, x_N) \coloneqq \begin{bmatrix} x_1^\top & x_2^\top & \ldots & x_N^\top \end{bmatrix}^\top$ the concatenated vector.
- Given N matrices $A_1, A_2, \ldots, A_N$, we denote by $\mathrm{diag}(A_1, A_2, \ldots, A_N)$ the block diagonal matrix with $A_1, A_2, \ldots, A_N$ on its diagonal.
- **Euclidean spaces**
  Given a symmetric positive definite matrix $\mathbb{R}^{q\times q} \ni \Phi \succ 0$:
  - $\langle x \mid y\rangle_\Phi = x^\top \Phi y$ is the Φ-weighted inner product;
  - $\|\cdot\|_\Phi$ is the Φ-induced norm, i.e., $\|x\|_\Phi^2 = \langle x \mid x\rangle_\Phi$;
  - $\mathcal{H}_\Phi \coloneqq (\mathbb{R}^q, \langle\cdot \mid \cdot\rangle_\Phi)$ is the Euclidean space obtained by endowing $\mathbb{R}^q$ with the Φ-weighted inner product;
  - For a matrix $A \in \mathbb{R}^{p\times q}$, its Φ-induced norm is $\|A\|_\Phi \coloneqq \sup_{x\neq\mathbf{0}} \frac{\|Ax\|_\Phi}{\|x\|_\Phi}$.

  In all cases, we omit the subscripts if $\Phi = I$. Unless otherwise stated, we assume to work in $\mathcal{H} = \mathcal{H}_I$. We emphasize that in this dissertation we always work in finite dimensional spaces.

B

GRAPH THEORY

GRAPHS are used to mathematically model pairwise relations between objects. In this appendix, we recall relevant notation, definitions and results from graph theory; we refer to [70] for a complete discussion.

B.1. DEFINITIONS

Let us start by introducing the main definitions and notation used in this thesis.

- **Graph**
  A (directed) graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a nonempty set of vertices (or nodes) $\mathcal{V} = \{1, 2, \dots, N\}$ and a set of edges (or links) $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. We emphasize that each edge $(v, u) \in \mathcal{E}$ is an ordered pair. If $(u, v) \in \mathcal{E}$ whenever $(v, u) \in \mathcal{E}$, then we say that $\mathcal{G}$ is undirected.

- **Operations on graphs**
  We use roman font superscripts to distinguish between different graphs and the corresponding quantities (e.g., vertices, edges), as in $\mathcal{G}^{\mathrm{A}} = (\mathcal{V}^{\mathrm{A}}, \mathcal{E}^{\mathrm{A}})$ and $\mathcal{G}^{\mathrm{B}} = (\mathcal{V}^{\mathrm{B}}, \mathcal{E}^{\mathrm{B}})$. We write $\mathcal{G}^{\mathrm{A}} \subseteq \mathcal{G}^{\mathrm{B}}$ if $\mathcal{G}^{\mathrm{A}}$ is a subgraph of $\mathcal{G}^{\mathrm{B}}$, i.e., if $\mathcal{V}^{\mathrm{A}} \subseteq \mathcal{V}^{\mathrm{B}}$ and $\mathcal{E}^{\mathrm{A}} \subseteq \mathcal{E}^{\mathrm{B}}$. We define $\mathcal{G}^{\mathrm{A}} \bigcup \mathcal{G}^{\mathrm{B}} := (\mathcal{V}^{\mathrm{A}} \cup \mathcal{V}^{\mathrm{B}}, \mathcal{E}^{\mathrm{A}} \cup \mathcal{E}^{\mathrm{B}})$, and $\mathcal{G}^{\mathrm{A}} \bigcap \mathcal{G}^{\mathrm{B}} := (\mathcal{V}^{\mathrm{A}} \cap \mathcal{V}^{\mathrm{B}}, \mathcal{E}^{\mathrm{A}} \cap \mathcal{E}^{\mathrm{B}})$. The restriction of the graph $\mathcal{G}$ to a set of vertices $\mathcal{V}^{\mathrm{A}} \subseteq \mathcal{V}$ is defined as $\mathcal{G}|_{\mathcal{V}^{\mathrm{A}}} := (\mathcal{V}^{\mathrm{A}}, \mathcal{E} \cap (\mathcal{V}^{\mathrm{A}} \times \mathcal{V}^{\mathrm{A}}))$.

- **Neighbors, paths and connectedness**
  Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we denote by $\mathcal{N}_v := \{u \mid (u, v) \in \mathcal{E}\}$ and $\overline{\mathcal{N}}_v := \{u \mid (v, u) \in \mathcal{E}\}$ the set of in-neighbors (or simply neighbors) and out-neighbors of vertex $v \in \mathcal{V}$, respectively. A path from $v_1 \in \mathcal{V}$ to $v_N \in \mathcal{V}$ of length T is a sequence of vertices $(v_1, v_2, \dots, v_T)$ such that $(v_t, v_{t+1}) \in \mathcal{E}$ for all $t = 1, \dots, T-1$. $\mathcal{G}$ is rooted at $v \in \mathcal{V}$ if there exists a path from v to each $u \in \mathcal{V} \backslash \{v\}$; we say that $\mathcal{G}$ is rooted if there exists $v \in \mathcal{V}$ such that $\mathcal{G}$ is rooted at v. $\mathcal{G}$ is strongly connected if there exist a path from u to v, for all $u, v \in \mathcal{V}$; if $\mathcal{G}$ is undirected, we simply say that $\mathcal{G}$ is connected. Clearly, a

connected graph is strongly connected, and a strongly connected graph is rooted. A sequence of graphs $(\mathcal{G}_k)_{k\in\mathbb{N}}$, where $\mathcal{G}_k = (\mathcal{V}, \mathcal{E}_k)$, is called Q-strongly connected if the joint graph $\bigcup_{t=kQ}^{(k+1)Q-1} \mathcal{G}_k$ is strongly connected for all $k \in \mathbb{N}$.

- **Bipartite graphs**
  A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is called bipartite if there exist two sets $\mathcal{V}^{\mathrm{A}}, \mathcal{V}^{\mathrm{B}} \subset \mathcal{V}$ such that $\mathcal{V}^{\mathrm{A}} \cup \mathcal{V}^{\mathrm{B}} = \mathcal{V}$, $\mathcal{V}^{\mathrm{A}} \cap \mathcal{V}^{\mathrm{B}} = \varnothing$ and $\mathcal{E} \subseteq \mathcal{V}^{\mathrm{A}} \times \mathcal{V}^{\mathrm{B}}$, namely there are only edges from vertices in $\mathcal{V}^{\mathrm{A}}$ to vertices in $\mathcal{V}^{\mathrm{B}}$. We also write $\mathcal{G} = (\mathcal{V}^{\mathrm{A}}, \mathcal{V}^{\mathrm{B}}, \mathcal{E})$ to highlight that $\mathcal{G}$ is bipartite.

- **Graph weights and matrices**
  Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $\mathcal{V} = \{1, 2, \dots, N\}$. We may associate to $\mathcal{G}$ a weight matrix $W \in \mathbb{R}_{\geq 0}^{N\times N}$ compliant with $\mathcal{G}$, namely

$$w_{u,v} \coloneqq [W]_{u,v} > 0 \Leftrightarrow (v,u) \in \mathcal{E},$$

  $w_{u,v} = 0$ otherwise. We denote by

$$\begin{aligned} \deg(v) &= \textstyle\sum_{u\in\mathcal{V}} w_{u,v} \\ \overline{\deg}(v) &= \textstyle\sum_{v\in\mathcal{V}} w_{v,u} \end{aligned}$$

  the in-degree and out degree of vertex v, respectively. Let

$$\begin{aligned} D &= \operatorname{diag}\big((\deg(v))\, v \in \mathcal{V}\big) = \operatorname{diag}\big(\deg(1), \deg(2), \dots \deg(N)\big) \in \mathbb{R}_{\geq 0}^{N\times N} \\ L &= D - W \in \mathbb{R}^{N\times N} \end{aligned}$$

  be the in-degree and Laplacian matrices, respectively. A graph with associated weight matrix W is called:

  - *unweighted*, if $w_{u,v} = 1$ whenever $(v,u) \in \mathcal{E}$;
  - *balanced*, if $\deg(v) = \overline{\deg}(v)$, for all $v \in \mathcal{V}$;
  - *doubly stochastic*, if $\deg(v) = \overline{\deg}(v) = 1$, for all $v \in \mathcal{V}$;
  - *row stochastic*, if $\deg(v) = 1$, for all $v \in \mathcal{V}$ (equivalently, $W\mathbf{1}_N = \mathbf{1}_N$);
  - *column stochastic*, if $\overline{\deg}(v) = 1$, for all $v \in \mathcal{V}$ (equivalently, $\mathbf{1}_N^\top W = \mathbf{1}_N^\top$).

  Clearly, a doubly stochastic graph is both row stochastic and column stochastic; furthermore it is balanced. A graph with symmetric weight matrix $W = W^\top$ is balanced: this is the situation commonly considered for undirected graphs.

- **Weighted incidence matrix for undirected graphs**
  Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph, with associated weight matrix $W \in \mathbb{R}_{\geq 0}^{N\times N}$. Assume that the graph has no self-loops, i.e., $(v,v) \notin \mathcal{E}$ and $w_{v,v} = 0$, for all $v \in \mathcal{V}$. Denote by $\bar{\mathcal{E}} \subset \mathcal{E}$ an arbitrary set of edges obtained by removing from $\mathcal{E}$ one edge from each couple $(u,v), (v,u) \in \mathcal{E}$ (i.e., selecting only one direction for each bidirectional edge). Furthermore, label the edges in $\bar{\mathcal{E}}$ as $(e_\ell)_{\ell\in\{1,\dots,E\}}$, where E is the cardinality of $\bar{\mathcal{E}}$. We define the weighted incidence matrix $V \in \mathbb{R}^{E\times N}$, as $[V]_{\ell,u} = \sqrt{(w_{u,v})}$ if $e_\ell = (u,v)$, $[V]_{\ell,u} = -\sqrt{(w_{u,v})}$ if $e_\ell = (v,u)$, $[V]_{\ell,u} = 0$ otherwise.

B.2. PROPERTIES

We next recall some well-known graph-theoretic properties that are used throughout the dissertation. The first result is based on the Perron–Frobenius theorem.

Lemma B.1 (*Properties of the weight matrix*)**.** Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with compliant weight matrix $W \in \mathbb{R}^{N\times N}_{\geq 0}$, and assume that W is row stochastic. Then, its spectral radius $\rho(W) = 1$. Also, 1 is an eigenvalue of W, $W\mathbf{1}_N = \mathbf{1}_N$, and there exists a nonnegative vector $w \in \mathbb{R}^N_{\geq 0}$ such that $w^\top W = w^\top$, with $w^\top \mathbf{1}_N = 1$. Furthermore, the following statements hold:

- If 1 is a simple eigenvalue of W and every other eigenvalue has absolute value strictly smaller than 1, then $\lim_{k\to\infty} W^k = \mathbf{1}_N w^\top$; furthermore, there exists $\Phi \succ 0$ such that $\|W - \mathbf{1}_N w^\top\|_\Phi < 1$. A necessary condition for the hypothesis to hold is that $\mathcal{G}$ is rooted;
- If $\mathcal{G}$ is strongly connected and $w_{v,v} > 0$ for all $v \in \mathcal{V}$, then 1 is a simple eigenvalue of W, every other eigenvalue has absolute value strictly smaller than 1, w is strictly positive, $\lim_{k\to\infty} W^k = \mathbf{1}_N w^\top$; furthermore, there exists $\Phi \succ 0$ such that $\|W - \mathbf{1}_N w^\top\|_\Phi < 1$;
- if $\mathcal{G}$ is strongly connected, W is doubly stochastic and $w_{v,v} > 0$ for all $v \in \mathcal{V}$, then $\lim_{k\to\infty} W^k = \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top$ and $\|W - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top\| < 1$; □

Lemma B.2 (*Properties of the Laplacian*)**.** Let $\mathcal{G}$ be a graph with Laplacian $L \in \mathbb{R}^{N\times N}$. Then, the following statements hold:

- $L\mathbf{1}_N = \mathbf{0}_N$;
- $\mathbf{1}_N^\top L = \mathbf{0}_N$ if and only if $\mathcal{G}$ is balanced;
- $\text{null}(L) = \{\kappa\mathbf{1}_N, \kappa \in \mathbb{R}\}$ if and only if $\mathcal{G}$ is rooted: in this case, L has a simple eigenvalue in zero, and the other eigenvalues have positive real parts;
- if $\mathcal{G}$ is undirected and connected, and $L = L^\top$, then $L \succcurlyeq 0$ and L has a simple eigenvalue in zero, thus $\lambda_2(L) > 0$ (the quantity $\lambda_2(L)$ is named algebraic connectivity).
- if $\mathcal{G}$ is strongly connected and balanced, then $\lambda_2\left(\frac{L+L^\top}{2}\right) > 0$ and $x^\top\left(\frac{L+L^\top}{2}\right)x \geq \lambda_2\left(\frac{L+L^\top}{2}\right)\left\|(I - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top)x\right\|^2$, for all $x \in \mathbb{R}^N$. □

Lemma B.3 (*Properties of the incidence matrix*)**.** Let $\mathcal{G}$ be an undirected graph with Laplacian $L = L^\top \in \mathbb{R}^{N\times N}$. Let $V \in \mathbb{R}^{E\times N}$ be the associate weighted incidence matrix. Then, $L = V^\top V$, and $V\mathbf{1}_N = \mathbf{0}_E$. Furthermore, if $\mathcal{G}$ is connected, then $\text{null}(V) = \text{null}(L) = \{\kappa\mathbf{1}_N, \kappa \in \mathbb{R}\}$. □

B.3. STEINER PROBLEMS

The term Steiner problems refers to a class of problems in combinatorial optimization; here we introduce some particular instances that are used in Chapter 7. Specifically, given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, compliant weights $W \in \mathbb{R}^{N\times N}_{\geq 0}$, a root $r \in \mathcal{V}$ and a set of terminals $\mathcal{T} \subseteq \mathcal{V}$, we define:

- *Steiner tree problem* ST($\mathcal{G}, \mathcal{T}, W$): find an undirected connected subgraph $\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*) \subseteq \mathcal{G}$, $\mathcal{T} \subseteq \mathcal{V}^*$, with minimum cost (i.e., minimizing $\sum_{(v,u)\in\mathcal{E}^*} w_{u,v}$);
- *Unweighted Steiner tree problem,* UST($\mathcal{G}, \mathcal{T}$): find an undirected connected subgraph $\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*) \subseteq \mathcal{G}$, $\mathcal{T} \subseteq \mathcal{V}^*$, with minimum number of edges;
- *Directed Steiner tree problem,* DST($\mathcal{G}, r, \mathcal{T}, W$): find a subgraph $\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*) \subseteq \mathcal{G}$, $\mathcal{T} \subseteq \mathcal{V}^*$, rooted at r, with minimum cost (i.e., minimizing $\sum_{(v,u)\in\mathcal{E}^*} w_{u,v}$);
- *Unweighted directed Steiner tree problem,* UDST($\mathcal{G}, r, \mathcal{T}$): find a subgraph $\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*) \subseteq \mathcal{G}$, $\mathcal{T} \subseteq \mathcal{V}^*$, rooted at r, with minimum number of edges;
- *Strongly connected Steiner subgraph problem,* SCSS($\mathcal{G}, \mathcal{T}$): find a strongly connected subgraph $\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*) \subseteq \mathcal{G}$, $\mathcal{T} \subseteq \mathcal{V}^*$, with minimal number of vertices.

C

OPERATOR THEORY AND FIXED POINT ITERATIONS

IN this appendix, we review some selected concepts and theorems from fixed point theory and operator theory, including the notions of monotone and nonexpansive operators. The interested reader can find an exhaustive collection of results in [8], [113]. We will assume that the reader is familiar with standard convex analysis.

C.1. OPERATORS

The notation $\mathcal{F} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$ means that the set-valued mapping $\mathcal{F}$, or *operator*, maps every point $x \in \mathbb{R}^q$ to a (possibly empty) set $\mathcal{F}(x) \subseteq \mathbb{R}^q$. An operator $\mathcal{F} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$ is characterized by its *graph*,

$$\operatorname{gra}(\mathcal{F}) := \{(x, u) \mid u \in \mathcal{F}(x)\}.$$

For an operator $\mathcal{F} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$, we also define its:

- *domain*, $\operatorname{dom}(\mathcal{F}) := \left\{x \in \mathbb{R}^q \mid \mathcal{F}(x) \neq \varnothing\right\}$;
- *set of fixed points*, $\operatorname{fix}(\mathcal{F}) := \left\{x \in \mathbb{R}^q \mid x \in \mathcal{F}(x)\right\}$;
- *set of zeros*, $\operatorname{zer}(\mathcal{F}) := \left\{x \in \mathbb{R}^q \mid 0 \in \mathcal{F}(x)\right\}$;
- *inverse operator*, $\mathcal{F}^{-1} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$, defined via its graph as $\operatorname{gra}\left(\mathcal{F}^{-1}\right) = \left\{(u, x) \mid (x, u) \in \operatorname{gra}(\mathcal{F})\right\}$.

Given two operators $\mathcal{F} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$ and $\mathcal{A} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$, and a scalar $\lambda \in \mathbb{R}$, the operator $\mathcal{F} + \lambda\mathcal{A}$ is defined by

$$\operatorname{gra}(\mathcal{F} + \lambda\mathcal{A}) = \left\{(x, u + \lambda v) \mid (x, u) \in \operatorname{gra}(\mathcal{F}), (x, v) \in \operatorname{gra}(\mathcal{A})\right\}.$$

In particular, note that $\operatorname{dom}(\mathcal{F}+\lambda\mathcal{A}) = \operatorname{dom}(\mathcal{F}) \cap \operatorname{dom}(\mathcal{A})$. We use the notation $\mathcal{F}: \mathcal{X} \to \mathbb{R}^q$ to highlight that the operator $\mathcal{F}$ is single-valued on the its domain $\mathcal{X} := \operatorname{dom}(\mathcal{F})$, i.e., $\mathcal{F}(x)$ is a singleton for all $x \in \mathcal{X}$. We denote by Id the identity operator, i.e., $\operatorname{Id}: \mathbb{R}^q \to \mathbb{R}^q$,

$$(\forall x \in \mathbb{R}^q) \quad \operatorname{Id}(x) = x.$$

C.2. NONEXPANSIVE OPERATORS

Let $\mathcal{H}_\Phi = (\mathbb{R}^q, \langle \cdot \mid \cdot \rangle_\Phi)$, be the Φ-weighted Euclidean space, with $\Phi \succ 0$. Let $\mathcal{X} \subseteq \mathbb{R}^q$ be a nonempty set. The single-valued operator $\mathcal{F}: \mathcal{X} \to \mathbb{R}^q$ is:

- *Lipschitz continuous* in $\mathcal{H}_\Phi$ with constant $\theta \in \mathbb{R}_{>0}$, if

$$(\forall x \in \mathcal{X})(\forall y \in \mathcal{X}) \quad \|\mathcal{F}(x) - \mathcal{F}(y)\|_\Phi \le \theta \|x - y\|_\Phi;$$

- *Nonexpansive* in $\mathcal{H}_\Phi$, if it is 1-Lipschitz continuous in $\mathcal{H}_\Phi$, i.e.,

$$(\forall x \in \mathcal{X})(\forall y \in \mathcal{X}) \quad \|\mathcal{F}(x) - \mathcal{F}(y)\|_\Phi \le \|x - y\|_\Phi;$$

- *Contractive* in $\mathcal{H}_\Phi$, if it is θ-Lipschitz continuous in $\mathcal{H}_\Phi$ with $\theta < 1$, i.e.,

$$(\exists \theta \in (0,1))(\forall x \in \mathcal{X})(\forall y \in \mathcal{X}) \quad \|\mathcal{F}(x) - \mathcal{F}(y)\|_\Phi \le \theta \|x - y\|_\Phi;$$

- *Averaged* in $\mathcal{H}_\Phi$ with parameter $\alpha \in (0,1)$, if there exists an operator $\mathcal{B}: \mathcal{X} \to \mathbb{R}^q$ that is nonexpansive in $\mathcal{H}_\Phi$, such that $\mathcal{F} = (1-\alpha)\operatorname{Id} + \alpha\mathcal{B}$. Equivalently, if it holds that

$$(\forall x \in \mathcal{X})(\forall y \in \mathcal{X}) \quad \|\mathcal{F}(x) - \mathcal{F}(y)\|_\Phi^2 \le \|x - y\|_\Phi^2 - \tfrac{1-\alpha}{\alpha} \|(\operatorname{Id} - \mathcal{F})(x) - (\operatorname{Id} - \mathcal{F})(y)\|_\Phi^2;$$

- *Firmly nonexpansive* in $\mathcal{H}_\Phi$, if it is $\frac{1}{2}$-averaged in $\mathcal{H}_\Phi$, i.e.,

$$(\forall x \in \mathcal{X})(\forall y \in \mathcal{X}) \quad \|\mathcal{F}(x) - \mathcal{F}(y)\|_\Phi^2 \le \|x - y\|_\Phi^2 - \|(\operatorname{Id} - \mathcal{F})(x) - (\operatorname{Id} - \mathcal{F})(y)\|_\Phi^2;$$

- *Quasi-nonexpansive* in $\mathcal{H}_\Phi$, if it is nonexpansive in $\mathcal{H}_\Phi$ w.r.t. $\operatorname{fix}(\mathcal{F})$, i.e.,

$$(\forall x \in \mathcal{X})(\forall y \in \operatorname{fix}(\mathcal{F})) \quad \|\mathcal{F}(x) - \mathcal{F}(y)\|_\Phi \le \|x - y\|_\Phi;$$

- *Firmly quasi-nonexpansive* in $\mathcal{H}_\Phi$, if it is firmly nonexpansive in $\mathcal{H}_\Phi$ w.r.t. $\operatorname{fix}(\mathcal{F})$, i.e.,

$$(\forall x \in \mathcal{X})(\forall y \in \operatorname{fix}(\mathcal{F})) \quad \|\mathcal{F}(x) - \mathcal{F}(y)\|_\Phi^2 \le \|x - y\|_\Phi^2 - \|(\operatorname{Id} - \mathcal{F})(x)\|_\Phi^2;$$

- *Inverse Lipschitz* in $\mathcal{H}_\Phi$, with constant $R \in \mathbb{R}_{>0}$, if $\mathcal{F}^{-1}$ is R-Lipschitz continuous in $\mathcal{H}_\Phi$, i.e.,

$$(\forall x \in \mathcal{X})(\forall y \in \mathcal{X}) \quad R\|\mathcal{F}(x) - \mathcal{F}(y)\|_\Phi \ge \|x - y\|_\Phi.$$

In all cases, whenever $\Phi = I$, we omit the indication "in $\mathcal{H}_\Phi$."

C.3. MONOTONE OPERATORS

Let $\mathcal{H}_\Phi = (\mathbb{R}^q, \langle\cdot \mid \cdot\rangle_\Phi)$ be the Φ-weighted Euclidean space, with $\Phi \succ 0$. A set-valued operator $\mathcal{F} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$ is:

- *Monotone* in $\mathcal{H}_\Phi$, if

$$(\forall (x,u) \in \operatorname{gra}(\mathcal{F}))(\forall (y,v) \in \operatorname{gra}(\mathcal{F})) \quad \langle x-y \mid u-v\rangle_\Phi \geq 0;$$

- *Maximally monontone* in $\mathcal{H}_\Phi$, if it is monotone in $\mathcal{H}_\Phi$ and there exists no operator $\mathcal{B} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$ monotone in $\mathcal{H}_\Phi$ such that $\operatorname{gra}(\mathcal{F})$ is a strict subset of $\operatorname{gra}(\mathcal{B})$;

- *Strictly monotone* in $\mathcal{H}_\Phi$, if

$$(\forall (x,u) \in \operatorname{gra}(\mathcal{F}))(\forall (y,v) \in \operatorname{gra}(\mathcal{F})) \quad x \neq y \Rightarrow \langle x-y \mid u-v\rangle_\Phi > 0;$$

- *Strongly monotone* in $\mathcal{H}_\Phi$ with constant $\mu \in \mathbb{R}_{>0}$, if

$$(\forall (x,u) \in \operatorname{gra}(\mathcal{F}))(\forall (y,v) \in \operatorname{gra}(\mathcal{F})) \quad \langle x-y \mid u-v\rangle_\Phi \geq \mu \|x-y\|_\Phi^2;$$

- *Hypomonotone* in $\mathcal{H}_\Phi$ with constant $\nu \in \mathbb{R}_{>0}$, if

$$(\forall (x,u) \in \operatorname{gra}(\mathcal{F}))(\forall (y,v) \in \operatorname{gra}(\mathcal{F})) \quad \langle x-y \mid u-v\rangle_\Phi \geq -\nu \|x-y\|_\Phi^2.$$

Furthermore, a single-valued operator $\mathcal{F} : \mathbb{R}^q \to \mathbb{R}^q$ is

- *Cocoercive* in $\mathcal{H}_\Phi$ with constant $\beta \in \mathbb{R}_{>0}$, if $\beta\mathcal{F}$ is firmly nonexpansive in $\mathcal{H}_\Phi$. Equivalently, if $\mathcal{F}^{-1}$ is β-strongly monotone in $\mathcal{H}_\Phi$. Equivalently, if it holds that

$$(\forall x \in \mathbb{R}^q)(\forall y \in \mathbb{R}^q) \quad \langle x-y \mid \mathcal{F}(x)-\mathcal{F}(y)\rangle_\Phi \geq \beta \|\mathcal{F}(x)-\mathcal{F}(y)\|_\Phi^2.$$

In all cases, whenever $\Phi = I$, we omit the indication "in $\mathcal{H}_\Phi$".

C.4. EXAMPLES AND RELEVANT OPERATORS

We present next some examples of monotone and nonexpansive operators, and we introduce some important operators, of major interest for this dissertation.

- The identity operator Id is firmly-nonexpansive and 1-strongly monotone, with $\operatorname{fix}(\mathrm{Id}) = \mathbb{R}^q$, and $\operatorname{zer}(\mathrm{Id}) = \{\mathbf{0}\}$;

- $\mathcal{F} = -\mathrm{Id}$ is nonexpansive and 1-hypomonotone, with $\operatorname{fix}(\mathcal{F}) = \operatorname{zer}(\mathcal{F}) = \{\mathbf{0}\}$;

- **Subdifferential**
  Consider a function $\psi : \mathbb{R}^q \to \overline{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$; we denote by $\operatorname{dom}(\psi) := \{x \in \mathbb{R}^q \mid \psi(x) < \infty\}$ its domain. Its subdifferential operator $\partial\psi : \operatorname{dom}(\psi) \rightrightarrows \mathbb{R}^q$ is defined as

$$\partial\psi(x) := \{v \in \mathbb{R}^q \mid \psi(z) \geq \psi(x) + \langle v \mid z-x\rangle, \forall z \in \operatorname{dom}(\psi)\}.$$

$\partial\psi$ is a monotone operator, for any function ψ. Furthermore, if ψ is closed, convex and proper, $\partial\psi$ is a maximally monotone operator. The set $\partial\psi(x)$ is always closed and convex, but it can be empty. However, if ψ is convex, then $\partial\psi(x) \neq \varnothing$ for all $x \in \operatorname{relint}(\operatorname{dom}(\psi))$, where relint denotes the relative interior. Furthermore, if ψ is differentiable and convex, then $\partial\psi$ is a single valued operator, and it coincides with the gradient of ψ, i.e., $\partial\psi = \nabla\psi$. Conversely, the gradient $\nabla\psi$ of a differentiable function $\psi : \mathbb{R}^q \to \mathbb{R}^q$ is monotone if and only if ψ is convex (we emphasize that we only consider gradients and subdifferentials in $\mathcal{H}_I$ in this dissertation);

- **Normal cone**
  For a set $S \subseteq \mathbb{R}^q$, let us denote by $\iota_S : \mathbb{R}^q \to \overline{\mathbb{R}}$ the indicator function of S, i.e., $\iota_S(x) = 0$ if $x \in S$, ∞ otherwise. The operator $\mathrm{N}_S : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$, defined as

$$\mathrm{N}_S(x) := \begin{cases} \varnothing, & \text{if} \quad x \notin S \\ \left\{ v \in \mathbb{R}^q \mid \sup_{z \in S} \langle v \mid z - x \rangle \leq 0 \right\}, & \text{otherwise} \end{cases}$$

  is called the normal cone of S. If S is closed and convex, then $\partial\iota_S = \mathrm{N}_S$, hence N_S is a maximally monotone operator.

- **Resolvent**
  The resolvent operator $\mathrm{J}_{\mathcal{F}} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$ of an operator $\mathcal{F} : \mathbb{R}^q \rightrightarrows \mathbb{R}^q$ is

$$\mathrm{J}_{\mathcal{F}} := (\mathrm{Id} + \mathcal{F})^{-1}.$$

  If $\mathcal{F}$ is maximally monotone in $\mathcal{H}_\Phi$, then $\mathrm{J}_{\mathcal{F}}$ is single-valued, $\operatorname{dom}(\mathrm{J}_{\mathcal{F}}) = \mathbb{R}^q$, and $\mathrm{J}_{\mathcal{F}}$ is firmly nonexpansive in $\mathcal{H}_\Phi$.

- **Projection**
  If $S \subseteq \mathbb{R}^q$ is closed and convex, then the resolvent of the normal cone equals the Euclidean projection onto S, i.e.,

$$\mathrm{J}_{\mathrm{N}_S} = (\mathrm{Id} + \mathrm{N}_S)^{-1} = \mathrm{proj}_S,$$

  where $\mathrm{proj}_S : \mathbb{R}^q \to \mathbb{R}^q$, $\mathrm{proj}_S(x) = \operatorname{argmin}_{\xi \in S} \|x - \xi\|$. It follows that proj_S is firmly nonexpansive.

C.5. FIXED POINTS OF NONEXPANSIVE OPERATORS

A variety of mathematical and engineering problems can be reformulated as the problem of finding a fixed point of an operator. In this section, we recall some fundamental results concerning the existence of such a fixed point, and algorithms for their computation (also called fixed point iterations).

Theorem C.1 (*Browder, [8, Th. 4.29]*)**.** Let $\mathcal{X} \subset \mathbb{R}^q$ be a nonempty compact set, and let $\mathcal{F} : \mathcal{X} \to \mathcal{X}$ be nonexpansive in some space $\mathcal{H}_\Phi$. Then, $\operatorname{fix}(\mathcal{F}) \neq \varnothing$. □

Theorem C.2 (*Banach [17, Th. 2.1]*)**.** Let $\mathcal{X} \subset \mathbb{R}^q$ be nonempty, and let $\mathcal{F} : \mathcal{X} \to \mathcal{X}$ be contractive in some space $\mathcal{H}_\Phi$. Then, $\operatorname{fix}(\mathcal{F})$ is a singleton. □

Theorem C.3 (*Banach–Picard iteration [17, Th. 2.1]*)**.** Let $\mathcal{X} \subset \mathbb{R}^q$ be nonempty, and let $\mathcal{F} : \mathcal{X} \to \mathcal{X}$ be θ-contractive in some space $\mathcal{H}_\Phi$, with $\theta \in (0,1)$. Then, for any initial consition $x^0 \in \mathcal{X}$, the sequence $(x^k)_{k\in\mathbb{N}}$ generated by the iteration

$$(\forall k \in \mathbb{N}) \quad x^{k+1} = \mathcal{F}(x^k),$$

converges to the unique fixed point $x^\star \in \mathrm{fix}(\mathcal{F})$ with linear rate, i.e., $\|x^k - x^\star\|_\Phi \leq \theta^k \|x^k - x^*\|_\Phi$, for all $k \in \mathbb{N}$. □

Theorem C.4 (*Krasnosel'skii–Maan iteration [8, Th. 5.15]*)**.** Let $\mathcal{X} \subset \mathbb{R}^q$ be nonempty, let $\mathcal{F} : \mathcal{X} \to \mathcal{X}$ be nonexpansive in some space $\mathcal{H}_\Phi$ and such that $\mathrm{fix}(\mathcal{F}) \neq \varnothing$. Let $(\gamma^k)_{k\in\mathbb{N}}$ be a sequence in $[0,1]$ such that $\sum_{k\in\mathbb{N}} \gamma^k(1-\gamma^k) = \infty$. Then, for any initial condition $x^0 \in \mathcal{X}$, the sequence $(x^k)_{k\in\mathbb{N}}$ generated by the iteration

$$(\forall k \in \mathbb{N}) \quad x^{k+1} = x^k + \gamma^k(\mathcal{F}(x^k) - x^k),$$

converges to a fixed point $x^\star \in \mathrm{fix}(\mathcal{F})$. □

Corollary C.1. Let $\mathcal{X} \subset \mathbb{R}^q$ be nonempty, and let $\mathcal{F} : \mathcal{X} \to \mathcal{X}$ be α-averaged in some space $\mathcal{H}_\Phi$, with $\alpha \in (0,1)$, and such that $\mathrm{fix}(\mathcal{F}) \neq \varnothing$. Then, for any initial condition $x^0 \in \mathcal{X}$, the sequence $(x^k)_{k\in\mathbb{N}}$ generated by the iteration

$$(\forall k \in \mathbb{N}) \quad x^{k+1} = \mathcal{F}(x^k),$$

converges to a fixed point $x^* \in \mathrm{fix}(\mathcal{F})$. □

C.6. ZEROS OF MONOTONE OPERATORS

The problem of finding a zero of (monotone) operators plays an important role in optimization, games and in general variational inequalities.

Example C.1 (*Variational inequality*)**.** Given a convex closed set $S \subseteq \mathbb{R}^q$ and a single-valued operator $\mathcal{F} : S \to \mathbb{R}^q$, the variational inequality $\mathrm{VI}(\mathcal{F}, S)$ is the problem of finding $x^\star \in S$ such that

$$\langle \mathcal{F}(x^\star) \mid x - x^\star \rangle \geq 0, \quad \forall x \in S.$$

By definition, the problem is equivalent to finding $x^\star$ such that

$$\mathbf{0} \in \mathcal{F}(x^\star) + \mathrm{N}_S(x^\star)$$

i.e., finding a zero of the operator $\mathcal{F} + \mathrm{N}_S$. □

Zero-finding problems are also strictly related to fixed point problems, since $\mathrm{zer}(\mathcal{F}) = \mathrm{fix}(\mathrm{Id} - \mathcal{F})$ for any operator $\mathcal{F}$. In fact, the solution to both problems is typically based on fixed point iterations, where monotonicity/nonexpansiveness properties are exploited to prove convergence. We present two examples in the remainder of this section.

C.6.1. PROXIMAL-POINT ALGORITHM

Given an operator $\mathcal{F}$, consider the problem of finding $x^\star$ such that

$$\mathbf{0} \in \mathcal{F}(x^\star).$$

Note that

$$\begin{aligned} \mathbf{0} \in \mathcal{F}(x^\star) &\Leftrightarrow x^\star \in (\mathrm{Id}+\mathcal{F})x^\star \\ &\Leftrightarrow (\mathrm{Id}+\mathcal{F})^{-1} \ni x^\star \\ &\Leftrightarrow x^\star \in \mathrm{fix}(\mathrm{J}_{\mathcal{F}}). \end{aligned}$$

When $\mathcal{F}$ is maximally monotone, a solution can be found based on Corollary C.1, since $\mathrm{J}_{\mathcal{F}}$ is firmly nonexpansive (hence, $\frac{1}{2}$ averaged).

Theorem C.5 (*Proximal-point algorithm [8, Th. 23.41]*)**.** Let $\mathcal{F}:\mathbb{R}^q \rightrightarrows \mathbb{R}^q$ be maximally monotone operator such that $\mathrm{zer}(\mathcal{F}) \neq \varnothing$. Then, for any initial condition $x^0 \in \mathbb{R}^q$, the sequence $(x^k)_{k\in\mathbb{N}}$ generated by the proximal-point iteration

$$(\forall\, k \in \mathbb{N}) \quad x^{k+1} = \mathrm{J}_{\mathcal{F}}(x^k)$$

converges to some $x^\star \in \mathrm{zer}(\mathcal{F})$. □

C.6.2. FORWARD-BACKWARD ALGORITHM

Given two operators $\mathcal{F}$ and $\mathcal{B}$, consider the problem of finding $x^\star$ such that

$$\mathbf{0} \in \mathcal{F}(x^\star) + \mathcal{B}(x^\star).$$

The problem could be solved via the proximal-point algorithm applied to the operator $(\mathcal{F}+\mathcal{B})$. However, computing the resolvent of the latter operator might be complex; instead, it is often convenient to devise methods where the operators $\mathcal{F}$ and $\mathcal{B}$ are employed at different computational steps. This general idea motivated the development of numerous *operator splitting* methods. We provide an example next. It can be shown that

$$x^\star \in \mathrm{zer}(\mathcal{F}+\mathcal{B}) \Leftrightarrow x^\star \in \mathrm{J}_{\gamma\mathcal{B}}(x^\star - \gamma\mathcal{F}(x^\star))$$

for any $\gamma \in \mathbb{R}_{>0}$. This fact suggests the so-called forward-backward algorithm.

Theorem C.6 (*Forward-backward algorithm [8, p. 26.14]*)**.** Let $\mathcal{B}:\mathbb{R}^q \rightrightarrows \mathbb{R}^q$ be maximally monotone; let $\mathcal{F}$ be β-cocoercive, for some $\beta \in \mathbb{R}_{>0}$; let $\gamma \in (0, 2\beta)$. Assume that $\mathrm{zer}(\mathcal{F}+\mathcal{B}) \neq \varnothing$. Then, for any initial condition $x^0 \in \mathbb{R}^q$, the sequence $(x^k)_{k\in\mathbb{N}}$ generated by the forward-backward iteration

$$(\forall\, k \in \mathbb{N}) \quad \begin{cases} y^k = x^k - \gamma\mathcal{B}(x^k) \\ x^{k+1} = \mathrm{J}_{\gamma\mathcal{F}}(y^k) \end{cases}$$

converges to some $x^\star \in \mathrm{zer}(\mathcal{F}+\mathcal{B})$. □

BIBLIOGRAPHY

[1] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey," *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, 2006. DOI: 10.1016/j.comnet.2006.05.001.

[2] S. A. Alghunaim and A. H. Sayed, "Distributed coupled multiagent stochastic optimization," *IEEE Transactions on Automatic Control*, vol. 65, no. 1, pp. 175–190, 1 2020. DOI: 10.1109/TAC.2019.2906495.

[3] S. A. Alghunaim, K. Yuan, and A. H. Sayed, "A proximal diffusion strategy for multiagent optimization with sparse affine constraints," *IEEE Transactions on Automatic Control*, vol. 65, no. 11, pp. 4554–4567, 11 2020. DOI: 10.1109/TAC.2019.2960265.

[4] T. Alpcan and T. Basar, "A game-theoretic framework for congestion control in general topology networks," in *41st IEEE Conference on Decision and Control (CDC)*, vol. 2, 2002, pp. 1218–1224. DOI: 10.1109/CDC.2002.1184680.

[5] W. Ananduta, J. Barreiro-Gomez, C. Ocampo-Martinez, and N. Quijano, "Mitigation of communication failures in distributed model predictive control strategies," *IET Control Theory Applications*, vol. 12, pp. 2507–2515, 18 2018. DOI: 10.1049/iet-cta.2018.5044.

[6] J. Barrera and A. Garcia, "Dynamic incentives for congestion control," *IEEE Transactions on Automatic Control*, vol. 60, no. 2, pp. 299–310, 2015. DOI: 10.1109/TAC.2014.2348197.

[7] N. Bastianello, R. Carli, L. Schenato, and M. Todescato, "Asynchronous distributed optimization over lossy networks via relaxed ADMM: Stability and linear convergence," *IEEE Transactions on Automatic Control*, vol. 66, no. 6, pp. 2620–2635, 6. DOI: 10.1109/TAC.2020.3011358.

[8] H. H. Bauschke and P. L. Combettes, *Convex analysis and monotone operator theory in Hilbert spaces*. Springer New York, 2017, vol. 2011. DOI: 10.1007/978-1-4419-9467-7.

[9] G. Belgioioso and S. Grammatico, "Semi-decentralized Nash equilibrium seeking in aggregative games with separable coupling constraints and non-differentiable cost functions," *IEEE Control Systems Letters*, vol. 1, no. 2, pp. 400–405, 2017. DOI: 10.1109/LCSYS.2017.2718842.

[10] G. Belgioioso, A. Nedić, and S. Grammatico, "Distributed generalized Nash equilibrium seeking in aggregative games on time-varying networks," *IEEE Transactions on Automatic Control*, vol. 66, no. 5, pp. 2061–2075, 2021. DOI: 10.1109/TAC.2020.3005922.

[11] G. Belgioioso and S. Grammatico, "A distributed proximal-point algorithm for Nash equilibrium seeking in generalized potential games with linearly coupled cost functions," in *2019 18th European Control Conference (ECC)*, 2019, pp. 1–6. DOI: 10.23919/ECC.2019.8795852.

[12] G. Belgioioso and S. Grammatico, "A Douglas-Rachford splitting for semi-decentralized equilibrium seeking in generalized aggregative games," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 3541–3546. DOI: 10.1109/CDC.2018.8619610.

[13] G. Belgioioso and S. Grammatico, "Projected-gradient algorithms for generalized equilibrium seeking in aggregative games are preconditioned forward-backward methods," in *2018 European Control Conference (ECC)*, 2018, pp. 2188–2193. DOI: 10.23919/ECC.2018.8550342.

[14] G. Belgioioso and S. Grammatico, "Semi-decentralized generalized Nash equilibrium seeking in monotone aggregative games," *IEEE Transactions on Automatic Control*, 2021. DOI: 10.1109/TAC.2021.3135360.

[15] G. Belgioioso, D. Liao-McPherson, M. H. de Badyn, S. Bolognani, J. Lygeros, and F. Dörfler, "Sampled-data online feedback equilibrium seeking: Stability and tracking," in *2021 60th IEEE Conference on Decision and Control (CDC)*, 2021, pp. 2702–2708. DOI: 10.1109/CDC45484.2021.9683614.

[16] E. Benenati, W. Ananduta, and S. Grammatico, *Optimal selection and tracking of generalized Nash equilibria in monotone games*, 2022. [Online]. Available: https://arxiv.org/abs/2203.07765.

[17] V. Berinde, *Iterative Approximation of Fixed Points*. Springer Berlin, Heidelberg, 2007. DOI: doi.org/10.1007/978-3-540-72234-2.

[18] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Prentice-Hall, USA, 1989. [Online]. Available: http://athenasc.com/pdcbook.html.

[19] M. Bianchi, A. van der Maas, E. Maljaars, and W. Heemels, "Offset-free MPC for resource sharing on a nonlinear SCARA robot," in *6th IFAC Conference on Nonlinear Model Predictive Control NMPC 2018*, vol. 51, 2018, pp. 265–272. DOI: 10.1016/j.ifacol.2018.11.024.

[20] M. Bianchi, W. Ananduta, and S. Grammatico, "The distributed dual ascent algorithm is robust to asynchrony," *IEEE Control Systems Letters*, vol. 6, pp. 650–655, 2022. DOI: 10.1109/LCSYS.2021.3084883.

[21] M. Bianchi, G. Belgioioso, and S. Grammatico, "A fully-distributed proximal-point algorithm for Nash equilibrium seeking with linear convergence rate," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 2303–2308. DOI: 10.1109/CDC42340.2020.9304145.

[22] M. Bianchi, G. Belgioioso, and S. Grammatico, "Fast generalized Nash equilibrium seeking under partial-decision information," *Automatica*, vol. 136, p. 110 080, 2022. DOI: 10.1016/j.automatica.2021.110080.

[23] M. Bianchi and S. Grammatico, "A continuous-time distributed generalized Nash equilibrium seeking algorithm over networks for double-integrator agents," in *2020 European Control Conference (ECC)*, 2020, pp. 1474–1479. DOI: 10.23919/ECC51009.2020.9143714.

[24] M. Bianchi and S. Grammatico, "Continuous-time fully distributed generalized Nash equilibrium seeking for multi-integrator agents," *Automatica*, vol. 129, p. 109 660, 2021. DOI: 10.1016/j.automatica.2021.109660.

[25] M. Bianchi and S. Grammatico, "Fully distributed Nash equilibrium seeking over time-varying communication networks with linear convergence rate," *IEEE Control Systems Letters*, vol. 5, pp. 499–504, 2021. DOI: 10.1109/LCSYS.2020.3002734.

[26] M. Bianchi and S. Grammatico, "Nash equilibrium seeking under partial decision information: Monotonicity, smoothness and proximal-point algorithms," in *2022 61st IEEE Conference on Decision and Control (CDC)*, 2022, pp. 5080–5085. DOI: 10.1109/CDC51059.2022.9993145.

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

[28] M. Bianchi and S. Grammatico, "The END: Estimation Network Design for efficient distributed equilibrium seeking," *IEEE Transactions on Automatic Control*, under review. [Online]. Available: https://arxiv.org/abs/2208.11377.

[29] M. Bianchi, S. Grammatico, and J. Cortés, "Data-driven stabilization of switched and constrained linear systems," *IEEE Transactions on Automatic Control*, under review. [Online]. Available: https://arxiv.org/abs/2208.11392.

[30] K. Bimpikis, S. Ehsani, and R. Ilkiliç, "Cournot competition in networked markets," in *15th ACM Conference on Economics and Computation*, ACM, 2014, p. 733. DOI: 10.1145/2600057.2602882.

[31] E. Börgens and C. Kanzow, "A distributed regularized Jacobi-Type ADMM-Method for generalized Nash equilibrium problems in Hilbert spaces," *Numerical Functional Analysis and Optimization*, vol. 39, no. 12, pp. 1316–1349, 2018. DOI: 10.1080/01630563.2018.1477800.

[32] R. I. Boţ, E. R. Csetnek, and C. Hendrich, "Inertial Douglas–Rachford splitting for monotone inclusion problems," *Applied Mathematics and Computation*, vol. 256, pp. 472–487, 2015. DOI: 10.1016/j.amc.2015.01.017.

[33] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 1 2010. DOI: 10.1561/2200000016.

[34] L. M. Briceño-Arias and D. Davis, "Forward-backward-half forward algorithm for solving monotone inclusions," *SIAM Journal on Optimization*, vol. 28, no. 4, pp. 2839–2871, 2018. DOI: 10.1137/17M1120099.

[35] C. Cenedese, G. Belgioioso, S. Grammatico, and M. Cao, "An asynchronous distributed and scalable generalized Nash equilibrium seeking algorithm for strongly monotone games," *European Journal of Control*, vol. 58, pp. 143–151, 2021. DOI: https://doi.org/10.1016/j.ejcon.2020.08.006.

[36] C. Cenedese, G. Belgioioso, Y. Kawano, S. Grammatico, and M. Cao, "Asynchronous and time-varying proximal type dynamics in multiagent network games," *IEEE Transactions on Automatic Control*, vol. 66, no. 6, pp. 2861–2867, 2021. DOI: 10.1109/TAC.2020.3011916.

[37] M. Charikar, C. Chekuri, T.-y. Cheung, Z. Dai, A. Goel, S. Guha, and M. Li, "Approximation algorithms for directed Steiner problems," *Journal of Algorithms*, vol. 33, pp. 73–91, 1 1999. DOI: doi.org/10.1006/jagm.1999.1042.

[38] G. Chen, Y. Ming, Y. Hong, and P. Yi, "Distributed algorithm for ε-generalized Nash equilibria with uncertain coupled constraints," *Automatica*, vol. 123, p. 109 313, 2021. DOI: 10.1016/j.automatica.2020.109313.

[39] Z. Cheng, M. S. Pang, and P. A. Pavlou, "Mitigating traffic congestion: The role of intelligent transportation systems," *Information Systems Research*, vol. 31, pp. 653–674, 3 2020. DOI: 10.1287/ISRE.2019.0894.

[40] A. Cherukuri, E. Mallada, and J. Cortés, "Asymptotic convergence of constrained primal–dual dynamics," *Systems & Control Letters*, vol. 87, pp. 10–15, 2016. DOI: https://doi.org/10.1016/j.sysconle.2015.10.006.

[41] P. L. Combettes, "Quasi-fejérian analysis of some optimization algorithms," in *Inherently Parallel Algorithms in Feasibility and Optimization and their Applications*, ser. Studies in Computational Mathematics, D. Butnariu, Y. Censor, and S. Reich, Eds., vol. 8, Elsevier, 2001, pp. 115–152. DOI: 10.1016/S1570-579X(01)80010-0.

[42] A. A. Cournot, *Recherches sur les principes mathématiques de la théorie des richesses*, Cambridge University Press. 1838.

[43] E. Dall'Anese, S. V. Dhople, and G. B. Giannakis, "Regulation of dynamical systems to optimal solutions of semidefinite programs: Algorithms and applications to AC optimal power flow," in *2015 American Control Conference (ACC)*, 2015, pp. 2087–2092. DOI: 10.1109/ACC.2015.7171041.

[44] E. Dall'Anese, H. Zhu, and G. B. Giannakis, "Distributed optimal power flow for smart microgrids," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1464–1475, 3 2013. DOI: 10.1109/TSG.2013.2248175.

[45] P. Dasgupta and J. B. Collins, "A survey of game theoretic approaches for adversarial machine learning in cybersecurity tasks," *AI Magazine*, vol. 40, e1259, 2 2019. DOI: 10.1609/aimag.v40i2.2847.

[46] C. De Persis and S. Grammatico, "Continuous-time integral dynamics for a class of aggregative games with coupling constraints," *IEEE Transactions on Automatic Control*, vol. 65, no. 5, pp. 2171–2176, 2020. DOI: 10.1109/TAC.2019.2939639.

[47] C. De Persis and S. Grammatico, "Distributed averaging integral Nash equilibrium seeking on networks," *Automatica*, vol. 110, p. 108 548, 2019. DOI: https://10.1016/j.automatica.2019.108548.

[48] C. De Persis and N. Monshizadeh, "A feedback control algorithm to steer networks to a Cournot–Nash equilibrium," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 4, pp. 1486–1497, 2019. DOI: 10.1109/TCNS.2019.2897907.

[49] Z. Deng and S. Liang, "Distributed algorithms for aggregative games of multiple heterogeneous Euler–Lagrange systems," *Automatica*, vol. 99, pp. 246–252, 2019. DOI: https://doi.org/10.1016/j.automatica.2018.10.041.

[50] Z. Deng and X. Nian, "Distributed generalized Nash equilibrium seeking algorithm design for aggregative games over weight-balanced digraphs," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 695–706, 2019. DOI: 10.1109/TNNLS.2018.2850763.

[51] H.-B. Dürr, M. S. Stanković, and K. H. Johansson, "Distributed positioning of autonomous mobile sensors with application to coverage control," in *Proceedings of the 2011 American Control Conference*, 2011, pp. 4822–4827. DOI: 10.1109/ACC.2011.5991324.

[52] N. El Farouq, "Pseudomonotone Variational Inequalities: Convergence of Proximal Methods," *Journal of Optimization Theory and Applications*, vol. 109, no. 2, pp. 311–326, 2001.

[53] T. Erseghe, "A distributed and scalable processing method based upon ADMM," *IEEE Signal Processing Letters*, vol. 19, no. 9, pp. 563–566, 9 2012, ISSN: 10709908. DOI: 10.1109/LSP.2012.2207719.

[54] E. Eslami, H. Kebriaei, H. Atrianfar, and F. Farhadi, "Incentive mechanism design for unicast service provisioning with network aggregative game," *IEEE Transactions on Automatic Control*, p. 1, 2022. DOI: 10.1109/TAC.2022.3144132.

[55] F. Facchinei and C. Kanzow, "Generalized Nash equilibrium problems," *Annals of Operations Research*, vol. 175, pp. 177–211, 2010. DOI: 10.1007/s10479-009-0653-x.

[56] F. Facchinei and J. Pang, *Finite-dimensional variational inequalities and complementarity problems*. Springer New York, 2007. DOI: 10.1007/b97543.

[57] F. Facchinei and J. Pang, "Nash equilibria: The variational approach," in *Convex Optimization in Signal Processing and Communications*, D. P. Palomar and Y. C. Eldar, Eds., Cambridge University Press, 2009, pp. 443–493. DOI: 10.1017/CBO9780511804458.013.

[58] A. Falsone, I. Notarnicola, G. Notarstefano, and M. Prandini, "Tracking-ADMM for distributed constraint-coupled optimization," *Automatica*, vol. 117, p. 108 962, 2020. DOI: 10.1016/j.automatica.2020.108962.

[59] B. Franci and S. Grammatico, "Stochastic generalized Nash equilibrium seeking under partial-decision information," *Automatica*, vol. 137, p. 110 101, 2022. DOI: https://doi.org/10.1016/j.automatica.2021.110101.

[60] P. Frihauf, M. Krstic, and T. Basar, "Nash equilibrium seeking in noncooperative games," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1192–1207, 2012. DOI: 10.1109/TAC.2011.2173412.

[61] D. Fudenberg and J. Tirole, *Game Theory*. Cambridge, MA: MIT Press, 1991.

[62] D. Fullmer and A. S. Morse, "A distributed algorithm for computing a common fixed point of a finite family of paracontractions," *IEEE Transactions on Automatic Control*, vol. 63, no. 9, pp. 2833–2843, 9 2018, ISSN: 15582523. DOI: 10.1109/TAC.2018.2800644.

[63] D. Gadjov and L. Pavel, "A passivity-based approach to Nash equilibrium seeking over networks," *IEEE Transactions on Automatic Control*, vol. 64, no. 3, pp. 1077–1092, 2019. DOI: 10.1109/TAC.2018.2833140.

[64] D. Gadjov and L. Pavel, "Single-timescale distributed GNE seeking for aggregative games over networks via forward–backward operator splitting," *IEEE Transactions on Automatic Control*, vol. 66, no. 7, pp. 3259–3266, 2021. DOI: 10.1109/TAC.2020.3015354.

[65] D. Gadjov and L. Pavel, "Distributed Nash equilibrium seeking resilient to adversaries," in *2021 60th IEEE Conference on Decision and Control (CDC)*, 2021, pp. 191–196. DOI: 10.1109/CDC45484.2021.9683060.

[66] D. Gadjov and L. Pavel, "On the exact convergence to Nash equilibrium in hypomonotone regimes under full and partial-decision information," *IEEE Transactions on Automatic Control*, pp. 1–15, 2022. DOI: 10.1109/TAC.2022.3204543.

[67] D. Gadjov and L. Pavel, *Resilient Nash equilibrium seeking in the partial information setting*, 2021. [Online]. Available: https://arxiv.org/abs/2111.13735.

[68] J. Ghaderi and R. Srikant, "Opinion dynamics in social networks with stubborn agents: Equilibrium and convergence rate," *Automatica*, vol. 50, no. 12, pp. 3209–3215, 2014. DOI: 10.1016/j.automatica.2014.10.034.

[69] B. Gharesifard and J. Cortés, "Distributed strategies for generating weight-balanced and doubly stochastic digraphs," *European Journal of Control*, vol. 18, no. 6, pp. 539–557, 2012. DOI: 10.3166/EJC.18.539-557.

[70] C. Godsil and G. Royle, *Algebraic Graph Theory* (Graduate Texts in Mathematics.). Springer New York, 2013, vol. 207. DOI: 10.1007/978-1-4613-0163-9.

[71] S. Grammatico, "Dynamic control of agents playing aggregative games with coupling constraints," *IEEE Transactions on Automatic Control*, vol. 62, no. 9, pp. 4537–4548, 2017. DOI: 10.1109/TAC.2017.2672902.

[72] Z. Han, D. Niyato, W. Saad, and T. Başar, *Game Theory for Next Generation Wireless and Communication Networks: Modeling, Analysis, and Design*. Cambridge University Press, 2019. DOI: 10.1017/9781108277402.

[73] B. F. Hobbs and J. S. Pang, "Nash-Cournot equilibria in electric power markets with piecewise linear demand functions and joint constraints," *Operations Research*, vol. 55, no. 1, pp. 113–127, 2007. DOI: 10.1287/opre.1060.0342.

[74] Y. Huang and J. Hu, *Distributed computation of stochastic GNE with partial information: An augmented best-response approach*, 2021. [Online]. Available: https://arxiv.org/abs/2109.12290.

[75] Y. Huang and J. Hu, "Distributed solution of GNEP over networks via the douglas-rachford splitting method," in *2021 60th IEEE Conference on Decision and Control (CDC)*, 2021, pp. 3110–3116. DOI: 10.1109/CDC45484.2021.9682862.

[76] A. Isidori, *Nonlinear control systems: An introduction*. Springer Berlin, Heidelberg, 1987. DOI: 10.1007/BFb0006368.

[77] F. Iutzeler and J. M. Hendrickx, "A generic online acceleration scheme for optimization algorithms via relaxation and inertia," *Optimization Methods and Software*, vol. 34, no. 2, pp. 383–405, 2019. DOI: 10.1080/10556788.2017.1396601.

[78] N. Ivkin, D. Rothchild, E. Ullah, V. braverman, I. Stoica, and R. Arora, "Communication-efficient distributed SGD with sketching," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.

[79] H. K. Khalil, *Nonlinear Systems* (Pearson Education). Upper Saddle River, NJ: Prentice Hall, 2002.

[80] A. Kiayias, E. Koutsoupias, M. Kyropoulou, and Y. Tselekounis, "Blockchain mining games," in *Proceedings of the 2016 ACM Conference on Economics and Computation*, 2016, pp. 365–382. DOI: 10.1145/2940716.2940773.

[81] A. Koloskova, S. U. Stich, and M. Jaggi, "Decentralized stochastic optimization and gossip algorithms with compressed communication," in *36th International Conference on Machine Learning (ICML) 2019*, 2019, pp. 3478–3487. [Online]. Available: proceedings.mlr.press/v97/koloskova19a.html.

[82] J. Koshal, A. Nedić, and U. V. Shanbhag, "Distributed algorithms for aggregative games on graphs," *Operations Research*, vol. 64, no. 3, pp. 680–704, 2016. DOI: 10.1287/opre.2016.1501.

[83] J. Koshal, A. Nedić, and U. V. Shanbhag, "A gossip algorithm for aggregative games on graphs," in *51st IEEE Conference on Decision and Control (CDC)*, 2012, pp. 4840–4845. DOI: 10.1109/CDC.2012.6426136.

[84] S. Krilašević and S. Grammatico, "Learning generalized Nash equilibria in multi-agent dynamical systems via extremum seeking control," *Automatica*, vol. 133, p. 109 846, 2021. DOI: 10.1016/j.automatica.2021.109846.

[85] A. A. Kulkarni and U. V. Shanbhag, "On the variational equilibrium as a refinement of the generalized Nash equilibrium," *Automatica*, vol. 48, no. 1, pp. 45–55, 2012. DOI: 10.1016/j.automatica.2011.09.042.

[86] J. Lei and U. V. Shanbhag, "Linearly convergent variable sample-size schemes for stochastic Nash games: Best-response schemes and distributed gradient-response schemes," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 3547–3552. DOI: 10.1109/CDC.2018.8618953.

[87] J. Lei, U. V. Shanbhag, and J. Chen, "Distributed computation of Nash equilibria for monotone aggregative games via iterative regularization," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 2285–2290. DOI: 10.1109/CDC42340.2020.9303804.

[88] N. Li, L. Chen, and M. A. Dahleh, "Demand response using linear supply function bidding," *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 1827–1838, 2015. DOI: 10.1109/TSG.2015.2410131.

[89] X. Li, G. Feng, and L. Xie, "Distributed proximal algorithms for multiagent optimization with coupled inequality constraints," *IEEE Transactions on Automatic Control*, vol. 66, no. 3, pp. 1223–1230, 2021. DOI: 10.1109/TAC.2020.2989282.

[90] X. Li, L. Xie, and Y. Hong, "Distributed aggregative optimization over multi-agent networks," *IEEE Transactions on Automatic Control*, vol. 67, no. 6, pp. 3165–3171, 2022. DOI: 10.1109/TAC.2021.3095456.

[91] Z. Li, W. Shi, and M. Yan, "A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates," *IEEE Transactions on Signal Processing*, vol. 67, no. 17, pp. 4494–4506, 2019. DOI: 10.1109/TSP.2019.2926022.

[92] Z. Li, W. Shi, and M. Yan, "A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates," *IEEE Transactions on Signal Processing*, vol. 67, no. 17, pp. 4494–4506, 17. DOI: 10.1109/tsp.2019.2926022.

[93] W. Lin, C. Li, Z. Qu, and M. A. Simaan, "Distributed formation control with open-loop Nash strategy," *Automatica*, vol. 106, pp. 266–273, 2019. DOI: 10.1016/j.automatica.2019.04.034.

[94] P. D. Lorenzo and G. Scutari, "NEXT: In-network nonconvex optimization," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 2, pp. 120–136, DOI: 10.1109/TSIPN.2016.2524588.

[95] M. A. Lotufo, L. Colangelo, and C. Novara, "Control design for UAV quadrotors via embedded model control," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 5, pp. 1741–1756, 2020. DOI: 10.1109/TCST.2019.2918750.

[96] Y. Malitsky, "Golden ratio algorithms for variational inequalities," *Mathematical Programming*, vol. 184, pp. 383–410, 2020. DOI: 10.1007/s10107-019-01416-w.

[97] S. A. Morrison, "A survey of road pricing," *Transportation Research Part A: General*, vol. 20, no. 2, pp. 87–97, 1986. DOI: 10.1016/0191-2607(86)90035-X.

[98] J. F. Mota, J. M. Xavier, P. M. Aguiar, and M. Puschel, "Distributed optimization with local domains: Applications in MPC and network flows," *IEEE Transactions on Automatic Control*, vol. 60, no. 7, pp. 2004–2009, 7. DOI: 10.1109/TAC.2014.2365686.

[99] A. Moudafi, "About proximal-type methods for a class of nonmonotone operators," *Bulletin of the Iranian Mathematical Society*, vol. 46, no. 1, pp. 115–125, 2020. DOI: 10.1007/s41980-019-00244-0.

[100] J. Nash, "Non-cooperative games," *Annals of Mathematics*, vol. 54, no. 2, pp. 286–295, 1951. DOI: 10.2307/1969529.

[101] I. Necoara and D. Clipici, "Parallel random coordinate descent method for composite minimization: Convergence analysis and error bounds," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 197–226, 1 2016. DOI: 10.1137/130950288.

[102] A. Nedić and A. Olshevsky, "Distributed optimization over time-varying directed graphs," *IEEE Transactions on Automatic Control*, vol. 60, no. 3, pp. 601–615, 3 2015. DOI: 10.1109/TAC.2014.2364096.

[103] A. Nedić, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM Journal on Optimization*, vol. 27, no. 4, pp. 2597–2633, 2017. DOI: 10.1137/16M1084316.

[104] I. Notarnicola, R. Carli, and G. Notarstefano, "Distributed partitioned big-data optimization via asynchronous dual decomposition," *IEEE Transactions on Control of Network Systems*, vol. 5, pp. 1910–1919, 4 2018. DOI: 10.1109/TCNS.2017.2774010.

[105] D. Paccagnan, B. Gentile, F. Parise, M. Kamgarpour, and J. Lygeros, "Distributed computation of generalized nash equilibria in quadratic aggregative games with affine coupling constraints," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, 2016, pp. 6123–6128. DOI: 10.1109/CDC.2016.7799210.

[106] Y. Pang and G. Hu, "Distributed Nash equilibrium seeking with limited cost function knowledge via a consensus-based gradient-free method," *IEEE Transactions on Automatic Control*, vol. 66, no. 4, pp. 1832–1839, 2021. DOI: 10.1109/tac.2020.2995666.

[107] J. C. Parinya and Fakcharoenphol, "Simple distributed algorithms for approximating minimum Steiner trees," L. Wang, Ed., Springer Berlin Heidelberg, 2005, pp. 380–389, ISBN: 978-3-540-31806-4. DOI: 10.1007/11533719_39.

[108] F. Parise, B. Gentile, and J. Lygeros, "A distributed algorithm for almost-Nash equilibria of average aggregative games with coupling constraints," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 2, pp. 770–782, 2020. DOI: 10.1109/TCNS.2019.2944300.

[109] L. Pavel, "Distributed GNE seeking under partial-decision information over networks via a doubly-augmented operator splitting approach," *IEEE Transactions on Automatic Control*, vol. 65, no. 4, pp. 1584–1597, 4 2020. DOI: 10.1109/TAC.2019.2922953.

[110] L. Pavel, *Game Theory for Control of Optical Networks*. Birkhäuser, 2012. DOI: 10.1007/978-0-8176-8322-1.

[111] J. I. Poveda, "Decentralized feedback equilibrium seeking in multi-agent cyber-physical systems," in *2022 2nd International Workshop on Computation-Aware Algorithmic Design for Cyber-Physical Systems (CAADCPS)*, 2022, pp. 13–14. DOI: 10.1109/CAADCPS56132.2022.00008.

[112] G. Qu and N. Li, "Harnessing smoothness to accelerate distributed optimization," *IEEE Transactions on Control of Network Systems*, vol. 5, pp. 159–166, 3 2018. DOI: 10.1109/TCNS.2017.2698261.

[113] R. J. B. W. R. Tyrrell Rockafellar, *Variational Analysis*. Springer Berlin, Heidelberg, 1998. DOI: doi.org/10.1007/978-3-642-02431-3.

[114] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Third International Symposium on Information Processing in Sensor Networks (IPSN)*, 2004, pp. 20–27. DOI: 10.1145/984622.984626.

[115] R. T. Rockafellar, "Monotone operators associated with saddle-functions and minimax problems," *Nonlinear functional analysis*, vol. 18, no. part 1, pp. 397–407, 1970. [Online]. Available: https://sites.math.washington.edu/~rtr/papers/rtr028-MonoSaddle.pdf.

[116] A. R. Romano and L. Pavel, "Dynamic NE seeking for multi-integrator networked agents with disturbance rejection," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 1, pp. 129–139, 2020. DOI: 10.1109/TCNS.2019.2920590.

[117] A. R. Romano and L. Pavel, "Exact convergence to GNE using penalty methods," in *2022 European Control Conference (ECC)*, 2022, pp. 2285–2290. DOI: 10.23919/ECC55457.2022.9838103.

[118] A. R. Romano and L. Pavel, "GNE seeking in games with passive dynamic agents via inexact-penalty methods," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 500–505. DOI: 10.1109/CDC42340.2020.9304499.

[119] W. Saad, Z. Han, H. V. Poor, and T. Basar, "Game-theoretic methods for the smart grid: An overview of microgrid systems, demand-side management, and smart grid communications," *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 86–105, 2012. DOI: 10.1109/MSP.2012.2186410.

[120] F. Salehisadaghiani, W. Shi, and L. Pavel, "Distributed Nash equilibrium seeking under partial-decision information via the alternating direction method of multipliers," *Automatica*, vol. 103, pp. 27–35, 2019. DOI: 10.1016/j.automatica.2019.01.012.

[121] F. Salehisadaghiani and L. Pavel, "Distributed Nash equilibrium seeking in networked graphical games," *Automatica*, vol. 87, 2018. DOI: 10.1016/j.automatica.2017.09.016.

[122] F. Salehisadaghiani and L. Pavel, "Distributed Nash equilibrium seeking: A gossip-based algorithm," *Automatica*, vol. 72, pp. 17–24, 2016. DOI: 10.1016/j.automatica.2016.06.004.

[123] F. Salehisadaghiani and L. Pavel, "Nash equilibrium seeking with non-doubly stochastic communication weight matrix," in *Game Theory for Networks*, 2017, pp. 3–15. DOI: 10.1007/978-3-319-67540-4_1.

[124] G. Scutari, F. Facchinei, J. Pang, and D. P. Palomar, "Real and complex monotone communication games," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 4197–4231, 2014. DOI: 10.1109/TIT.2014.2317791.

[125] G. Scutari and Y. Sun, "Distributed nonconvex constrained optimization over time-varying digraphs," *Mathematical Programming*, vol. 176, pp. 497–544, 2019. DOI: 10.1007/s10107-018-01357-w.

[126] J. S. Shamma and G. Arslan, "Dynamic fictitious play, dynamic gradient play, and distributed convergence to Nash equilibria," *IEEE Transactions on Automatic Control*, vol. 50, no. 3, pp. 312–327, 2005. DOI: 10.1109/TAC.2005.843878.

[127] W. Shi, Q. Ling, G. Wu, and W. Yin, "EXTRA: An exact first-order algorithm for decentralized consensus optimization," *SIAM Journal on Optimization*, vol. 25, no. 2, pp. 944–966, 2015. DOI: 10.1137/14096668X.

[128] J. W. Simpson-Porco, "Equilibrium-independent dissipativity with quadratic supply rates," *IEEE Transactions on Automatic Control*, vol. 64, no. 4, pp. 1440–1455, 2019. DOI: 10.1109/TAC.2018.2838664.

[129] T. Sousa, T. Soares, P. Pinson, F. Moret, T. Baroche, and E. Sorin, "Peer-to-peer and community-based markets: A comprehensive review," *Renewable and Sustainable Energy Reviews*, vol. 104, pp. 367–378, 2019. DOI: 10.1016/j.rser.2019.01.036.

[130] M. S. Stankovic, K. H. Johansson, and D. M. Stipanovic, "Distributed seeking of Nash equilibria with applications to mobile sensor networks," *IEEE Transactions on Automatic Control*, vol. 57, no. 4, pp. 904–919, 2012. DOI: 10.1109/TAC.2011.2174678.

[131] A. Stevenson, Ed., *Oxford Dictionary of English*. Oxford University Press, 2010. DOI: 10.1093/acref/9780199571123.001.0001.

[132] T. M. Stoenescu and D. Teneketzis, *Decentralized resource allocation mechanisms in networks: Realization and implementation*, 2005. DOI: 10.1007/0-8176-4409-1_13.

[133] B. Swenson, S. Kar, and J. Xavier, "Empirical centroid fictitious play: An approach for distributed learning in multi-agent games," *IEEE Transactions on Signal Processing*, vol. 63, no. 15, pp. 3888–3901, 2015. DOI: 10.1109/TSP.2015.2434327.

[134] W. Szeto and H. K. Lo, "Transportation network improvement and tolling strategies: The issue of intergeneration equity," *Transportation Research Part A: Policy and Practice*, vol. 40, no. 3, pp. 227–243, 2006. DOI: 10.1016/j.tra.2005.06.004.

[135] T. Tatarenko and A. Nedić, "Geometric convergence of distributed gradient play in games with unconstrained action sets," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 3367–3372, 2020, 21st IFAC World Congress. DOI: 10.1016/j.ifacol.2020.12.1501.

[136] T. Tatarenko, W. Shi, and A. Nedic, "Geometric convergence of gradient play algorithms for distributed Nash equilibrium seeking," *IEEE Transactions on Automatic Control*, vol. 66, no. 11, pp. 5342–5353, 11 2021. DOI: 10.1109/TAC.2020.3046232.

[137] T. Tatarenko, W. Shi, and A. Nedić, "Accelerated gradient play algorithm for distributed Nash equilibrium seeking," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 3561–3566. DOI: 10.1109/CDC.2018.8619479.

[138] M. Todescato, N. Bof, G. Cavraro, R. Carli, and L. Schenato, "Partition-based multi-agent optimization in the presence of lossy and asynchronous communication," *Automatica*, vol. 111, p. 108 648, 2020. DOI: 10.1016/j.automatica.2019.108648.

[139] W. Tushar, C. Yuen, H. Mohsenian-Rad, T. Saha, H. V. Poor, and K. L. Wood, "Transforming energy networks via peer-to-peer energy trading: The potential of game-theoretic approaches," *IEEE Signal Processing Magazine*, vol. 35, no. 4, pp. 90–111, 2018. DOI: 10.1109/MSP.2018.2818327.

[140] C. A. Uribe, S. Lee, A. Gasnikov, and A. Nedić, "A dual approach for optimal algorithms in distributed optimization over networks," *Optimization Methods and Software*, vol. 36, pp. 1–37, 1 2021. DOI: 10.1080/10556788.2020.1750013.

[141] B. Wang, Y. Wu, and K. R. Liu, "Game theory for cognitive radio networks: An overview," *Computer Networks*, vol. 54, no. 14, pp. 2537–2561, 2010. DOI: 10.1016/j.comnet.2010.04.004.

[142] M. Wang, S. P. Hoogendoorn, W. Daamen, B. van Arem, and R. Happee, "Game theoretic approach for predictive lane-changing and car-following control," *Transportation Research Part C: Emerging Technologies*, vol. 58, pp. 73–92, 2015. DOI: 10.1016/j.trc.2015.07.009.

[143] C. Xi, V. S. Mai, R. Xin, E. H. Abed, and U. A. Khan, "Linear convergence in optimization over directed graphs with row-stochastic matrices," *IEEE Transactions on Automatic Control*, vol. 63, no. 10, pp. 3558–3565, 2018. DOI: 10.1109/TAC.2018.2797164.

[144] J. Xu, Y. Tian, Y. Sun, and G. Scutari, "Distributed algorithms for composite optimization: Unified framework and convergence analysis," *IEEE Transactions on Signal Processing*, vol. 69, pp. 3555–3570, 2021. DOI: 10.1109/TSP.2021.3086579.

[145] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, "Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes," in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015, pp. 2055–2060. DOI: 10.1109/CDC.2015.7402509.

[146] M. Ye and G. Hu, "Distributed Nash equilibrium seeking by a consensus based approach," *IEEE Transactions on Automatic Control*, vol. 62, no. 9, pp. 4811–4818, 9 2017. DOI: 10.1109/TAC.2017.2688452.

[147] M. Ye and G. Hu, "Game design and analysis for price-based demand response: An aggregate game approach," *IEEE Transactions on Cybernetics*, vol. 47, no. 3, pp. 720–730, 2017. DOI: 10.1109/TCYB.2016.2524452.

[148] P. Yi and L. Pavel, "Distributed generalized Nash equilibria computation of monotone games via double-layer preconditioned proximal-point algorithms," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 1, pp. 299–311, 2019. DOI: 10.1109/TCNS.2018.2813928.

[149] P. Yi and L. Pavel, "An operator splitting approach for distributed generalized Nash equilibria computation," *Automatica*, vol. 102, pp. 111–121, 2019. DOI: https://10.1016/j.automatica.2019.01.008.

[150] P. Yi and L. Pavel, "Asynchronous distributed algorithms for seeking generalized Nash equilibria under full and partial-decision information," *IEEE Transactions on Cybernetics*, vol. 50, no. 6, pp. 2514–2526, 2020. DOI: 10.1109/TCYB.2019.2908091.

[151] C. Yu, M. van der Schaar, and A. H. Sayed, "Distributed learning for stochastic generalized Nash equilibrium problems," *IEEE Transactions on Signal Processing*, vol. 65, no. 15, pp. 3893–3908, 2017. DOI: 10.1109/TSP.2017.2695451.

[152] M. Zargham, A. Ribeiro, and A. Jadbabaie, "A distributed line search for network optimization," in *2012 American Control Conference (ACC)*, 2012, pp. 472–477. DOI: 10.1109/ACC.2012.6314986.

[153] X. Zhang, A. Papachristodoulou, and N. Li, "Distributed optimal steady-state control using reverse- and forward-engineering," in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015, pp. 5257–5264. DOI: 10.1109/CDC.2015.7403042.

[154] Y. Zhang, S. Liang, X. Wang, and H. Ji, "Distributed Nash equilibrium seeking for aggregative games with nonlinear dynamics under external disturbances," *IEEE Transactions on Cybernetics*, vol. 50, no. 12, pp. 4876–4885, 2020. DOI: 10.1109/TCYB.2019.2929394.

ACRONYMS

ADMM	*alternating direction method of multipliers*
DGD	*distributed gradient descent*
END	*Estimation Network Design*
FB	*forward-backward*
GNE	*generalized Nash equilibrium*
GNEP	*GNE problem*
ISO	*independent system operator*
ISS	*input-to-state-stable*
KKT	*Karush–Kuhn–Tucker*
NE	*Nash equilibrium*
NEP	*Nash equilibrium problem*
PF	*Perron-Frobenius*
PPA	*proximal-point algorithm*
PPP	*preconditioned proximal-point*
PPPA	*preconditioned PPA*
v-GNE	*variational GNE*

List of Symbols

Basic Relations

$:=$	equal to by definition
$\mid$	such that
$\in$	belongs to
$\exists$	there exists
$\forall$	for all
$\Rightarrow$	implies
$\Leftrightarrow$	if and only if
$\rightarrow$	maps to an element
$\rightrightarrows$	maps to a set

Set, Spaces and Set Operators

$\mathbb{N}$	set of natural numbers (including 0)
$\mathbb{R}$	set of real numbers
$\overline{\mathbb{R}}$	: $\mathbb{R} \cup \{\infty\}$, set of extended real numbers
$\mathbb{R}_{\geq 0}$	set of nonnegative real numbers
$\mathbb{R}_{>0}$	set of positive real numbers
$\mathbb{R}^n$	set of real n-dimensional vectors
$\mathbb{R}^{n \times m}$	set of real $n \times m$ -dimensional matrices
ℓ^1	set of absolutely summable sequences
$[a, b]$	closed interval of real numbers
(a, b)	open interval of real numbers
$A \cup B$	union of the sets A and B
$A \cap B$	intersection of the sets A and B
$A \subset B$	A is a strict subset of B
$A \subseteq B$	A is a subset of (or equals to) B
$A \backslash B$	set of elements that are in A but not in B
$A + B$	Minkowski sum of the sets A and B
$A \times B$	Cartesian product of the sets A and B
$\prod_{i=1}^{N} A_i$	$:= A_1 \times A_2 \times \ldots \times A_N$
$\bigcup_{i=1}^{N} A_i$	$:= A_1 \cup A_2 \cup \ldots A_N$

Operations on Vectors and Matrices

$\mathbf{0}$	matrix/vector with all elements equal to 0
$\mathbf{1}$	matrix/vector with all elements equal to 1
$\mathbf{e}_i$	vector with i-th element equal 1, all others are zero
$v^\top$	transpose of the vector/matrix v

$\mathrm{col}(v_1,\dots,v_N)$	$:= \begin{bmatrix} v_1^\top & v_2^\top & \dots & v_N^\top \end{bmatrix}^\top$
$\mathrm{diag}(A_1,\dots,A_N)$	block-diagonal matrix with matrices $A_1,\dots,A_N$ on the diagonal
$(v_k)_{k\in\mathbb{N}}$	sequence of vectors v_k
M^{-1}	inverse of the matrix M
$[M]_{i,j}$	element in row i and column j of the matrix M
$M_1 \otimes M_2$	Kronecker productof the matrices M_1 and M_2
$\lambda_i(M)$	i-th smallest eigenvalue of M (counted with multiplicity)
$\lambda_{\max}(M)$	bigger eigenvalue of M
$\lambda_{\min}(M)$	smallest eigenvalue of M
$\sigma_i(M)$	i-th smallest singular value of M (counted with multiplicity)
$\lambda_{\max}(M)$	bigger singular value of M
$\lambda_{\min}(M)$	smallest singular value of M
$M \succ 0$	M is positive definite
$M \succcurlyeq 0$	M is positive semidefinite
$\mathrm{null}(M)$	kernel of the matrix M
$\mathrm{range}(M)$	range of the matrix M

NORMS AND EUCLIDEAN SPACES

$\|v\|$	$:= \sqrt{v^\top v}$, 2-norm of the vector V
$\|M\|$	$:= \sigma_{\max}(M)$, 2-matrix norm
$\|\cdot\|$	infinity matrix/vector norm
$\langle \cdot \mid \cdot \rangle$	inner product, i.e., $\langle x \mid y \rangle = x^\top y$
$\langle \cdot \mid \cdot \rangle_M$	Φ-weighted inner product, i.e., $\langle x \mid y \rangle_\Phi = x^\top \Phi y$
$\|v\|_\Phi$	Φ weighted vector norm, i.e., $\|v\|_\Phi^2 = \langle v \mid v \rangle_\Phi$
$\|M\|_\Phi$	Φ-induced matrix norm
$\mathcal{H}_\Phi$	Euclidean space with inner product $\langle \cdot \mid \cdot \rangle_M$

OPERATOR THEORY

Id	identity operator
proj_Ω	Euclidean projection onto the set Ω
$\mathrm{dom}(F)$	domain of the operator F
$\mathrm{fix}(F)$	fixed-point set of the operator F
$\mathrm{range}(F)$	range of the operator F
$\mathrm{zer}(F)$	zero set of the operator F
F^{-1}	inverse operator of F
∂f	subdifferential of the function f
∇f	gradient of the differentiable function f
$\partial_x f$	subdifferential of the function f w.r.t. argument x
$\nabla_x f$	gradient of the differentiable function f w.r.t. argument x
J_F	$:= (\mathrm{Id}+F)^{-1}$ resolvent of the operator F
ι_Ω	indicator function of the set Ω
N_Ω	normal cone operator of the set Ω
proj_Ω	projection onto the set Ω

GRAPH THEORY

$\mathcal{G} = (\mathcal{I}, \mathcal{E})$	a graph (directed by default)
$\mathcal{N}_i$	in-neighbor set of agent i [Chapters 2-6]
$\overline{\mathcal{N}}_i$	out-neighbor set of agent i [Chapters 2-6]
$\mathcal{N}(i)$	in-neighbor set of agent i [Chapter 7]
$\overline{\mathcal{N}}(i)$	out-neighbor set of agent i [Chapter 7]
$\mathcal{E}$	set of edges
W	weight matrix
L	Laplacian matrix
D	in-degree matrix
V	weighted incidence matrix

GAME THEORY

N	number of agents
$\mathcal{I}$	$:= \{1, 2, \dots, N\}$, set of indeces of the agents
x_i	decision variable of agent i
x_{-i}	$:= \text{col}\,(x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_N)$
x	$:= \text{col}\,(x_1, x_2, \dots, x_N)$
$J_i(x_i, x_{-i})$	cost function of agent i
Ω_i	local feasible set of agent i
$F(x)$	$:= \text{col}\big(\nabla_{x_1} J_1(x_1, x_{-1}), \dots, J_N(x_1, x_{-N})\big)$, pseudo-gradient

PARTIAL-DECISION INFORMATION SCENARIO

Chapter 2-6

$\boldsymbol{x}_{i,j}$	estimate kept by agent i of x_j
$\boldsymbol{x}_i$	$:= \text{col}(\boldsymbol{x}_{i,1}, \dots, \boldsymbol{x}_{i,N})$, vector of estimates kept by agent i
$\boldsymbol{x}$	$:= \text{col}(\boldsymbol{x}_i, \dots, \boldsymbol{x}_N)$

Chapter 7

$\boldsymbol{y}_{i,p}$	estimate kept by agent i of y_p
$\boldsymbol{y}_p$	vector of estimates of y_p kept by different agents
$\tilde{\boldsymbol{y}}_i$	vector of estimates kept by agent i
$\boldsymbol{y}$	$:= \text{col}(\boldsymbol{y}_1, \dots, \boldsymbol{y}_P)$
$\tilde{\boldsymbol{y}}$	$:= \text{col}(\tilde{\boldsymbol{y}}_i, \dots, \tilde{\boldsymbol{y}}_N)$

List of Publications

Journal articles and preprints

6. M. Bianchi and S. Grammatico, "The END: Estimation Network Design for efficient distributed equilibrium seeking," *IEEE Transactions on Automatic Control*, under review. [Online]. Available: https://arxiv.org/abs/2208.11377

5. M. Bianchi, S. Grammatico, and J. Cortés, "Data-driven stabilization of switched and constrained linear systems," *IEEE Transactions on Automatic Control*, under review. [Online]. Available: https://arxiv.org/abs/2208.11392

4. M. Bianchi and S. Grammatico, "Continuous-time fully distributed generalized Nash equilibrium seeking for multi-integrator agents," *Automatica*, vol. 129, p. 109 660, 2021. DOI: 10.1016/j.automatica.2021.109660

3. M. Bianchi, G. Belgioioso, and S. Grammatico, "Fast generalized Nash equilibrium seeking under partial-decision information," *Automatica*, vol. 136, p. 110 080, 2022. DOI: 10.1016/j.automatica.2021.110080

2. M. Bianchi, W. Ananduta, and S. Grammatico, "The distributed dual ascent algorithm is robust to asynchrony," *IEEE Control Systems Letters*, vol. 6, pp. 650–655, 2022. DOI: 10.1109/LCSYS.2021.3084883

1. M. Bianchi and S. Grammatico, "Fully distributed Nash equilibrium seeking over time-varying communication networks with linear convergence rate," *IEEE Control Systems Letters*, vol. 5, pp. 499–504, 2021. DOI: 10.1109/LCSYS.2020.3002734

Conference papers

5. M. Bianchi and S. Grammatico, "Nash equilibrium seeking under partial decision information: Monotonicity, smoothness and proximal-point algorithms," in *2022 61st IEEE Conference on Decision and Control (CDC)*, 2022, pp. 5080–5085. DOI: 10.1109/CDC51059.2022.9993145

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

3. M. Bianchi, G. Belgioioso, and S. Grammatico, "A fully-distributed proximal-point algorithm for Nash equilibrium seeking with linear convergence rate," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 2303–2308. DOI: 10.1109/CDC42340.2020.9304145

2. M. Bianchi and S. Grammatico, "A continuous-time distributed generalized Nash equilibrium seeking algorithm over networks for double-integrator agents," in *2020 European Control Conference (ECC)*, 2020, pp. 1474–1479. DOI: 10.23919/ECC51009.2020.9143714

1. M. Bianchi, A. van der Maas, E. Maljaars, and W. Heemels, "Offset-free MPC for resource sharing on a nonlinear SCARA robot," in *6th IFAC Conference on Nonlinear Model Predictive Control NMPC 2018*, vol. 51, 2018, pp. 265–272. DOI: 10.1016/j.ifacol.2018.11.024

ABOUT THE AUTHOR

Mattia Bianchi was born in Guardiagrele, Italy, in 1994. He received the Bachelor's degree in Information and Communication Engineering (with special distinction) in July 2016, and the Master's degree in Systems Engineering (with special distinction) in July 2018, both from University of L'Aquila, Italy. From February to June 2018 he visited the Mechanical Engineering department, TU Eindhoven, The Netherlands.

Since November 2018 he has been a PhD candidate in the Delft Center for Systems and Control, TU Delft, The Netherlands, under the supervision of Prof. Sergio Grammatico. From September 2021 to March 2022 he visited the Mechanical and Aerospace Engineering Department, University of California San Diego, United States, where he was advised by Prof. Jorge Cortés.

His research interests include game theory, operator theory and data-driven control, to solve decision and control problems for complex systems of systems, in the presence of network coupling and uncertainty. He is a recipient of the 2021 Roberto Tempo Best Paper Award at the IEEE Conference on Decision and Control.

ACKNOWLEDGEMENTS

I started a PhD because I wanted to travel and see the world. Then a pandemic happened, and I was left with research; all and all, it turned out not too bad. At the end of this bumpy but fun ride, I want to express my sincerest gratitude to a number of outstanding people that accompanied me, taught me, and worked with me.

First and foremost, this journey would not have been possible without the endless support of my promotor, dr. Sergio Grammatico. He continuously challenged me to strive for the highest research quality; he taught me how to achieve meaningful mathematical results, first, and how to work my way into academia, then. I always enjoyed our technical discussions and brainstormings. I will never thank him enough for his patience, for the freedom he granted, for never sparing criticism – except then to take my part when things went wrong.

My gratitude also goes to my copromotor, Prof. Bart De Schutter, for providing timely and insightful feedback, whenever I asked for it. His work efficiency never ceases to amaze me and sets an example for all doctoral candidates of DCSC.

A special recognition to Prof. Jorge Cortés, for hosting me during my visit to UCSD. His passion for research, open-mindedness and hands-on supervision are truly inspiring. I have learned a lot from his technical skills during my time at UCSD.

I extend my appreciation to my PhD committee members, Prof. Tamas Keviczky, Prof. Geert Leus, Prof. Claudio De Persis, Prof. Lacra Pavel, Prof. Jorge Cortés, for their valuable feedback and the time invested into judging this dissertation.

My warmest thanks also to all the other brilliant researchers I had the honor to collaborate with throughout these years, that shared their brilliant ideas and insights with me, without whom many of my articles would not exist. Among them, I would like to acknowledge my friends and co-authors Giuseppe Belgioioso and Wicak Ananduta. And my students, Rens Vermeer and Lorenzo Flipse: I surely learned more from them than they did from me.

I would like to thank the amazing staff at TU Delft: Erica, Marieke, Heleen, Francy, for making DCSC a better place and for all the social events; Ms. Charlier, who helped me find a date for the defence; Naima, for her contagious happiness.

In the last four years, I was fortunate enough to meet a number of wonderful friends. Some special thanks are in (random) order. To Filippo, Alfiya, Giuseppe, Carlo, Aitazaz, Barbara, Suad, Wicak, Emilio, companions of infinite coffee breaks, beers, BBQs, who became much more than just colleagues. To the “Latinos” family: Abhimanyu, Tomas, Prof. Momo, Dr. Jesus, Jesus, Dr. Carlos, Alejandro. To the missing members of the “Chili con carne” community: Dr. Mahdiyeh, Rodrigo, Giovanni, Alvaro, Carlos, Carlos, Carlos. To Abhimanyu, the first person I met and my dearest friend in Delft, then my office mate, tennis partner, real estate agent (no worries, I’ll bring you a copy of the thesis at your wedding). To Barbara, for all the dinner invitations and the best homemade pizza in Delft. To Tomas, for hosting me in his house and for admirably carrying out

his role of dictator until his deposition. To Prof. Momo and Dr. Mahdiyeh, for all the BBQs and game nights; in particular, to Prof. Momo for patiently respecting the first rule of "Bang!". To Carlo and Giuseppe, for welcoming me to Zurich and for their precious help in the last months. To Filippo, to whom we all owe the "Fabiani coffee" recipe. To Alfiya, for being our guide in KAUST. To Michele, Wouter, Elisa, for all the board game nights; a particular note to Wouter for his incredible cooking skills and for feeding Marina in my absence. To Sid and Nitish, the best housemates, for the countless nights at home drinking beers, telling stories, and solving the world problems, and for the Indian recipes and celebrations. To all the futsal friends, Petros, Alejandro, Rafa, Samir, Daan, Ahmed: we could have been a good team, but we like to party too much. To Niklas, for never letting me win a single tennis set in four years. To the friends I met in San Diego: Josh, for keeping me company in the first weeks and for the football games; Jaap, Sander (the first two researchers I encountered when I arrived in California from Delft are, incredibly, Dutch), Nicklas, Tommaso, Pol, Henning, Brandon, Scott, for all the amazing hikes, dinners, philosophical discussions. To Alessandro, who shared with me the joys and sorrows of a PhD, in spite of the distance. To Mario, for the hospitality and all the BBQs.

Thank you also to every longtime friend from Abruzzo (the most beautiful region in Italy, you should come, one of the few places where you can ski looking at the sea) and from elsewhere, that were not directly involved in my PhD adventure, but that were always there for me: you have a special place in my heart, "state pensati".

A special note of gratitude to Marina, for her love. Thank you for all the beautiful moments shared in these years; thank you for your patience, I hope you have some left; thank you for being my fire during the coldest nights. Thank you for always wanting the best for me, and for continuously trying to convince me to find a normal job (although, until now, unsuccessfully).

I am profoundly grateful and I owe everything to my family, for their incessant support. To my brother: I am proud of you and I look up to you. To my mother, for her unconditional trust and for teaching me what is really important. To Nonna Pina and Zio Vittorio, for the 24/7 cooking helpline and the constant encouragement. To Zia Gabriella, Zio Armando, Angela, Giovanni, Andrea, Cecilia, Aurora, who invariably rooted for me. To my father, for trying.

Finally, I would like to dedicate this thesis to my grandfather Vincenzo, who always encouraged my curiosity and supported my studies.

Mattia Bianchi
Delft, January 2023