

UC San Diego

UC San Diego Electronic Theses and Dissertations

Title

Distributionally Robust Nash Equilibrium Seeking With Applications to Task Allocation Among Self-Interested Agents

Permalink

<https://escholarship.org/uc/item/934643sx>

ISBN

9798193290124

Author

Mandal, Nirabhra

Publication Date

2026-10-12

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA SAN DIEGO

Distributionally Robust Nash Equilibrium Seeking
With Applications to Task Allocation Among Self-Interested Agents

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy

in

Engineering Sciences (Mechanical Engineering)

by

Nirabhra Mandal

Committee in charge:

Professor Sonia Martínez, Chair
Professor Nikolay Atanasov
Professor Miroslav Krstic
Professor Jorge I. Poveda

Copyright

Nirabhra Mandal, 2026

All rights reserved.

The Dissertation of Nirabhra Mandal is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2026

DEDICATION

TO

My Parents

Dr. Ratna Mondal

and

Dr. Gautam Mondal

whose support is like a flickering candle in the all-consuming void of oblivion.

And to my late grandmother *Chhaya Rani Halder*, whose gentle and loving presence is beacon of light in my fondest memories; and whose absence is a shadow in my daily life.

EPIGRAPH

“... all men live not by care for themselves but by love (of others) ...”

- Leo Tolstoy, What men live by

TABLE OF CONTENTS

Dissertation Approval Page	iii
Dedication	iv
Epigraph.....	v
Table of Contents	vi
List of Figures	ix
List of Tables.....	xi
List of Algorithms.....	xii
Acknowledgements	xiii
Vita.....	xv
Abstract of the Dissertation	xvi
Chapter 1 Introduction	1
1.1 Motivation.....	1
1.2 Organization and Contributions	2
Chapter 2 Preliminaries.....	6
2.1 Notation	6
2.2 Measure theory and concentration results	8
2.3 Size functions and input-to-state stability	9
2.4 Convex analysis	11
2.5 Graph theory	12
Part I Distributionally Robust Games.....	13
Chapter 3 Distributionally Robust Games And An Algorithm To Seek Their Nash Equilibria	14
3.1 Introduction	14
3.2 Literature Review	15
3.3 Contributions	17
3.4 Problem Formulation	19
3.5 On the Nash Equilibria of the Games	24
3.6 A Centralized Better Response Supergradient Ascent Dynamics.....	32
3.7 Distributed DRoNE Seeking	47
3.8 When Samples are Shared	53
3.9 Simulations	59

3.9.1	Effect of step-size α_i :	59
3.9.2	Performance of d-ISBRAG:	62
3.10	Summary	62
Part II	Application of Proposed Framework	64
Chapter 4	Task Allocation Among Self-Interested Agents	65
4.1	Introduction	65
4.2	Literature review	65
4.3	Contributions	67
4.4	Preliminaries	68
4.5	Problem Formulation	69
4.6	On Nash Equilibria and Optimal Partitions	73
4.7	Best Response Projected Gradient Ascent	80
4.8	Distributed Task Allocation	83
4.9	Simulations	89
4.9.1	Fast convergence of PBRAG with large step-size	89
4.9.2	Effect of constant step-size on d-PBRAG	92
4.9.3	d-PBRAG with time-varying step-sizes	92
4.9.4	Comparison with Hungarian algorithm with large number of agents	94
4.10	Summary	99
Chapter 5	Behaviorally Heterogeneous Multi-Agent Exploration	100
5.1	Introduction	100
5.2	Literature Review	102
5.3	Contributions	103
5.4	Preliminaries	103
5.5	Problem Formulation	105
5.5.1	Exploration environment	105
5.5.2	Robot Team	106
5.5.3	Frontier assignment problem	106
5.5.4	Communication network	108
5.5.5	Problem Statement	108
5.6	Distributed Frontier Assignment	108
5.7	Heterogeneous Behavioral Exploration	115
5.8	Simulations and Results	121
5.8.1	Environment Setup	121
5.8.2	Exploration pipeline	124
5.8.3	Results and Discussion	128
5.9	Summary	132
Chapter 6	Robust Control Of Discrete-Time LTI Systems Using Stochastic Ensemble Systems	133
6.1	Introduction	133

6.2	Literature Review	133
6.3	Contributions	134
6.4	Problem Formulation	135
6.5	Control using Stochastic Ensemble Systems	139
6.6	On Stochastic Ensemble Systems Generated through Sampling.....	143
6.6.1	Sample Averaging using Least Squares Error Minimization	146
6.6.2	Comparison between the Averaging Methods	149
6.7	Simulations and Analysis	150
6.7.1	Error in Computed Control	150
6.7.2	Error in Computed State	152
6.8	Summary	153
Chapter 7	Conclusion and Future Work	155
	Bibliography	158

LIST OF FIGURES

Figure 3.1.	Effect of step-size parameters $\{\alpha_i\}_{i \in \mathcal{A}}$ on convergence of ISBRAG. The plots share a common legend. (Top) $\alpha_i = 0.1, \forall i \in \mathcal{A}$. (Bottom) $\alpha_i = 0.01, \forall i \in \mathcal{A}$	60
Figure 3.2.	Convergence properties of d-ISBRAG. The plots share a common legend. Different plots correspond to different values of communication time steps T_{con} and step-size parameters $\{\alpha_i\}_{i \in \mathcal{A}}$. The figures from the top are arranged as follows. (First) $T_{\text{con}} = 10, \alpha_i = 0.01, \forall i \in \mathcal{A}$. (Second) $T_{\text{con}} = 50, \alpha_i = 0.01, \forall i \in \mathcal{A}$. (Third) $T_{\text{con}} = 100, \alpha_i = 0.01, \forall i \in \mathcal{A}$. (Fourth) $T_{\text{con}} = 100, \alpha_i = 0.1, \forall i \in \mathcal{A}$	61
Figure 4.1.	PBRAG using (4.8) and large step size γ . Plots share a common legend and time axis.	90
Figure 4.2.	d-PBRAG for unknown rewards with constant step-size using (4.11) on cyclic communication graph. Step-size $\gamma_i^q(t)$ and time-period T were chosen as in Theorem 4.8.3 with different ε and $\nu = 0.1$. The plots share a common legend. (Top) $\varepsilon = 0.9$. (Bottom) $\varepsilon = 0.3$	91
Figure 4.3.	d-PBRAG for unknown rewards with time-varying step-size using (4.11) on cyclic communication graph. Plots share a common legend and time axis.	93
Figure 4.4.	Distributed Hungarian implementation for the same scenario as in Figure 4.3. Plot is shown for one agent only.	93
Figure 4.5.	d-PBRAG for a large number of agents on a cyclic communication graph. Weight evolution of agents 1-10.	94
Figure 4.6.	d-PBRAG for a large number of agents on a cyclic communication graph. Weight evolution of agents 11-20.	95
Figure 4.7.	d-PBRAG for a large number of agents on a cyclic communication graph. Weight evolution of agents 21-30.	96
Figure 4.8.	d-PBRAG for a large number of agents on a cyclic communication graph. Weight evolution of agents 31-40.	96
Figure 4.9.	d-PBRAG for a large number of agents on a cyclic communication graph. Weight evolution of agents 41-50.	97
Figure 4.10.	Assignment evolution of all agents using distributed Hungarian method. The agents are arranged in order from left to right and top to bottom.	98

Figure 5.1.	Proposed exploration framework with 3 robots in close proximity. Proximity determines the communication architecture. Each robot shares the same internal logic framework as in the teal box. The SLAM system provides a local map $\mathcal{M}_l$, a global map $\mathcal{M}$ (using an external map merger scheme) and robot localization x . Then, frontiers $\mathcal{F}$ (clustered using some scheme) and perceived occupancy are extracted from $\mathcal{M}$ and further Behavioral Entropy H_α is calculated for each cluster. This, along with the distance to the cluster, gives the sample Behavioral Utility $\mathbb{E}[R]$. Then, utilities are communicated with other robots to allocate the appropriate frontiers. The navigation planner takes these allocations and finds a traveling salesman tour among them. The path finder then finds a path to the frontier using RRT, which, in turn, sends global and local plans u to the controller.	101
Figure 5.2.	Scaled entropy vs. probability, <i>i.e.</i> $c.H_\alpha(p)$ vs. p . Upper (resp. lower) dotted lines have $c = d_M$ (resp. $c = d_m$). Bold line has $c = 1$	117
Figure 5.3.	Indoor maps used for simulations. The plots use the same colorbar. The robot positions $x_i(0)$ is represented by dots and the circle around each robot shows its initial region of frontier consideration.	121
Figure 5.4.	Outdoor camp map used for simulations. The robot positions $x_i(0)$ is represented by dots and the circle around each robot shows its initial region of frontier consideration.	122
Figure 5.5.	Simulation metric for $r = 2$, $\sigma^m = 2$. X-axis shows different α_i ranges (a, b)	125
Figure 5.6.	Cost for DARPA subT map. The rows from top to bottom have $\sigma^m = 2, 1, 0$ resp. The columns from left to right have $r = 2, 3, 4$ resp.	126
Figure 5.7.	Cost for indoor office map. The rows from top to bottom have $\sigma^m = 2, 1, 0$ resp. The columns from left to right have $r = 2, 3, 4$ resp.	127
Figure 6.1.	Relative error in estimation versus number of samples for example in Section 6.7.1. (a) Relative error in estimated initial condition $\mathbf{x}_0$. (b) Relative error in estimated system matrix $\mathbf{A}$	151
Figure 6.2.	Relative error versus number of samples for example in Section 6.7.1. (a) Relative error in control $\frac{\ \mathbf{u}_{\text{computed}} - \mathbf{u}_{\text{actual}}\ }{\ \mathbf{u}_{\text{actual}}\ }$. (b) Relative error in state.	152
Figure 6.3.	Relative error in state $\frac{\ \mathbf{x}_{\text{computed}} - \mathbf{x}_{\text{actual}}\ }{\ \mathbf{x}_{\text{actual}}\ }$ versus number of samples for example in Section 6.7.2. (a) Relative error in $\mathbf{x}(1)$. (b) Relative error in $\mathbf{x}(2)$	153

LIST OF TABLES

Table 2.1.	Notation.	6
Table 4.1.	Approximate $f_i(q)$ values for Simulations.....	88
Table 5.1.	Mean cost compared with standard algorithms on DARPA subT map. The (a, b) range denotes the range of BE parameters, <i>i.e.</i> $\alpha_i \sim \text{PWC}[a, b]$, $\forall i \in \mathcal{A}$. Least cost for each row is given in bold and italics.	129
Table 5.2.	Mean cost compared with standard algorithms on urban office environment. The (a, b) range denotes the range of BE parameters, <i>i.e.</i> $\alpha_i \sim \text{PWC}[a, b]$, $\forall i \in \mathcal{A}$. Least cost for each row is given in bold and italics.	130
Table 5.3.	Mean cost compared with standard algorithms on outdoor camp map. The (a, b) range denotes the range of BE parameters, <i>i.e.</i> $\alpha_i \sim \text{PWC}[a, b]$, $\forall i \in \mathcal{A}$. Least cost for each row is given in bold and italics.	130
Table 5.4.	Mean cost values for selected α ranges for DARPA SubT map with varying robot team size n . The (a, b) range denotes the range of BE parameters, <i>i.e.</i> $\alpha_i \sim \text{PWC}[a, b]$, $\forall i \in \mathcal{A}$. Least cost for each case is given in bold and italics.	131

LIST OF ALGORITHMS

Algorithm 1.	d-ISBRAG for shared samples	59
Algorithm 2.	Behavioral explorer	123
Algorithm 3.	<code>communicate</code>	124
Algorithm 4.	Sequential control computation with least square error estimation	149

ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. Sonia Martínez, who simultaneously made me feel welcome in foreign lands, nurtured my curiosity, and prevented me from straying far from the designated goal; and whose faith in me helped me overcome hurdles and develop into an independent researcher.

I thank my committee members Dr. Miroslav Krstic, Dr. Nikolay Atanasov, and Dr. Jorge I. Poveda for taking the time to review my work and provide valuable feedback. I would like to thank the Army Research Lab for supporting my work with a grant and hosting me during the summers of 2023 and 2024 for two research internships. In particular, I would like thank to Dr. Carlos Nieto-Granda and Dr. Aamodh Suresh for mentoring me in completing the associated robotics project.

I would like to express my special gratitude to Suborno da, Satadeepa di, and little Coco (Somriddho) for accepting me as a part of their family and being there for me through the thick and thin of it all. I also thank my closest friends Scott (Addams), Parth, Aneesh, and Subhadeep for making my stay at UCSD a memorable one. I thank Rakshit, Toshit, Mrinmoy, Chirantan, Abhishek, Neilabh, Ayantik, Vipul, Darshan, Sovan, Shuvam, Sheshnag (Shubhankar), Will, Biswarup, Bijit, Aneeshan, Anwasha, Joydeep, and Suryadeep for helping me reach this point in my life. I cherish every moment I have spent with you all.

I would also like to thank the faculty and staff of the Mechanical and Aerospace Engineering Department of UCSD for providing me with this wonderful opportunity to conduct research in one of the top schools in USA. I take this opportunity to thank my remaining lab-mates (past and present); Masih, Pol, Scott (Brown), Xuting, Vineet, Arnau, Ahmed, Mohammed, Dhruv, Ben (Benjamin), Micheal, Paul, William, Guillem, Azra, Steven, Aleix, Adil, Xuwei, and Prof. Cortés (Dr. Jorge Cortés). Thank you for all the insightful conversations, discussions, and meetings that inspired many ideas for my work.

I am grateful to all my mentors, especially Dr. Pavankumar Tallapragada, and Dr.

Ankur Bhattacharjee for cultivating my passion for conducting rigorous scientific research. I would also like to acknowledge my co-authors Prashil, Zeru, Guanxiang, Bernie (Bailin), Jonathan, and Allen (Cheng-Chou). Thank you for putting in the time and effort with me to finish interesting projects. I thank Tanumoy, Amit da, Diya (Nibedita), Souvick da, Prakriti di, Avik da, and Janu di for making my stay in San Diego enjoyable and exciting. And to all my well-wishers, thank you for your continued support.

Last, but in no way the least, I express my profound gratitude to my wonderful parents Dr. Ratna Mondal and Dr. Gautam Mondal and my sister Debadrita for standing beside me through every adversity of life. Everything I am today is because of you.

The research reported in this dissertation was partially supported by the Army Research Lab grant: W911NF-23-2-0009.

Chapter 3 is a reprint of material that has been submitted for publication as “Distributionally Robust Nash Equilibrium Seeking with Partial Observations and Distributed Communication,” N. Mandal, and S. Martínez.

Chapter 4 is a reprint of material as it appears in “Distributed task allocation for self-interested agents with partially unknown rewards,” N. Mandal, M. Khajenejad, and S. Martínez, *IEEE Transactions on Automatic Control*, 70(9):6284–6291, 2025.

Chapter 5 is a reprint of material as it appears in “Behaviorally heterogeneous multi-agent exploration using distributed task allocation,” N. Mandal, A. Suresh, C. Nieto, and S. Martínez, *IEEE Robotics and Automation Letters*, pages 2983 – 2990, 2026.

Chapter 6 is a reprint of material as it appears in “Control of discrete-time LTI systems using stochastic ensemble systems,” N. Mandal, M. Khajenejad, and S. Martínez, in *SIAM Conf. on Control and Its Applications (CT)*, Philadelphia, PA, USA, July 2023.

The dissertation author was the primary investigator and author of all of the papers mentioned above.

VITA

- 2017 B. Tech. in Electrical Engineering, Institute of Engineering & Management, Kolkata, India
- 2021 M. Tech. (Research) in Electrical Engineering, Indian Institute of Science, Bengaluru, India
- 2026 Ph.D. in Engineering Sciences (Mechanical Engineering), University of California San Diego, USA

PUBLICATIONS

N. Mandal, and S. Martínez, “Distributionally Robust Nash Equilibrium Seeking with Partial Observations and Distributed Communication,” under review in *Automatica*.

N. Mandal*, Z. Li*, G. Jing*, S. Addams, S. Martínez, and J. Cortés, “Feasibility-Certified Transport Using an Interconnected Robot Team and Distributed Communication,” in final stages of preparation. *Equal contribution.

P. Wankhede, **N. Mandal**, S. Martínez, and P. Tallapragada, “Opinion Dynamics for Multi-Topic Constrained Resource Allocation,” in final stages of preparation.

N. Mandal, A. Suresh, C. Nieto-Granda, and S. Martínez, “Behaviorally Heterogeneous Multi-agent Exploration Using Distributed Task Allocation,” *IEEE Robotics and Automation Letters*, pages 2983 – 2990, 2026.

N. Mandal, M. Khajenejad, and S. Martínez, “Distributed Task Allocation for Self-Interested Agents With Partially Unknown Rewards,” *IEEE Transactions on Automatic Control*, 70(9):6284–6291, 2025.

P. Wankhede, **N. Mandal**, S. Martínez, and P. Tallapragada, “Multi-Topic Projected Opinion Dynamics for Resource Allocation,” in *IEEE Conf. on Decision and Control*, pages 3469–3476, Rio de Janeiro, Brazil, December 2025.

P. Wankhede, **N. Mandal**, S. Martínez, and P. Tallapragada, “Opinion Dynamics for Utility Maximizing Agents: Exploring the Impact of Resource Penalty,” *IEEE Transactions on Control of Network Systems*, 12(1):5–17, 2025.

N. Mandal, M. Khajenejad, and S. Martínez, “Control of Discrete-Time LTI Systems Using Stochastic Ensemble Systems,” in *SIAM Conf. on Control and Its Applications (CT)*, Philadelphia, PA, USA, July 2023.

ABSTRACT OF THE DISSERTATION

Distributionally Robust Nash Equilibrium Seeking
With Applications to Task Allocation Among Self-Interested Agents

by

Nirabhra Mandal

Doctor of Philosophy in Engineering Sciences (Mechanical Engineering)

University of California San Diego, 2026

Professor Sonia Martínez, Chair

The behaviors of complex systems with interacting decision-making agents are governed not only by each agent's selfish response to others' decisions, but also by unforeseen environmental uncertainties. As such, analytical characterization of equilibrium behaviors of such systems must be robust to unknown disturbances. This dissertation explores stochastic one-shot games, where the agents' utilities depend not only on the collective strategy profiles but also on some well-behaved randomness; and a distributionally robust version of such games. The main focus is on providing conditions under which these games admit a Nash equilibrium (NE), relating the NE of the games, and coming up

with distributed algorithms that converge near an NE, when it exists. Another focus of this dissertation is on applying these algorithms to solve task allocation problems among self-interested agents in a distributed way.

We first study the game where each decision-maker is agnostic to the random variable’s underlying distribution, but have access to finitely many i.i.d. samples generated from it. To hedge against the unknown uncertainty, each agent plays a distributionally robust game and aims to maximize the worst-case expected utility over a Wasserstein ball around the sample average distribution. In this setting, we provide conditions under which the game has a non-empty set of distributionally robust Nash equilibria (DRoNE) and then characterize the closeness of the DRoNE set to the NE of the associated stochastic game. We then propose an Inertial, Supported, Better Response, Ascending superGradient dynamics (ISBRAG) that seeks the DRoNE set. This forms the basis of a distributed version (d-ISBRAG) where agents estimate others’ strategies by means of a dynamic consensus subroutine over a directed communication network.

Next, leveraging this algorithm, we propose a novel solution to a task allocation problem, by which a group of agents assigns a discrete set of tasks in a distributed manner. In this setting, heterogeneous agents have individual rewards for doing each task; however, these rewards are only known asymptotically. The assignment problem is formulated by means of a combinatorial partition game, with no constraints on the number of tasks per agent. We relax this into a weight game, which together with the former, are shown to contain the optimal task allocation in the corresponding NE set. We then propose a distributed, Projected, Best-Response, Ascending Gradient dynamics (d-PBRAG) as a simplification of d-ISBRAG. This distributed online algorithm can deal with a converging reward sequence by means of an agreement sub-routine.

Finally, we deploy our algorithms to study a problem of multi-agent exploration with behaviorally heterogeneous robots. Each robot maps its surroundings using SLAM and identifies a set of areas of interest (AoIs) or frontiers that are the most informative to

explore next. The robots assess the utility of going to a frontier using Behavioral Entropy (BE) and then determine which frontier to visit via the d-PBRAG algorithm. We test our algorithm in simulation, where we explore the effect of sensing radii, sensing accuracy, and heterogeneity among robotic teams with respect to the time taken to complete exploration and path traveled.

All together, these contributions address some of the significant challenges in designing and deployment of distributed algorithms for NE seeking in robust games. In addition, at the end, we provide robust control properties of a class of stochastic ensemble systems that consists of families of random variables. We relate the reachability properties of the stochastic ensemble system with that of the limiting, discrete, linear, time-invariant (DLTI) system, and provide a method to combine the control inputs from the stochastic ensemble systems to compute a control input for the DLTI system. We finally characterize variance of state and control when the data is updated sequentially in a streaming fashion.

Chapter 1

Introduction

1.1 Motivation

Complex interacting systems with reward oriented agents are staples of today's increasingly interconnected world. As collective strategies deployed by decision-making agents have far-reaching implications in various fields, e.g. robotics, artificial intelligence, or operations management; the analytical characterization of their behavior is of the utmost importance. This can be studied succinctly through the lens of game theory, the science of strategic decision making. However, an individual's strategy is shaped not only by the selfish response to others' decisions, but also by unforeseen environmental uncertainties. This motivates us to explore stochastic one-shot games where the agents' utilities depend not only on the collective strategy profiles but also on some well-behaved randomness. In particular, we assume that the agents are agnostic to the random variable's underlying distribution and instead have access to finitely many i.i.d. samples from this random variable. The main focus is on proposing and analyzing a distributionally robust game where the agents are interested in maximizing their worst-case expected utility over some uncertainty set built using the samples. We are particularly interested in how the equilibria of such a robust game relate to the equilibria of the underlying stochastic game. Moreover, once we provide guarantees of existence, we are also interested to producing algorithms that seek out these equilibria in a distributed way. Such methods are useful in

various problems related to controlling a group of decision making agents towards a class of desired equilibria.

As an application of such a framework we turn towards the field of task allocation among self-interested agents. For example, a prototypical coordination problem aims to find an efficient assignment of a group of agents to a collection of tasks. The tasks can range from abstract objectives to specific physical jobs, the nature of which may not be known. The agents may have heterogeneous capabilities, and react to different sets of incentives that are learned progressively. In the field of robotics, exploring uncertain, hazardous environments is often handled by such coordination of multiple autonomous robots, with the goal of improving efficiency, adaptiveness, and robustness to individual failures. To deal with this, we convert the task assignment problem into a robust non-cooperative game whose equilibria coincide with the optimal task allocation solution. Further, such teams of robots might not have individual capabilities for enhanced computation or storage. This motivates us to propose distributed algorithms that are able to compute the equilibria of the aforementioned robust game in finite time. As a side consequence, we observe that having a team of agents with heterogeneous behaviors is beneficial.

1.2 Organization and Contributions

This dissertation studies the application of distributionally robust games towards solving multi-agent task allocation problems with uncertain rewards. We specifically deal with situations where the underlying distribution of the uncertainty is unknown, but we have access to finitely many samples from this well-behaved random variable. In Part I, we deal solely with Distributionally Robust Games (DRG). We relate their Nash equilibria (NE) to the NE of the underlying stochastic game, and provide distributed algorithms to seek these NE when they exist. Part II deals with the applications of the framework proposed in Part I, prominently towards task allocation among self interested

agents. Chapter 4 proposes a distributed task allocation scheme using a game theoretic method, and Chapter 5 uses this method to solve multi-robot exploration problems with heterogeneous robot teams. Finally, in Chapter 6, we provide an application to ensemble control of DLTI systems, which can be used to steer multi-agent groups randomly interacting with each other.

We now provide a more detailed overview of each chapter.

Chapter 2 lists our major notations and some preliminary concepts and results that are used frequently in the later chapters. We begin by introducing concepts from measure theory, including important definitions and major concentration results. Next, we provide definitions and results from the literature of input-to-state-stability with generalized versions that utilize size functions. Then, we briefly mention some non-smooth analysis techniques for convex functions. We conclude the chapter with basic definitions from graph theory, which is used to model the communication network for our distributed algorithms.

Chapter 3 deals with Distributionally Robust Games. Specifically, we study stochastic one-shot games where agents' utilities depend on the collective strategy profiles of other agents as well as on some well-behaved randomness. While each decision-maker is agnostic to the random variable's underlying distribution, they have access to finitely many i.i.d. samples generated from it. We consider two cases: one where samples are shared; and another, more special one, where samples are individually accessible. To hedge against the unknown uncertainty, each agent plays a distributionally robust game and aims to maximize the worst-case expected utility over a Wasserstein ball around the sample average distribution. In this setting, we provide conditions under which the game has a non-empty set of distributionally robust Nash equilibria (DRoNE) and then characterize the closeness of the DRoNE set to the Nash equilibria (NE) of the associated stochastic game. We then propose an inertial, supported, better response, ascending supergradient dynamics (ISBRAG) that seeks the DRoNE's when the distributionally

robust game possesses what we term as amicable supergradients. This forms the basis of a distributed version (d-ISBRAG) where agents estimate others' strategies by means of a dynamic consensus subroutine over a directed communication network. While initially the distributed algorithm works in the case where agents have individual samples, we later extend this to the case of shared observations under certain simplifying assumptions. This involves analyzing a tractable reformulation of the distributionally robust optimization problem and solving it in a distributed manner to compute the required supergradients.

Chapter 4 specializes the game to find a solution to a task allocation problem, by which a group of agents assigns a discrete set of tasks in a distributed manner. In this setting, heterogeneous agents have individual preferences and associated rewards for doing each task; however, these rewards are only known asymptotically. The assignment problem is formulated by means of a combinatorial partition game for known rewards, with no constraints on the number of tasks per agent. We relax this into a weight game, which together with the former, are shown to contain the optimal task allocation in the corresponding set of Nash equilibria (NE). We then propose a projected, best-response, ascending gradient dynamics (PBRAG) that converges to an NE in finite time. This forms the basis of a distributed on line version that can deal with a converging sequence of rewards by means of an agreement subroutine.

Chapter 5 uses the previous algorithms to study a problem of multi-agent exploration with behaviorally heterogeneous robots. Each robot maps its surroundings using SLAM and identifies a set of areas of interest (AoIs) or frontiers that are the most informative to explore next. The robots assess the utility of going to a frontier using Behavioral Entropy (BE) and then determine which frontier to visit via a distributed task assignment scheme. We convert the task assignment problem into a non-cooperative game and use the distributed algorithm d-PBRAG to converge to the Nash equilibrium, which we have shown to be the optimal task allocation solution. For unknown utility cases, we provide robust bounds using approximate rewards. We test our algorithm (which has less

communication cost and fast convergence) in simulation, where we explore the effect of sensing radii, sensing accuracy, and heterogeneity among robotic teams with respect to the time taken to complete exploration and path traveled. We observe that having a team of agents with heterogeneous behaviors is beneficial.

In **Chapter 6** we study the control properties of a class of stochastic ensemble systems that consists of families of random variables. These random variables provide an increasingly good approximation of an unknown discrete, linear-time invariant (DLTI) system, and can be obtained by a standard, data-driven procedure. Our first result relates the reachability properties of the stochastic ensemble system with that of the limiting DLTI system. We then provide a method to combine the control inputs obtained from the stochastic ensemble systems to compute a control input for the DLTI system. Later, we deal with a particular kind of stochastic ensemble system generated from realizing Bernoulli random variables. For this, we characterize the variance of the computed state and control. We also do the same for a situation where the data is updated sequentially in a streaming fashion.

Finally, **Chapter 7** summarizes the main contributions of this dissertation. It also provides a vision for future research that might build on our results.

Chapter 2

Preliminaries

This chapter introduces background material necessary for the development of later chapters. We begin with standard notations that are used throughout the dissertation. Specialized notation will be introduced in each chapter if needed. Then we collect concepts from measure theory that help us deal with providing uncertainty quantifications for the randomness. Following that, we present definitions and results from the field of input-to-state stability. This helps us in providing ultimate bounds for the behavior of the algorithms. Finally, we present basic ideas from non-smooth convex analysis and graph theory.

2.1 Notation

We begin this chapter with a table of the general notation used throughout the rest of this work. More specific notation will be defined in individual chapters.

Table 2.1. Notation.

Symbol	Description
Spaces and Sets	
$\mathbb{R}$	Real numbers.
$\mathbb{R}_{\geq 0}(\mathbb{R}_{> 0})$	Non-negative (positive) real numbers.
$\mathbb{Z}_{\geq 0}(\mathbb{Z}_{> 0})$	Non-negative (positive) integers.

Table 2.1. Notation (continued).

Symbol	Description
$\mathbb{R}^n$	n -dimensional Euclidean space.
$\mathbb{R}^{n \times m}$	Set of $n \times m$ real matrices.
Set operations	
$ \mathcal{S} $	Cardinality of set $\mathcal{S}$.
$\mathcal{S}^n$	n Cartesian product of set $\mathcal{S}$ with itself.
$\mathcal{S}_1 + \mathcal{S}_2 := \{s_1 + s_2 \mid s_1 \in \mathcal{S}_1, s_2 \in \mathcal{S}_2\}$	Minkowski sum of sets $\mathcal{S}_1$ and $\mathcal{S}_2$.
$\mathcal{S}_1 \setminus \mathcal{S}_2$	Set difference.
Vectors and matrices	
$\mathbf{1}$	Vector with all entries <i>one</i> .
$\mathbf{0}$	Vector with all entries <i>zero</i> .
$\mathbf{I}$	Identity matrix.
$\mathbf{O}$	Zero Matrix.
$\mathbf{M}^\dagger$	Moore-Penrose pseudoinverse of $\mathbf{M}$.
Distances and norms	
$\ \mathbf{x}\ _p$	p -norm of $\mathbf{x} \in \mathbb{R}^n$, with $p \geq 1$.
$\ \mathbf{x}\ := \sqrt{\mathbf{x}^\top \mathbf{x}}$	2-norm of $\mathbf{x} \in \mathbb{R}^n$, unless otherwise mentioned.
$d(\mathbf{x}, \mathcal{S}) := \inf_{\mathbf{y} \in \mathcal{S}} \ \mathbf{x} - \mathbf{y}\ $	Distance of vector $\mathbf{x}$ to the set $\mathcal{S}$.
$\mathcal{B}_\varepsilon(x) := \{y \in \mathcal{X} \mid d(x, y) \leq \varepsilon\}$	ε -radius closed ball around $x \in \mathcal{X}$.
$\ f\ $	L^2 -norm of measurable function $f : \mathbb{R}^n \rightarrow [0, \infty)$.
Misc.	
$\Pr\{\cdot\}$	Probability of an event over a measurable space.
$f \circ g$	Function composition.
$\emptyset$	Empty set.

2.2 Measure theory and concentration results

Suppose $\mathcal{F}$ is a σ -algebra on $\mathbb{R}^m$; and let μ and ν be measures on the measure space $(\mathbb{R}^m, \mathcal{F})$. Then, ν is said to be absolutely continuous w.r.t. μ (denoted as $\nu \ll \mu$) if $\mu(\mathcal{S}) = 0$ implies that $\nu(\mathcal{S}) = 0$. Suppose that $f : \mathbb{R}^m \rightarrow \mathbb{R}^m$ is a measurable function. Then, the *f-pushforward measure* of μ is a new measure denoted by $f_{\#}\mu$; and defined as $f_{\#}\mu(\mathcal{S}) = \mu(f^{-1}(\mathcal{S}))$, $\forall \mathcal{S} \in \mathcal{F}$. Finally, we denote the Dirac delta function at $\mathbf{x}$ using $\delta_{\mathbf{x}}$; and the multivariate Gaussian distribution with mean $\mathbf{x}$ and covariance Σ as $\mathbf{N}_{\mathbf{x}}^{\Sigma}$.

Assume that $\xi \in \mathbb{R}^m$ is a random variable with an unknown probability distribution, $\mathbb{P}^*$. When $\mathbb{P}^*$ is supported on $\Xi \subseteq \mathbb{R}^m$ and $\exists a > 1$ such that $\mathbb{E}_{\xi \sim \mathbb{P}^*} [\exp(\|\xi\|^a)] < \infty$; we say that $\mathbb{P}^*$ is light-tailed. Let $\mathcal{M}(\Xi)$ be the set of all probability distributions $\mathbb{Q}$ on Ξ with bounded first-order and second-order moments. Then, the 1-Wasserstein distance (we drop the 1 in the sequel for brevity) between two distributions $\mathbb{Q}_1, \mathbb{Q}_2 \in \mathcal{M}(\Xi)$ is defined as [1]

$$d_W(\mathbb{Q}_1, \mathbb{Q}_2) := \inf_{\pi \in \Pi(\mathbb{Q}_1, \mathbb{Q}_2)} \int_{\Xi^2} \|\xi_1 - \xi_2\|_1 \pi(d\xi_1, d\xi_2), \quad (2.1)$$

where, $\Pi(\mathbb{Q}_1, \mathbb{Q}_2)$ is the set of joint probability distributions of ξ_1 and ξ_2 with marginals $\mathbb{Q}_1$ and $\mathbb{Q}_2$ respectively. Further,

$$\mathcal{B}_{\varepsilon}(\mathbb{P}) := \{\mathbb{Q} \in \mathcal{M}(\Xi) \mid d_W(\mathbb{Q}, \mathbb{P}) \leq \varepsilon\},$$

is the closed Wasserstein ball of radius ε around $\mathbb{P} \in \mathcal{M}(\Xi)$. Under some conditions, the aforementioned Wasserstein distance can be written in different equivalent ways, which become useful later. When $\mathbb{Q}_1$ is absolutely continuous w.r.t. the Lebesgue measure, the

Monge formulation of optimal transport gives,

$$d_W(\mathbb{Q}_1, \mathbb{Q}_2) = \inf \left\{ \int_{\Xi} \|\xi - T(\xi)\|_1 \mathbb{Q}_1(d\xi) \mid T_{\#}\mathbb{Q}_1 = \mathbb{Q}_2 \right\}, \quad (2.2)$$

where $T : \mathbb{R}^m \rightarrow \mathbb{R}^m$ is a measurable function, or deterministic transport map. Finally, the Kantorovich-Rubinstein description of the Wasserstein distance is,

$$d_W(\mathbb{Q}_1, \mathbb{Q}_2) = \sup_{f \in \mathcal{L}} \left\{ \int_{\Xi} f(\xi) \mathbb{Q}_1(d\xi) - \int_{\Xi} f(\xi) \mathbb{Q}_2(d\xi) \right\}, \quad (2.3)$$

where $\mathcal{L} := \{f \mid |f(\xi_1) - f(\xi_2)| \leq \|\xi_1 - \xi_2\|, \forall \xi_1, \xi_2 \in \Xi\}$ is the set of all Lipschitz continuous functions on Ξ with Lipschitz constant 1.

Let $\{\widehat{\xi}(k)\}_{k=1}^N \subseteq \mathbb{R}^m$ be i.i.d. samples generated by a distribution $\mathbb{P}^*$. The following result characterizes the closeness of $\mathbb{P}^*$ to the sample average distribution $\widehat{\mathbb{P}}^N := \frac{1}{N} \sum_{k=1}^N \delta_{\widehat{\xi}(k)}$. The result is useful to provide guarantees from *distributionally robust optimization* [2].

Theorem 2.2.1. [3] Suppose $\theta \in (0, 1)$, and

$$\varepsilon = \begin{cases} \left(\frac{\log(c_1 \theta^{-1})}{c_2 N} \right)^{1/\max\{m, 2\}}, & \text{if } N \geq \frac{\log(c_1 \theta^{-1})}{c_2}; \\ \left(\frac{\log(c_1 \theta^{-1})}{c_2 N} \right)^{1/a}, & \text{if } N < \frac{\log(c_1 \theta^{-1})}{c_2}; \end{cases} \quad (2.4)$$

where c_1, c_2 are positive constants that only depend on a and m . Then, $\Pr \{ \mathbb{P}^* \in \mathcal{B}_{\varepsilon}(\widehat{\mathbb{P}}^N) \} \geq 1 - \theta$. •

2.3 Size functions and input-to-state stability

We recall that a continuous function $\sigma : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is class- $\mathcal{K}$ (i.e. $\sigma \in \mathcal{K}$) if it is strictly increasing and $\sigma(0) = 0$. Further, σ is class- $\mathcal{K}_{\infty}$ (i.e. $\sigma \in \mathcal{K}_{\infty}$) if $\sigma \in \mathcal{K}$ and $\sigma(x) \rightarrow \infty$ as $x \rightarrow \infty$. Finally, a continuous function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is class- $\mathcal{KL}$

(i.e. $\beta \in \mathcal{KL}$) if $\forall y \in \mathbb{R}_{\geq 0}$, $\beta(\cdot, y) \in \mathcal{K}$ and $\forall x \in \mathbb{R}_{\geq 0}$, $\beta(x, \cdot)$ is decreasing with $\beta(x, y) \rightarrow 0$ as $y \rightarrow \infty$. The following notion will be used in the stability analysis in the sequel.

Definition 2.3.1. (*Size function [4]*). A function $\omega : \mathbb{R}^n \rightarrow \mathbb{R}$ is said to be a size function for the compact subset $\mathcal{S} \subset \mathbb{R}^n$ if ω is i) continuous; ii) positive definite with respect to $\mathcal{S}$, i.e. $\omega(\mathbf{x}) = 0$, if $\mathbf{x} \in \mathcal{S}$ and $\omega(\mathbf{x}) > 0$, if $\mathbf{x} \notin \mathcal{S}$; and iii) proper, i.e. $\forall r \in \mathbb{R}_{\geq 0}$, the sublevel set $\{\mathbf{x} \mid \omega(\mathbf{x}) \leq r\}$ is a compact subset of $\mathbb{R}^n$. •

Size functions have also been referred to as *proper indicators* in [5, 6]. For example, the previously defined distance function $d(\cdot, \mathcal{S})$ is a size function for $\mathcal{S}$. With this, the following result provides a useful comparison between size functions and class- $\mathcal{K}_\infty$ functions.

Lemma 2.3.2. [4] Suppose ω_1, ω_2 are size functions for $\mathcal{S} \subset \mathbb{R}^n$. Then there exists functions $\sigma_1, \sigma_2 \in \mathcal{K}_\infty$ such that

$$\sigma_1(\omega_2(\mathbf{x})) \leq \omega_1(\mathbf{x}) \leq \sigma_2(\omega_2(\mathbf{x})), \quad \forall \mathbf{x} \in \mathbb{R}^n. \quad \bullet$$

The previously discussed concepts are used to recall the notion of *input-to-state stability* for a non-linear autonomous system

$$\mathbf{x}(t+1) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \quad \forall t \in \mathbb{Z}_{\geq 0}; \quad (2.5)$$

with continuous $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$.

Definition 2.3.3. (*Input-to-state stability [7]*). Suppose ω is a size function for $\mathcal{S} \subset \mathbb{R}^n$. Then, (2.5) is said to be *input-to-state stable (ISS)* if there exists $\beta \in \mathcal{KL}$ and $\gamma \in \mathcal{K}$, such that $\forall \mathbf{x}_0 \in \mathbb{R}^n$ and $\forall \{\mathbf{u}(t)\}_{t \in \mathbb{Z}_{\geq 0}}$ (bounded), the solution $\{\mathbf{x}(t)\}_{t \in \mathbb{Z}_{\geq 0}}$ satisfies, $\forall t \in \mathbb{Z}_{\geq 0}$,

$$\omega(\mathbf{x}(t)) \leq \max \left\{ \beta(\omega(\mathbf{x}_0), t), \gamma \left(\sup_{\tau \in [0, t]} \|\mathbf{u}(\tau)\| \right) \right\}. \quad \bullet$$

Finally, (2.5) is ISS [6] [4] if and only if there exists a continuous (Lyapunov) function $V : \mathbb{R}^n \rightarrow \mathbb{R}$; $\sigma_1, \sigma_2, \sigma_3 \in \mathcal{K}_\infty$; and $\gamma \in \mathcal{K}$ such that, $\forall \mathbf{x} \in \mathbb{R}^n, \forall \mathbf{u} \in \mathbb{R}^m$,

$$\sigma_1(\omega(\mathbf{x})) \leq V(\mathbf{x}) \leq \sigma_2(\omega(\mathbf{x})), \quad (2.6a)$$

$$V(\mathbf{f}(\mathbf{x}, \mathbf{u})) - V(\mathbf{x}) \leq -\sigma_3(V(\mathbf{x})) + \gamma(\|\mathbf{u}\|). \quad (2.6b)$$

2.4 Convex analysis

A set $\mathcal{D} \in \mathbb{R}^n$ is said to be convex if for any $\mathbf{x}, \mathbf{y} \in \mathcal{D}$, $\lambda \mathbf{x} + [1 - \lambda] \mathbf{y} \in \mathcal{D}$, $\forall \lambda \in [0, 1]$. For a convex set $\mathcal{D}$ and a point $\mathbf{x} \in \mathcal{D}$, $\mathcal{NC}_{\mathcal{D}}(\mathbf{x}) := \{\mathbf{v} \in \mathbb{R}^n \mid \mathbf{v}^\top (\mathbf{y} - \mathbf{x}) \leq 0, \forall \mathbf{y} \in \mathcal{D}\}$ is the normal cone at $\mathbf{x}$ to the set $\mathcal{D}$.

A function $f : \mathcal{D} \rightarrow \mathbb{R}$ with a convex domain $\mathcal{D} \subseteq \mathbb{R}^n$ is convex if $f(\lambda \mathbf{x} + [1 - \lambda] \mathbf{y}) \leq \lambda f(\mathbf{x}) + [1 - \lambda] f(\mathbf{y})$, $\forall \mathbf{x}, \mathbf{y} \in \mathcal{D}$, $\forall \lambda \in [0, 1]$. Further, f is concave iff $-f$ is convex. The concavity of functions can be characterized via *supergradients*, as defined next.

Definition 2.4.1. (*Supergradient of a concave function*). Consider the concave function f defined earlier. Then $\boldsymbol{\zeta} \in \mathbb{R}^n$ is said to be a supergradient of f at $\mathbf{x}$ if

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \boldsymbol{\zeta}^\top [\mathbf{y} - \mathbf{x}], \forall \mathbf{y} \in \mathcal{D}. \quad (2.7)$$

The set of all supergradients of f at $\mathbf{x}$ is $\partial f(\mathbf{x})$. •

For such a concave function, $\partial f(\mathbf{x})$ is always a closed convex set. Consider a function $f : \mathcal{D}_1 \times \mathcal{D}_2 \rightarrow \mathbb{R}$ (with $\mathcal{D}_1 \subseteq \mathbb{R}^n$ convex) for which $f(\cdot, \mathbf{y})$ is concave, $\forall \mathbf{y} \in \mathcal{D}_2$. Then, we use $\partial_{\mathbf{x}}[f(\mathbf{x}, \mathbf{y})]|_{\bar{\mathbf{x}}}$ to denote the set of supergradients of $f(\cdot, \mathbf{y})$ at $\bar{\mathbf{x}} \in \mathcal{D}_1$. The equivalent notion for a convex function is referred to as a *subgradient*.

2.5 Graph theory

A *directed graph* [8], $\mathcal{G} := (\mathcal{A}, \mathcal{E})$, is a tuple that has the following elements,

- a) a set of *nodes* (here agents $\mathcal{A}$); and
- b) a set of *arcs* $\mathcal{E} \subseteq \mathcal{A} \times \mathcal{A}$ between the nodes.

The set $\mathcal{N}_i := \{j \in \mathcal{A} \mid (j, i) \in \mathcal{E}\}$ denotes the (in) *neighbors* of node $i \in \mathcal{A}$ and $\overline{\mathcal{N}}_i := \mathcal{N}_i \cup \{i\}$. A *path* is an ordered set of non-repeating nodes such that each tuple of adjacent nodes belongs to $\mathcal{E}$. The graph $\mathcal{G}$ is said to be strongly connected if there exists a path from every node to every other node. The *diameter* of the graph $\text{diam}(\mathcal{G})$ is the length of the largest possible path between any two nodes.

Part I

Distributionally Robust Games

Chapter 3

Distributionally Robust Games And An Algorithm To Seek Their Nash Equilibria

3.1 Introduction

The design of complex interacting systems via reward-oriented agents is transforming real-world applications in autonomous systems, critical infrastructure, and mixed human-robot teams. Strategic settings, where emerging multi-agent behaviors depend on the selfish actions of its members, can greatly be degraded by unknown environmental uncertainty, potentially furthering their inherent inefficiencies. With broad and distributed accessibility to data, it is now possible to construct more powerful agents that exploit new techniques such as distributionally robust optimization (DRO). Although this can result in more robust equilibria, the question remains as to how to effectively leverage this framework to design provably-correct distributed learning algorithms for multi-agent networks. Motivated by this, we study here the integration of DRO with the game-theoretic framework, and propose novel learning algorithms that can practically converge to an associated equilibrium.

3.2 Literature Review

Non-cooperative Game Theory [9, 10], provides a powerful framework for the analysis and design of engineered multi-agent systems. Recent applications include multi-vehicle networks [11, 12], telecommunication [13] and cyber-physical systems [14]. In particular, the consideration of self-interested agents allows for the modeling of human behavior [15, 16], and the evaluation of their impact in socio-technical systems [17], which opens the door to incentive design [18, 19].

In this framework, the concept of Nash equilibria (NE) characterizes those lucrative equilibrium strategies that rational agents adopt in response to others' decisions. Thus, a variety of Nash-seeking algorithms; *e.g.*, [20, 21, 22, 23], have been developed to learn such equilibria. Often, these algorithms require strong assumptions on utility functions and their derivatives. In practice, only imperfect information about other agents' strategies is available [23]. While the early work [20] addresses this lack of information by means of fictitious play [20], this requires a complete communication graph among the agents. Subsequently, this lack of information has been mostly addressed via consensus [24, 22], and gossip-based algorithms [25], which borrow ideas from the literature on distributed optimization [26] and dynamic consensus [27]. When utilities are differentiable, these Nash-seeking algorithms can be interpreted as passivity-based designs [28] [29], where the gradient play of agents are corrected using an estimation term. In general, equilibria associated with games with “convex” utilities over continuous strategy sets can be characterized via variational inequalities [30, 31], which has been exploited in the development of Nash-seeking algorithms [30, 32]. In particular, [30] uses proximal operators as regularization terms similarly to how [33] use proximal terms for bi-level gradient descent. We refer the reader to [23] for an extensive list of distributed Nash-seeking algorithms.

Optimization under uncertainty is studied extensively in the field of stochastic programming [34] and its robust counterpart [35]. However, these setups often require

knowledge of the underlying probability distribution, or over-approximating bounds on the support of this distribution. When many samples are available, deterministic optimization algorithms using sample average approximations [36] provide appropriate results. For limited samples, distributionally robust optimization (DRO) formulations, where one tries to maximize the worst-case expected reward over an ambiguity set related to the unknown data-generating distribution, can provide decisions with a high confidence performance. These problems have been studied since [37] and, more recently in [38, 2], and [39]. When dealing with continuous random variables, DRO problems translate in an infinite-dimensional optimization. As such, [2] leverages concentration of measure inequalities to provide tractable finite-dimensional reformulations under certain conditions, and out-of-sample guarantees. This form of Wasserstein DRO is extended to the case of streaming samples in [40] and to the case where samples are obtained from a dynamical system under partial observations [41]. In this chapter, we apply results from DRO over a continuous sample space with fixed number of samples.

Distributional robust game theory [42, 43, 44] is receiving increasing attention, with current research differing in their treatment of uncertainty. The work [42] focuses on discrete random variables, while [43, 44, 45] handle continuous ones. In these studies every agent is affected by their own uncertainty, and neither account for the effect of limited observations from shared samples among players in the equilibrium formation. More recently, [45] produces a tractable reformulation of a distributional robust game for finite strategy spaces and mixed strategy equilibria. Further, [45] shows that, as the number of data samples increases to infinity, the distributionally robust equilibria asymptotically approach those of the stochastic game. However, no out-of-sample guarantees under a finite number of data points is provided. A similar result is provided in [46] for generalized Nash games. However, here the uncertainty is used to define chance constraints on the the feasible set of strategies, which does not directly affect the utilities of the agents. None of these works aim to solve the robust equilibrium seeking problem in a distributed

manner. The work [47] proposes a distributionally robust Nash equilibrium seeking algorithm. However, it assumes differentiability of the associated utilities to produce a monotone variational inequality reformulation that facilitates a pseudo-gradient update towards the Nash equilibria. On a different take, the work in [48] rather assumes that it is the opponents' strategies that are generated by an unknown distribution, and study an associated strategically robust game. Distributionally robust Stackleberg games are considered in [49], where the authors assume strong Lipschitz assumptions on the utility functions in order to provide gradient based updates for the leader-follower dynamic game. In previous work [50], we studied a distributionally robust task allocation game among self-interested agents. However these results apply for specific utility functions that model task allocation and that are linear in the random variables.

3.3 Contributions

This chapter considers a distributionally robust game (DRG) among a group of strategic agents. In this setting, the distributionally robust utilities are in general non-smooth. Unlike previous works in the literature, agents construct their ambiguity sets from partially observed samples obtained as the image of an observation function.

Assuming that these measurements can be stacked to define an invertible mapping, we propose a modification of the ambiguity ball that contains the original distribution with high confidence. Roughly speaking, this approximation requires that the observation mapping does not deviate too much from an optimal transport of the unknown distribution to the empirical one. We then identify conditions that guarantee the existence of NE for the DRG and the associated stochastic game. These conditions, which are standard requirements for NE of continuous games, generalize well to DRGs. In particular, the aforementioned uncertainty bounds allow us to obtain a finite-sample error bound between the distributionally robust NE (DRoNE) and the NE of the stochastic game. As robust

utilities may be non-smooth, our DRoNE seeking strategy leverages supergradients and inertial terms to produce an algorithm (ISBRAG) that replicates monotonicity-like behavior for the game. The algorithm is equally useful for seeking NE for any non-smooth utility game when the game possesses what we term “amicable” supergradients. After this, we provide ultimate boundedness guarantees to an arbitrarily small region around the DRoNE set. Moreover, we report a heuristic to tune the algorithm parameters to attain aforementioned arbitrarily small region to which the algorithm converges to. We then exploit input-to-state stability properties of dynamic average consensus algorithms and make ISBRAG distributed by allowing agents to estimate others’ strategies over a communication network. Finally, to tackle shared observations, we adapt the tractable reformulation of the distributionally robust optimization into a distributed optimization problem. This allows agents to compute their supergradients in a fully distributed manner. As such, d-ISBRAG for shared observations does not require full information from all agents and can handle the effect of partial observations shared among the group of agents. We present simulations that support our results.

The rest of the chapter is organized as follows. In Section 3.4, we formalize the distributionally robust game and state the problems we are interested in addressing. Then in Section 3.5, we relate the Nash equilibria of the distributionally robust game with the underlying stochastic game and provide conditions under which the aforementioned equilibria exist. In Section 3.6, we provide an algorithm that seeks out the Nash equilibria of the distributionally robust game. This algorithm requires central information from all agents, so in the next two sections we provide ways to make the algorithm distributed. Specifically, in Section 3.7 we provide a distributed algorithm under the simplifying assumption that the agents are able to compute their own strategy updates provided they have an estimate of other agents’ strategies. In Section 3.8 we provide a distributed algorithm under the full generality where agents have to estimate their own strategy updates and other agents’ strategies over a communication network. Finally, we provide simulations

demonstrating our results in Section 3.9 and conclude the chapter in Section 3.10.

3.4 Problem Formulation

We recall that a *strategic form game* [10] is a tuple

$$\mathcal{G} := \langle \mathcal{A}, \{\mathcal{S}_i\}_{i \in \mathcal{A}}, \{U_i\}_{i \in \mathcal{A}} \rangle,$$

consisting of a set of *players* (or *agents*) $\mathcal{A} := \{1, \dots, n\}$; a set of *strategies* $\mathbf{s}_i \in \mathcal{S}_i \subseteq \mathbb{R}^{n_i}$ (with components $s_{i,l}$) available to each $i \in \mathcal{A}$; and a set of *utility functions* $U_i : \times_{j \in \mathcal{A}} \mathcal{S}_j \rightarrow \mathbb{R}$ over the strategy profiles of all the agents. We assume that each agent is selfish and interested in maximizing its own utility. In what follows, $\mathbf{s}_{-i}$ denotes the strategy profile of all players other than $i \in \mathcal{A}$. For the sake of brevity, we denote $\mathcal{S} := \times_{i \in \mathcal{A}} \mathcal{S}_i$ the set of combined strategies, and $\forall i \in \mathcal{A}$, $\mathcal{S}_{-i} := \times_{j \in \mathcal{A} \setminus \{i\}} \mathcal{S}_j$ the set of strategy profiles of all players other than $i \in \mathcal{A}$. Moreover, $\mathbf{s} := [\mathbf{s}_1^\top, \dots, \mathbf{s}_n^\top]^\top$ denotes the stacked vector of all strategies. A well-known concept that captures the equilibrium behavior of such a group of agents trying to selfishly maximize their own utility is the Nash equilibrium (NE). Next, we formally state the definition of an η -NE of a strategic form game.

Definition 3.4.1. (*η -Nash equilibrium*). A strategy $(\mathbf{s}_i^*, \mathbf{s}_{-i}^*)$ is said to be an *η -Nash equilibrium* (with $\eta \geq 0$) of $\mathcal{G}$ iff

$$U_i(\mathbf{s}_i^*, \mathbf{s}_{-i}^*) + \eta \geq U_i(\mathbf{s}_i, \mathbf{s}_{-i}^*), \forall \mathbf{s}_i \in \mathcal{S}_i, \forall i \in \mathcal{A}.$$

The set of η -Nash equilibria (η -NE) of $\mathcal{G}$ is given by $\mathcal{NE}_\eta(\mathcal{G})$. If $\eta = 0$, this set becomes the set of Nash equilibria (NE), $\mathcal{NE}(\mathcal{G})$, of the game. •

In this chapter, we focus on a case where every agent's utility is affected by some random variable, which can be caused by noisy measurements or the environment. Specifically, suppose agent $i \in \mathcal{A}$ obtains a utility of $U_i(\mathbf{s}_i, \mathbf{s}_{-i}; \xi)$, where $\xi \in \Xi \subseteq \mathbb{R}^m$ is

a random variable. If the distribution of ξ were known (say $\xi \sim \mathbb{P}^*$), each agent could maximize their expected utility given as

$$U_i^S(\mathbf{s}_i, \mathbf{s}_{-i}) := \mathbb{E}_{\xi \sim \mathbb{P}^*} [U_i(\mathbf{s}_i, \mathbf{s}_{-i}; \xi)] . \quad (3.1)$$

Here, we assume that $\mathbb{P}^* \in \mathcal{M}(\Xi)$ with $\Xi \subset \mathbb{R}^m$ a compact subset. Moreover we assume that $\mathbb{P}^*$ is absolutely continuous w.r.t the Lebesgue measure and $\mathbb{P}^*$ is light-tailed. This results into the *stochastic game*

$$\mathcal{G}_S := \langle \mathcal{A}, \{\mathcal{S}_i\}_{i \in \mathcal{A}}, \{U_i^S\}_{i \in \mathcal{A}} \rangle. \quad (3.2)$$

We make the following assumptions throughout the chapter regarding the utilities and the strategies of agents.

Standing Assumption 3.4.2. The components of $\mathcal{G}_S$ in (3.2) satisfy the following properties for each $i \in \mathcal{A}$;

1. $\mathcal{S}_i \subseteq \mathbb{R}^{n_i}$ is non-empty, convex, and compact with diameter $D_i \in \mathbb{R}_{\geq 0}$;
2. U_i is continuous and $U_i(\mathbf{s}_i, \mathbf{s}_{-i}; \xi)$ is concave in $\mathbf{s}_i$, $\forall \mathbf{s}_{-i}, \xi$; and
3. $U_i(\mathbf{s}_i, \mathbf{s}_{-i}; \xi)$ is L_i -Lipschitz in ξ , $\forall \mathbf{s}_i, \mathbf{s}_{-i}$. •

In particular, we will assume that the original distribution $\mathbb{P}^*$ is unknown to the agents, yet they will be able to sample from it. However, such samples may correspond to partial measurements, as we state next.

Assumption 3.4.3. (Shared i.i.d. samples). *Each $i \in \mathcal{A}$ has access to $\{\mathbf{h}_i(\widehat{\xi}^{(1)}), \dots, \mathbf{h}_i(\widehat{\xi}^{(N)})\}$, a set of observations from $\{\widehat{\xi}^{(1)}, \dots, \widehat{\xi}^{(N)}\}$ i.i.d. samples of $\mathbb{P}^*$.* •

The previous assumption applies to agents endowed with heterogeneous sensors

and/or computation capabilities. For the sake of brevity, we define

$$\mathbf{h} := [\mathbf{h}_1^\top, \dots, \mathbf{h}_n^\top]^\top,$$

as the stacked vector of functions that take in the samples as an input and return the stacked observations of all the agents as an output; *i.e.* $\mathbf{h}(\hat{\xi}^{(k)}) := [\mathbf{h}_1(\hat{\xi}^{(k)})^\top, \dots, \mathbf{h}_n(\hat{\xi}^{(k)})^\top]^\top$, $\forall k \in \{1, \dots, N\}$.

Further, we assume the following on the function $\mathbf{h}$.

Assumption 3.4.4. (Observability). *The observation function $\mathbf{h} : \mathbb{R}^m \rightarrow \mathbb{R}^m$ is invertible and measurable.* •

The previous assumption requires that there exists a sufficiently large number of agents so that $\mathbf{h}$ is invertible. This requirement could be mitigated by means of an observer when agents are able to obtain a continuous stream of samples by themselves [40] or from a dynamical system [41]. We leave these considerations for future work.

To deal with the unknown distribution, an ambiguity set centered at the average sample distribution

$$\hat{\mathbb{P}}_{\mathbf{h}}^N := \frac{1}{N} \sum_{k=1}^N \delta_{\mathbf{h}(\hat{\xi}^{(k)})},$$

is constructed. Recall that $\delta_{\mathbf{x}}$ represents the Dirac delta function at $\mathbf{x}$. The $\mathbf{h}$ in the subscript of $\hat{\mathbb{P}}_{\mathbf{h}}^N$ is used to distinguish that this sample average distribution relies on observations of the samples. Then, a distributionally robust utility for agent $i \in \mathcal{A}$ is defined as

$$\bar{U}_i^{\text{DR}}(\mathbf{s}_i, \mathbf{s}_{-i}) := \inf_{\mathbb{Q} \in \mathcal{B}_{\varepsilon_i}(\hat{\mathbb{P}}_{\mathbf{h}}^N)} \mathbb{E}_{\xi \sim \mathbb{Q}} [U_i(\mathbf{s}_i, \mathbf{s}_{-i}; \xi)], \quad (3.3)$$

where, recall, $\mathcal{B}_{\varepsilon_i}(\hat{\mathbb{P}}_{\mathbf{h}}^N)$ is the Wasserstein ball of radius ε_i centered at $\hat{\mathbb{P}}_{\mathbf{h}}^N$. Even though

$\widehat{\mathbb{P}}_{\mathbf{h}}^N$ relies on global information and is common among the agents, $\varepsilon_i \in \mathbb{R}_{\geq 0}$ is a parameter that agent $i \in \mathcal{A}$ can independently choose. The above utility description allows us to formulate the *distributionally robust game*,

$$\overline{\mathcal{G}}_{\text{DR}} := \left\langle \mathcal{A}, \{\mathcal{S}_i\}_{i \in \mathcal{A}}, \{\overline{U}_i^{\text{DR}}\}_{i \in \mathcal{A}} \right\rangle. \quad (3.4)$$

We refer to the set of NE of $\mathcal{G}_{\text{DR}}$, *i.e.* $\mathcal{NE}(\mathcal{G}_{\text{DR}})$, as *distributionally robust Nash equilibria* or *DRoNE*.

When uncertainty is individual:

A special case of the previous scenario occurs when the random variable $\xi \sim \mathbb{P}^*$ affects each agent's utility in an independent manner. We state this precisely next.

Assumption 3.4.5. (Individual uncertainty). *The random variable $\xi \sim \mathbb{P}^*$ satisfies, for each agent $i \in \mathcal{A}$:*

1. $\xi \in \Xi = \Xi_1 \times \cdots \times \Xi_n \subseteq \mathbb{R}^m$ can be separated as $\xi = [\xi_1^\top, \dots, \xi_n^\top]^\top$, with $\xi_i \in \Xi_i \subseteq \mathbb{R}^{m_i}$;
2. $\mathbb{P}^* = \mathbb{P}_1^* \otimes \cdots \otimes \mathbb{P}_n^*$, with $\xi_i \sim \mathbb{P}_i^*$;
3. $\mathbf{h}_i$ is such that $\forall \xi \in \Xi$, $\mathbf{h}_i(\xi) = \xi_i \in \Xi_i \subseteq \mathbb{R}^{m_i}$; and
4. $\forall \mathbf{s} \in \mathcal{S}$, $U_i(\mathbf{s}; \xi) = U_i(\mathbf{s}; \xi_i)$, $\forall \xi \in \Xi$. •

This makes it so that each agent's uncertainty does not affect another agent in any way. When this is the case, with a slight abuse of notation, we assume the following.

Assumption 3.4.6. (Individual i.i.d. samples). *For each $i \in \mathcal{A}$, $\{\widehat{\xi}_i^{(1)}, \dots, \widehat{\xi}_i^{(N_i)}\}$ are i.i.d. samples from $\mathbb{P}_i^*$.* •

Thus, each agent is capable of individually measuring the uncertainty it is subject to. Moreover, we allow different $N_i \in \mathbb{Z}_{\geq 0}$ for each agent $i \in \mathcal{A}$, which means that agents can now hold different numbers of samples from the original distribution.

In this special case, each agent can create an ambiguity set by considering the sample average distribution

$$\widehat{\mathbb{P}}_i^{N_i} := \frac{1}{N_i} \sum_{k=1}^{N_i} \delta_{\xi_i^{(k)}}.$$

Then, the distributionally-robust utility for agent $i \in \mathcal{A}$ can be modified as

$$U_i^{\text{DR}}(\mathbf{s}_i, \mathbf{s}_{-i}) := \inf_{\mathbb{Q} \in \mathcal{B}_{\varepsilon_i}(\widehat{\mathbb{P}}_i^{N_i})} \mathbb{E}_{\xi \sim \mathbb{Q}} [U_i(\mathbf{s}_i, \mathbf{s}_{-i}; \xi)], \quad (3.5)$$

where, $\mathcal{B}_{\varepsilon_i}(\widehat{\mathbb{P}}_i^{N_i})$ is the Wasserstein ball of radius ε_i around $\widehat{\mathbb{P}}_i^{N_i}$. The above utility description allows us to reformulate the distributionally robust game as

$$\mathcal{G}_{\text{DR}} := \langle \mathcal{A}, \{\mathcal{S}_i\}_{i \in \mathcal{A}}, \{U_i^{\text{DR}}\}_{i \in \mathcal{A}} \rangle. \quad (3.6)$$

Communication protocol:

In order to update its own strategy, an agent will need information about the strategies and possibly other data of other agents. In this chapter, we assume that the agents obtain information by communicating with others over a static communication network $\mathcal{G} := (\mathcal{A}, \mathcal{E})$ with vertex set $\mathcal{A}$. The arc set $\mathcal{E}$ defines the connections between agents, with $(i, j) \in \mathcal{E}$ if and only if $i \in \mathcal{A}$ can send information to $j \in \mathcal{A}$. We make the following assumption regarding the network.

Assumption 3.4.7. (Connectivity). *The communication graph $\mathcal{G} = (\mathcal{A}, \mathcal{E})$ is strongly connected.* •

This allows information from each agent to reach every other agent in the group. Note that we fix the communication graph to be static to make the notation simpler; however, the results in the sequel can be extended to time-varying graphs with periodic strong connectivity.

Now, we are ready to state the goals of this work.

Problem 3.4.8. Given the aforementioned setup, under the Standing Assumption 3.4.2, and Assumption 3.4.3, on shared i.i.d. samples, Assumption 3.4.4, on proper observations, and 3.4.7, on connectivity; provide

1. conditions under which a DRoNE exists;
2. relations between DRoNE and NE of the stochastic game; *i.e.* $\mathcal{G}_S$ and $\overline{\mathcal{G}}_{DR}$; and
3. a distributed algorithm that converges to a DRoNE.

Further, discuss these solutions under the special case given by Assumption 3.4.5, on individual uncertainty, and Assumption 3.4.6, on individual i.i.d. samples. •

We would like to point out that providing a solution for Problem 3.4.8: (3), under Assumption 3.4.3 is difficult. Due to this, we provide a solution for Problem 3.4.8: (1) and (2) in the next section, then address Problem 3.4.8: (3) for the special case given by Assumptions 3.4.5 and 3.4.6, which we adapt later to the more general case in Section 3.8.

3.5 On the Nash Equilibria of the Games

In this section, we deal with Problem 3.4.8: (1), (2), and provide qualitative properties of the Nash equilibria of the two games; $\mathcal{G}_S$ and $\overline{\mathcal{G}}_{DR}$. In particular, we provide conditions under which the sets of Nash equilibria are non-empty, and then we relate the Nash equilibria sets.

First, we show that the distributionally robust utility enjoys helpful properties under certain assumptions.

Lemma 3.5.1. (Concavity and continuity of distributionally robust utility). *Suppose the Standing Assumptions 3.4.2: (1) and 3.4.2: (2) hold. Then, $\overline{U}_i^{DR}(\cdot, \mathbf{s}_{-i})$ is concave $\forall \mathbf{s}_{-i} \in \mathcal{S}_{-i}$, for each $i \in \mathcal{A}$. Moreover, for each $i \in \mathcal{A}$, $\overline{U}_i^{DR}$ is continuous.*

Proof. First, we show concavity. Fix an $i \in \mathcal{A}$ and $\mathbf{s}_{-i} \in \mathcal{S}_{-i}$. From the hypothesis, the function $g_{\mathbb{Q}} : \mathbb{R}^{n_i} \rightarrow \mathbb{R}$,

$$g_{\mathbb{Q}}(\mathbf{x}) := \mathbb{E}_{\xi \sim \mathbb{Q}}[U_i(\mathbf{x}, \mathbf{s}_{-i}, \xi)]$$

is concave. From the concavity of $g_{\mathbb{Q}}$ and the defining property of infimum,

$$\begin{aligned} \bar{U}_i^{\text{DR}}(\lambda \mathbf{x} + [1 - \lambda] \mathbf{y}, \mathbf{s}_{-i}) &:= \inf_{\mathbb{Q} \in \mathcal{B}_{\varepsilon_i}(\hat{\mathbb{P}}_{\mathbf{h}}^N)} g_{\mathbb{Q}}(\lambda \mathbf{x} + [1 - \lambda] \mathbf{y}) \\ &\geq \lambda \inf_{\mathbb{Q} \in \mathcal{B}_{\varepsilon_i}(\hat{\mathbb{P}}_{\mathbf{h}}^N)} g_{\mathbb{Q}}(\mathbf{x}) + [1 - \lambda] \inf_{\mathbb{Q} \in \mathcal{B}_{\varepsilon_i}(\hat{\mathbb{P}}_{\mathbf{h}}^N)} g_{\mathbb{Q}}(\mathbf{y}) \\ &=: \lambda \bar{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i}) + [1 - \lambda] \bar{U}_i^{\text{DR}}(\mathbf{y}, \mathbf{s}_{-i}), \end{aligned}$$

for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^m$, and $\lambda \in [0, 1]$. Hence, the first claim.

Next, we prove continuity. To begin, notice that $\mathcal{S}, \Xi$ are compact sets and U_i is continuous on $\mathcal{S} \times \Xi$. Hence, U_i is bounded on $\mathcal{S} \times \Xi$. Then, applying the dominated convergence theorem [51], for any $\mathbb{Q} \in \mathcal{M}(\Xi)$, $\mathbb{E}_{\xi \sim \mathbb{Q}}[U_i(\mathbf{s}; \xi)]$ is continuous in $\mathbf{s}$. Now, $\mathcal{M}(\Xi)$ is a subset of a reflexive normed vector space. Moreover, $\mathcal{B}_{\varepsilon_i}(\hat{\mathbb{P}}_{\mathbf{h}}^N) \subseteq \mathcal{M}(\Xi)$ is non-empty, bounded, closed, and convex (by construction). Then, by Tonelli's theorem [52], $\inf_{\mathbb{Q} \in \mathcal{B}_{\varepsilon_i}(\hat{\mathbb{P}}_{\mathbf{h}}^N)} \mathbb{E}_{\xi \sim \mathbb{Q}}[U_i(\mathbf{s}_i, \mathbf{s}_{-i}; \xi)]$ attains its minimum; *i.e.* $\forall \mathbf{s} \in \mathcal{S}, \exists \mathbb{Q}_{\mathbf{s}} \in \mathcal{M}(\Xi)$, such that $\bar{U}_i^{\text{DR}}(\mathbf{s}) = \mathbb{E}_{\xi \sim \mathbb{Q}_{\mathbf{s}}}[U_i(\mathbf{s}; \xi)]$. We have used the subscript to show that this $\mathbb{Q}_{\mathbf{s}}$ can depend on $\mathbf{s}$.

With this setup, we prove continuity using contradiction. Suppose $\bar{U}_i^{\text{DR}}$ is not continuous at $\hat{\mathbf{s}} \in \mathcal{S}$. Then, $\exists a_1 > 0$, such that $\forall a_2 > 0, \exists \mathbf{s} \in \mathcal{S}$ with $\|\hat{\mathbf{s}} - \mathbf{s}\| < a_2$ and $|\bar{U}_i^{\text{DR}}(\hat{\mathbf{s}}) - \bar{U}_i^{\text{DR}}(\mathbf{s})| > a_1$. Assume w.l.o.g. that $\bar{U}_i^{\text{DR}}(\hat{\mathbf{s}}) > \bar{U}_i^{\text{DR}}(\mathbf{s}) + a_1$. Then, it is easy to see from the definition in (3.5) and the prior discussion, that this would imply that $\exists \mathbb{Q}_{\mathbf{s}}, \mathbb{Q}_{\hat{\mathbf{s}}} \in \mathcal{B}_{\varepsilon_i}(\hat{\mathbb{P}}_{\mathbf{h}}^N)$, such that $\mathbb{E}_{\xi \sim \mathbb{Q}_{\hat{\mathbf{s}}}}[U_i(\hat{\mathbf{s}}; \xi)] > \mathbb{E}_{\xi \sim \mathbb{Q}_{\mathbf{s}}}[U_i(\mathbf{s}; \xi)] - a_1$. Now as $\mathbb{Q}_{\hat{\mathbf{s}}} \in \arg\min_{\mathbb{Q} \in \mathcal{B}_{\varepsilon_i}(\hat{\mathbb{P}}_{\mathbf{h}}^N)} \mathbb{E}_{\xi \sim \mathbb{Q}}[U_i(\hat{\mathbf{s}}; \xi)]$, $\mathbb{E}_{\xi \sim \mathbb{Q}_{\mathbf{s}}}[U_i(\hat{\mathbf{s}}; \xi)] \geq \mathbb{E}_{\xi \sim \mathbb{Q}_{\hat{\mathbf{s}}}}[U_i(\hat{\mathbf{s}}; \xi)] > \mathbb{E}_{\xi \sim \mathbb{Q}_{\mathbf{s}}}[U_i(\mathbf{s}; \xi)] - a_1$.

This then contradicts the continuity of $\mathbb{E}_{\xi \sim \mathbb{Q}_s} [U_i(\cdot; \xi)]$. $\blacksquare$

Using this result, we can immediately show that the distributionally robust game possesses a Nash equilibrium.

Proposition 3.5.2. (Existence of a DRoNE). *Suppose that the Standing Assumption 3.4.2 holds. Then, $\mathcal{NE}(\overline{\mathcal{G}}_{\text{DR}}) \neq \emptyset$. Further, $\mathcal{NE}(\mathcal{G}_S) \neq \emptyset$.*

Proof. We provide the proof for $\mathcal{NE}(\overline{\mathcal{G}}_{\text{DR}}) \neq \emptyset$, as the proof of $\mathcal{NE}(\mathcal{G}_S) \neq \emptyset$ is analogous. Consider an arbitrary but fixed $i \in \mathcal{A}$. From Lemma 3.5.1, $\overline{U}_i^{\text{DR}}(\cdot, \mathbf{s}_{-i})$ is concave $\forall \mathbf{s}_{-i} \in \mathcal{S}_{-i}$ and $\overline{U}_i^{\text{DR}}$ is continuous. Moreover, $\mathcal{S}_i$ is a compact set, $\forall i \in \mathcal{A}$, which makes $\mathcal{S} = \times_{i \in \mathcal{A}} \mathcal{S}_i$ also compact. Thus, $\overline{U}_i^{\text{DR}}$ is continuous, defined on a compact set, and concave in its first argument $\forall i \in \mathcal{A}$. The existence claim follows from the application of Kakutani's fixed point theorem [10]. $\blacksquare$

Note that the Standing Assumptions 3.4.2: (1) and (2) are standard in showing the existence of Nash equilibria in games [10]. As such, Proposition 3.5.2 extends this to distributionally robust games in a seamless way.

Now that we have shown that the set of DRoNE's is non-empty, we are interested in relating the DRoNE set with the Nash equilibria of the stochastic game. In order to do this, we first need to extend the result in Theorem 2.2.1 to account for samples based on observations. For the sake of brevity, we define

$$\widehat{\mathbb{P}}_{\mathbf{h}, \Sigma}^N := \frac{1}{N} \sum_{k=1}^N \mathbf{N}_{\mathbf{h}(\widehat{\xi}^{(k)})}^{\Sigma}, \quad (3.7)$$

as the density that smoothenes out sample point masses using a Gaussian of covariance Σ . Similarly, denote $\widehat{\mathbb{P}}_{\Sigma}^N := \frac{1}{N} \sum_{k=1}^N \mathbf{N}_{\widehat{\xi}^{(k)}}^{\Sigma}$. We restrict our observation functions $\mathbf{h}$ to be in the following class.

Definition 3.5.3. (*Inferable observations*). Consider any matrix sequence $\{\Sigma_l\}_{l \in \mathbb{Z}_{>0}}$ that satisfies $\Sigma_l \rightarrow \mathbf{O}$, as $l \rightarrow \infty$. For each $l \in \mathbb{Z}_{>0}$ define the set of mappings $\mathcal{T}_l := \{T \mid T_{\#} \widehat{\mathbb{P}}_{\Sigma_l}^N =$

$\mathbb{P}^*\}$. Suppose $\mathbf{h}$ is such that $\forall T \in \mathcal{T}_l$,

$$\int_{\mathbb{R}^m} \|\xi - \mathbf{h}(\xi)\|_1 \widehat{\mathbb{P}}_{\Sigma_l}^N(d\xi) \leq C_l \int_{\mathbb{R}^m} \|\xi - T(\xi)\|_1 \widehat{\mathbb{P}}_{\Sigma_l}^N(d\xi).$$

We call $\mathbf{h}$ *inferable* with *inflation* C , if $\lim_{l \rightarrow \infty} C_l = C$. •

The property of Definition 3.5.3 allows us to inflate the Wasserstein ball in Theorem 2.2.1, and account for the indirect observation of the samples. This leads to the following result.

Theorem 3.5.4. (Distributionally robust bound for observations). *Suppose Assumptions 3.4.3, and 3.4.4 hold; and let $\mathbf{h}$ be inferable with inflation C . Fix $\theta \in (0, 1)$. Then, $\Pr \{ \mathbb{P}^* \in \mathcal{B}_{[C+1]\varepsilon}(\widehat{\mathbb{P}}_{\mathbf{h}}^N) \} \geq 1 - \theta$, where ε is as in (2.4).*

Proof. We use the Monge form of the optimal transport problem in (2.2) to prove this claim. Recall that $\widehat{\mathbb{P}}^N = \frac{1}{N} \sum_{k=1}^N \delta_{\widehat{\xi}^{(k)}}$ and $\widehat{\mathbb{P}}_{\Sigma}^N = \frac{1}{N} \sum_{k=1}^N \mathbf{N}_{\widehat{\xi}^{(k)}}^{\Sigma}$. Suppose $\mathcal{F}$ is the Borel σ -algebra on $\mathbb{R}^m$. Then, notice that for the smoothened density,

$$\mathbf{h}_{\#} \widehat{\mathbb{P}}_{\Sigma}^N = \mathbf{h}_{\#} \left[\frac{1}{N} \sum_{k=1}^N \mathbf{N}_{\widehat{\xi}^{(k)}}^{\Sigma} \right] = \frac{1}{N} \sum_{k=1}^N \mathbf{h}_{\#} \mathbf{N}_{\widehat{\xi}^{(k)}}^{\Sigma} = \widehat{\mathbb{P}}_{\mathbf{h}, \Sigma}^N.$$

Take any sequence $\{\Sigma_l\}_{l \in \mathbb{Z}_{>0}}$ as in Definition 3.5.3. Then, for any $l \in \mathbb{Z}_{>0}$, and any set $\mathcal{F} \in \mathcal{F}$ we have that

$$\mathbf{h}_{\#} \widehat{\mathbb{P}}_{\Sigma_l}^N(\mathcal{F}) = \frac{1}{N} \sum_{k=1}^N \mathbf{h}_{\#} \mathbf{N}_{\widehat{\xi}^{(k)}}^{\Sigma_l}(\mathcal{F}) = \frac{1}{N} \sum_{k=1}^N \Pr \{ \mathbf{h}(X_k) \in \mathcal{F} \},$$

with $X_k \sim \mathbf{N}_{\widehat{\xi}^{(k)}}^{\Sigma_l}$, $\forall k \in \{1, \dots, N\}$. Thus, $\mathbf{h}_{\#} \widehat{\mathbb{P}}_{\Sigma_l}^N = \widehat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N$.

Now, consider a particular $l \in \mathbb{Z}_{>0}$. Recall that $\mathcal{T}_l := \{T \mid T_{\#} \widehat{\mathbb{P}}_{\Sigma_l}^N = \mathbb{P}^*\}$ and define the set of mappings $\widetilde{\mathcal{T}}_l := \{\widetilde{T} \mid \widetilde{T}_{\#} \widehat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N = \mathbb{P}^*\}$. Since $\mathbf{h}$ is invertible, for every T such that $T_{\#} \widehat{\mathbb{P}}_{\Sigma_l}^N = \mathbb{P}^*$ (i.e. $T \in \mathcal{T}_l$) it holds that $\widetilde{T}_{\#} \widehat{\mathbb{P}}_{\Sigma_l}^N = T_{\#}(\mathbf{h}^{-1})_{\#} \widehat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N = T_{\#} \widehat{\mathbb{P}}_{\Sigma_l}^N = \mathbb{P}^*$, for $\widetilde{T} = T \circ (\mathbf{h}^{-1})$ (i.e. $\widetilde{T} \in \widetilde{\mathcal{T}}_l$). Similarly, $\forall \widetilde{T} \in \widetilde{\mathcal{T}}_l$, we have $T \in \mathcal{T}_l$.

Next, define the following operators $F : \mathcal{T}_l \rightarrow \tilde{\mathcal{T}}_l$, with $F(T) = T \circ (\mathbf{h}^{-1})$ and $G : \tilde{\mathcal{T}}_l \rightarrow \mathcal{T}_l$, with $G(\tilde{T}) = \tilde{T} \circ \mathbf{h}$. It is easy to see that F and G are inverse operators. With this,

$$\begin{aligned} d_{\mathbf{W}}(\hat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N, \mathbb{P}^*) &= \inf_{\tilde{T} \in \tilde{\mathcal{T}}_l} \int_{\mathbb{R}^m} \|\xi - \tilde{T}(\xi)\|_1 \hat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N(d\xi) \\ &= \inf_{T \in \mathcal{T}_l} \int_{\mathbb{R}^m} \|\xi - (T \circ (\mathbf{h}^{-1}))(\xi)\|_1 \hat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N(d\xi). \end{aligned}$$

Since $\mathbf{h}_\# \hat{\mathbb{P}}_{\Sigma_l}^N(d\xi) = \hat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N(d\xi)$, using the change of variables for pushforward measures, we have,

$$\begin{aligned} &\int_{\mathbb{R}^m} \|\xi - (T \circ (\mathbf{h}^{-1}))(\xi)\|_1 \hat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N(d\xi) \\ &= \int_{\mathbb{R}^m} \|\mathbf{h}(\xi) - (T \circ (\mathbf{h}^{-1}))(\mathbf{h}(\xi))\|_1 \hat{\mathbb{P}}_{\Sigma_l}^N(d\xi) \\ &= \int_{\mathbb{R}^m} \|\mathbf{h}(\xi) - T(\xi)\|_1 \hat{\mathbb{P}}_{\Sigma_l}^N(d\xi) \\ &\leq \int_{\mathbb{R}^m} \|\mathbf{h}(\xi) - \xi\|_1 \hat{\mathbb{P}}_{\Sigma_l}^N(d\xi) + \int_{\mathbb{R}^m} \|\xi - T(\xi)\|_1 \hat{\mathbb{P}}_{\Sigma_l}^N(d\xi). \end{aligned}$$

The last inequality comes from triangle inequality and linearity of integrals. Then, because of Definition 3.5.3,

$$\begin{aligned} d_{\mathbf{W}}(\hat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N, \mathbb{P}^*) &= \inf_{T \in \mathcal{T}} \int_{\mathbb{R}^m} \|\xi - (T \circ (\mathbf{h}^{-1}))(\xi)\|_1 \hat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N(d\xi) \\ &\leq \inf_{T \in \mathcal{T}} \left[\int_{\mathbb{R}^m} \|\mathbf{h}(\xi) - \xi\|_1 \hat{\mathbb{P}}_{\Sigma_l}^N(d\xi) + \int_{\mathbb{R}^m} \|\xi - T(\xi)\|_1 \hat{\mathbb{P}}_{\Sigma_l}^N(d\xi) \right] \\ &\leq [C_l + 1] \inf_{T \in \mathcal{T}} \int_{\mathbb{R}^m} \|\xi - T(\xi)\|_1 \hat{\mathbb{P}}_{\Sigma_l}^N(d\xi) \\ &= [C_l + 1] d_{\mathbf{W}}(\hat{\mathbb{P}}_{\Sigma_l}^N, \mathbb{P}^*). \end{aligned}$$

Now, as $l \rightarrow \infty$, $d_{\mathbf{W}}(\hat{\mathbb{P}}_{\mathbf{h}, \Sigma_l}^N, \mathbb{P}^*) \rightarrow d_{\mathbf{W}}(\hat{\mathbb{P}}_{\mathbf{h}}^N, \mathbb{P}^*)$ and $d_{\mathbf{W}}(\hat{\mathbb{P}}_{\Sigma_l}^N, \mathbb{P}^*) \rightarrow d_{\mathbf{W}}(\hat{\mathbb{P}}^N, \mathbb{P}^*)$. Thus, since

$C_l \rightarrow C$ as $l \rightarrow \infty$, taking the limit as $l \rightarrow \infty$ of the previous inequality, we get

$$d_W(\hat{\mathbb{P}}_{\mathbf{h}}^N, \mathbb{P}^\star) \leq [C + 1] d_W(\hat{\mathbb{P}}^N, \mathbb{P}^\star).$$

Then, by applying Theorem 2.2.1, the proof is complete. ■

Remark 3.5.5. (*On sufficient conditions for Theorem 3.5.4*). The uncertainty bound in Theorem 3.5.4 relies on the “inflation” of the original Wasserstein ball in order to guarantee that the sample average using the observations lies in this larger ball with similar probability. The existence of this inflation, $C + 1$, depends on the observation function $\mathbf{h}$ through Assumption 3.4.4 and Definition 3.5.3. Notice, in fact, that $\mathbf{h}$ can itself be viewed as a transport map. Thus, the assumption roughly captures that, when $\mathbf{h}$ is inferable with inflation C , the effect of the transport defined by $\mathbf{h}$ is on average proportional to the effect of transport under the optimal transport map.

Moreover, notice that when there is no observation function involved, *i.e.* when $\mathbf{h}(\xi) = \xi$, then this result is equivalent to the original concentration of measure result in Theorem 2.2.1. This is because, under this condition,

$$\int_{\mathbb{R}^m} \|\xi - \mathbf{h}(\xi)\|_1 \hat{\mathbb{P}}_{\Sigma_l}^N(d\xi) = \int_{\mathbb{R}^m} \|\xi - \xi\|_1 \hat{\mathbb{P}}_{\Sigma_l}^N(d\xi) = 0,$$

and hence, the inflation parameter is in fact $C = 0$. ●

Thus, using the uncertainty quantification provided by observations we provide the following relation between the sets of Nash equilibria.

Theorem 3.5.6. (Every DRoNE is an η -NE of $\mathcal{G}_S$ with certain probability). *Suppose Assumption 3.4.3 holds and that $\mathbf{h}$ is inferable with inflation C . Suppose that, for all $i \in \mathcal{A}$, $\theta_i \in (0, 1)$ and $\varepsilon_i(N, \theta_i)$ are chosen according to (2.4). Assume that the strategy $(\mathbf{s}_i^\star, \mathbf{s}_{-i}^\star) \in \mathcal{NE}(\overline{\mathcal{G}}_{\text{DR}})$, where the inf in (3.3) is taken over closed Wasserstein balls of*

radii $\bar{\varepsilon}_i = [C + 1]\varepsilon_i$, i.e. $\mathbb{Q} \in \mathcal{B}_{\bar{\varepsilon}_i}(\widehat{\mathbb{P}}_{\mathbf{h}}^N)$, $\forall i \in \mathcal{A}$. Then, with probability at least $(1 - \theta)^n$, $(\mathbf{s}_i^*, \mathbf{s}_{-i}^*) \in \mathcal{NE}_\eta(\mathcal{G}_S)$, where $\eta = 2[C + 1] \max_{i \in \mathcal{A}} \varepsilon_i L_i$, and $\theta = \max_{i \in \mathcal{A}} \theta_i$.

Proof. Consider an arbitrary but fixed $i \in \mathcal{A}$. Since $\bar{\varepsilon}_i$ is chosen according to (2.4), then by Theorem 3.5.4, $\mathbb{P}^* \in \mathcal{B}_{\bar{\varepsilon}_i}(\widehat{\mathbb{P}}_{\mathbf{h}}^N)$ with probability at least $1 - \theta_i$. Thus,

$$\bar{U}_i^{\text{DR}}(\mathbf{s}_i^*, \mathbf{s}_{-i}^*) \leq U_i^S(\mathbf{s}_i^*, \mathbf{s}_{-i}^*), \quad (3.8)$$

with probability at least $1 - \theta_i$. Now, from the hypothesis we have that $(\mathbf{s}_i^*, \mathbf{s}_{-i}^*) \in \mathcal{NE}(\bar{\mathcal{G}}_{\text{DR}})$.

This gives us,

$$\bar{U}_i^{\text{DR}}(\mathbf{s}_i^*, \mathbf{s}_{-i}^*) \geq \bar{U}_i^{\text{DR}}(\mathbf{s}_i, \mathbf{s}_{-i}^*), \forall \mathbf{s}_i \in \mathcal{S}_i. \quad (3.9)$$

Next, consider an arbitrary but fixed $\mathbf{s}_i \in \mathcal{S}_i$ and let $\mathbb{Q} \in \mathcal{B}_{\bar{\varepsilon}_i}(\widehat{\mathbb{P}}_i^{N_i})$. Thus, again from Theorem 3.5.4, $d_W(\mathbb{P}_i^*, \mathbb{Q}) \leq 2\bar{\varepsilon}_i = 2C\varepsilon_i$, with probability at least $1 - \theta_i$. Thus, using the Kantorovich–Rubinstein description of the Wasserstein distance in (2.3), we have,

$$\begin{aligned} & [1/L_i] \left[U_i^S(\mathbf{s}_i, \mathbf{s}_{-i}^*) - \mathbb{E}_{\xi \sim \mathbb{Q}} [U_i(\mathbf{s}_i, \mathbf{s}_{-i}^*; \xi)] \right] \\ &= \frac{1}{L_i} \left[\int_{\Xi} U_i(\mathbf{s}_i, \mathbf{s}_{-i}^*; \xi) \mathbb{P}_i^*(d\xi) - \int_{\Xi} U_i(\mathbf{s}_i, \mathbf{s}_{-i}^*; \xi) \mathbb{Q}(d\xi) \right] \\ &\leq 2[C + 1]\varepsilon_i. \end{aligned} \quad (3.10)$$

with probability at least $1 - \theta_i$. Here, the first equality comes from the definitions of the utilities and the last inequality comes from (2.3), Standing Assumption 3.4.2: (3) and the

prior discussion. Thus, using (3.10),

$$\begin{aligned}
& U_i^S(\mathbf{s}_i, \mathbf{s}_{-i}^*) - \overline{U}_i^{\text{DR}}(\mathbf{s}_i, \mathbf{s}_{-i}^*) \\
&= U_i^S(\mathbf{s}_i, \mathbf{s}_{-i}^*) - \inf_{\mathbb{Q} \in \mathcal{B}_{\varepsilon_i}(\widehat{\mathbb{P}}_i^{N_i})} \mathbb{E}_{\xi \sim \mathbb{Q}} [U_i(\mathbf{s}_i, \mathbf{s}_{-i}^*; \xi)] \\
&= \sup_{\mathbb{Q} \in \mathcal{B}_{\varepsilon_i}(\widehat{\mathbb{P}}_i^{N_i})} \left[U_i^S(\mathbf{s}_i, \mathbf{s}_{-i}^*) - \mathbb{E}_{\xi \sim \mathbb{Q}} [U_i(\mathbf{s}_i, \mathbf{s}_{-i}^*; \xi)] \right] \\
&\leq 2[C + 1] \varepsilon_i L_i.
\end{aligned} \tag{3.11}$$

Moreover, notice that the upper bound is independent of the chosen $\mathbf{s}_i$. Hence, by combining (3.8), (3.9), and (3.11),

$$\begin{aligned}
\forall \mathbf{s}_i \in \mathcal{S}_i, \quad U_i^S(\mathbf{s}_i^*, \mathbf{s}_{-i}^*) &\geq \overline{U}_i^{\text{DR}}(\mathbf{s}_i^*, \mathbf{s}_{-i}^*) \geq \overline{U}_i^{\text{DR}}(\mathbf{s}_i, \mathbf{s}_{-i}^*) \\
&\geq U_i^S(\mathbf{s}_i, \mathbf{s}_{-i}^*) - 2[C + 1]\varepsilon_i L_i \geq U_i^S(\mathbf{s}_i, \mathbf{s}_{-i}^*) - \eta,
\end{aligned}$$

with probability at least $1 - \theta_i \geq 1 - \theta$. Here, η and θ are as defined in the hypothesis.

Rearranging the last equation and further applying it to all agents completes the proof. $\blacksquare$

Remark 3.5.7. (*On relaxing the Lipschitz dependence of utility on uncertainty*). In Assumption 3.4.2: (3), we require $U_i(\mathbf{s}; \xi)$ to be L_i -Lipschitz in ξ , $\forall \mathbf{s} \in \mathcal{S}$. It is possible to relax this condition to require a Lipschitz constant $l_i(\mathbf{s})$ that depends on the strategy profile $\mathbf{s} \in \mathcal{S}$; and obtain the exact same result as in Theorem 3.5.6 with bound $\eta = 2[C + 1] \max_{i \in \mathcal{A}, \mathbf{s} \in \mathcal{S}} \varepsilon_i l_i(\mathbf{s})$. The extra requirement would impose that $\max_{\mathbf{s} \in \mathcal{S}} l_i(\mathbf{s})$ exists $\forall i \in \mathcal{A}$; which, in turn, is equivalent to Assumption 3.4.2: (3) to begin with. $\bullet$

The bounds provided in the previous result take into account the heterogeneity among the agents in terms of the dependence (through L_i) of their utility on the random variable and the size of their ambiguity sets (due to ε_i). Informally, for each agent $i \in \mathcal{A}$, ε_i and θ_i can be made smaller as the number of samples N_i grows (see [2]). Hence, the DRoNE becomes a NE of the stochastic game with high probability as the agents gather

more samples from the unknown distribution. In this chapter, we skip formal details regarding this matter, since we keep N_i fixed $\forall i \in \mathcal{A}$.

The previous result can be understood as follows. Suppose that every agent $i \in \mathcal{A}$ chooses ε_i independently, then Theorem 3.5.6 quantifies the uncertainty that the DRoNE belongs to $\mathcal{NE}(\mathcal{G}_S)$ using θ_i accordingly. We conclude this section by adapting the previous result for the special case with Assumptions 3.4.5, and 3.4.6. Here, we can use the ε, θ relation in Theorem 2.2.1 directly for the η bound, since there is no effect of observations. We skip the proof since it follows the exact same arguments as Theorem 3.5.6.

Corollary 3.5.8. (Under individual uncertainty, every DRoNE is an η -NE of $\mathcal{G}_S$ with certain probability). *Suppose Assumptions 3.4.5, and 3.4.6 hold. Suppose $\forall i \in \mathcal{A}$, $\theta_i \in (0, 1)$ and $\varepsilon_i(N_i, \theta_i)$ is chosen according to (2.4). Assume that the strategy $(\mathbf{s}_i^*, \mathbf{s}_{-i}^*) \in \mathcal{NE}(\mathcal{G}_{\text{DR}})$ with Wasserstein ball radii ε_i . Then with probability at least $(1 - \theta)^n$, $(\mathbf{s}_i^*, \mathbf{s}_{-i}^*) \in \mathcal{NE}_\eta(\mathcal{G}_S)$, with $\eta = 2 \max_{i \in \mathcal{A}} \varepsilon_i L_i$ and $\theta = \max_{i \in \mathcal{A}} \theta_i$. ■*

3.6 A Centralized Better Response Supergradient Ascent Dynamics

In the previous section, we established the existence of a DRoNE and showed its relation to the NE of the stochastic game. In this section, we provide a centralized algorithm that allows the agents to learn said DRoNE. We build up to a distributed solution that addresses Problem 3.4.8: (3) in the next section.

First, recall that Lemma 3.5.1 shows that for each agent $i \in \mathcal{A}$, $\overline{U}_i^{\text{DR}}$ is concave in its own strategy if the opponents' strategies are fixed. This means that (due to Definition 2.4.1) there is a non-empty set of supergradients of i 's utility with respect to its own strategy (for fixed opponents' strategies). We assume an additional feature of the distributionally robust utilities as follows.

Assumption 3.6.1. (Uniformly bounded supergradient). *For each $i \in \mathcal{A}$, $\exists B_i \in \mathbb{R}_{\geq 0}$ such that $\forall \mathbf{s} \in \mathcal{S}$, $\|\zeta_i\| \leq B_i$, $\forall \zeta_i \in \partial_{\mathbf{x}}[\overline{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i})]_{\mathbf{s}_i}$. •*

From the definitions of the NE in Definition 3.4.1 and the supergradient in Definition 2.4.1, it is well known that an alternative characterization of the Nash equilibria comes in the form of the following variational inequality [31],

$$\begin{aligned} \mathbf{s}^* \in \mathcal{NE}(\overline{\mathcal{G}}_{\text{DR}}) &\iff \forall \mathbf{s} \in \mathcal{S}, \exists \boldsymbol{\zeta} \in \bigtimes_{i \in \mathcal{A}} \partial_{\mathbf{x}} \left[\overline{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i}) \right] \Big|_{\mathbf{s}_i} \\ &\text{such that } (\mathbf{s} - \mathbf{s}^*)^\top \boldsymbol{\zeta} \leq 0. \end{aligned}$$

Hence, inspired by [32], we define the following Lyapunov function candidate, used for analysis later.

$$V(\mathbf{s}, \boldsymbol{\phi}) := \max_{\mathbf{x} \in \mathcal{S}} \mathbf{x}^\top \boldsymbol{\phi} - \mathbf{s}^\top \boldsymbol{\phi} \quad (3.12a)$$

$$= \sum_{i \in \mathcal{A}} \left[\max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i - \mathbf{s}_i^\top \boldsymbol{\phi}_i \right]. \quad (3.12b)$$

Here, $\boldsymbol{\phi}_i$ (like $\mathbf{s}_i$) are the components of a variable $\boldsymbol{\phi}$ (to be defined later) corresponding to agent $i \in \mathcal{A}$. The equality in (3.12b) comes from the fact that $\max_{\mathbf{x} \in \mathcal{S}} \mathbf{x}^\top \boldsymbol{\phi} = \max_{\mathbf{x} \in \mathcal{S}} \sum_{i \in \mathcal{A}} \mathbf{x}_i^\top \boldsymbol{\phi}_i = \sum_{i \in \mathcal{A}} \max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i$. This is because $\mathcal{S} = \times_{i \in \mathcal{A}} \mathcal{S}_i$ and the decision variables do not affect each other through other constraints. Note that V is Lipschitz continuous in its arguments as $\max_{\mathbf{x} \in \mathcal{S}} \mathbf{x}^\top \boldsymbol{\phi}$ is convex in $\boldsymbol{\phi}$, $\mathcal{S}$ is compact (hence $\max_{\mathbf{x} \in \mathcal{S}} \mathbf{x}^\top \boldsymbol{\phi}$ has uniformly bounded subgradients), and $\mathbf{s}^\top \boldsymbol{\phi}$ is bilinear. Moreover, by construction, $V(\mathbf{s}, \boldsymbol{\phi}) \geq 0, \forall \mathbf{s}, \boldsymbol{\phi}$. The next result bounds the change in $\max_{\mathbf{x} \in \mathcal{S}} \mathbf{x}^\top \boldsymbol{\phi}$ w.r.t. $\boldsymbol{\phi}$.

Lemma 3.6.2. (On the change of max of linear functions). *For an agent $i \in \mathcal{A}$, define $f_i(\boldsymbol{\phi}_i) := \max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i$, where $\mathcal{S}_i$ is a compact set with diameter $D_i \in \mathbb{R}_{>0}$, and let*

$$\mathcal{X}_i^*(\boldsymbol{\phi}_i) := \operatorname{argmax}_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i. \quad (3.13)$$

Then, $\forall \phi_i^1, \phi_i^2 \in \mathcal{B}_{M_i}(\mathbf{0})$, and $\forall \mathbf{x}_i^* \in \mathcal{X}_i^*(\phi_i^1)$,

$$f_i(\phi_i^2) \leq f_i(\phi_i^1) + \mathbf{x}_i^{*\top} [\phi_i^2 - \phi_i^1] + \overline{D}_i \|\phi_i^2 - \phi_i^1\|,$$

for any $\overline{D}_i \geq D_i$.

Proof. Consider an arbitrary but fixed ϕ_i^1 , with $\|\phi_i^1\| \leq M_i$ and an arbitrary but fixed $\mathbf{x}_i^* \in \mathcal{X}_i^*(\phi_i^1)$. Then $f_i(\phi_i^1) = \mathbf{x}_i^{*\top} \phi_i^1$. Further, since $\|\cdot\|$ is non-negative,

$$\mathbf{x}_i^{*\top} \phi_i \leq \mathbf{x}_i^{*\top} \phi_i^1 + \mathbf{x}_i^{*\top} [\phi_i - \phi_i^1] + \mathfrak{C} \|\phi_i - \phi_i^1\|,$$

$\forall \phi_i \in \mathcal{B}_{M_i}(\mathbf{0})$, for any $\mathfrak{C} \in \mathbb{R}_{\geq 0}$. Now take an arbitrary but fixed $\phi_i^2 \in \mathcal{B}_{M_i}(\mathbf{0})$ and let $\mathbf{y}_i^* \in \mathcal{X}_i^*(\phi_i^2)$. Thus, $f_i(\phi_i^2) = \mathbf{y}_i^{*\top} \phi_i^2$. Then, using the previous inequality,

$$\begin{aligned} \mathbf{y}_i^{*\top} \phi_i^2 &\leq \mathbf{x}_i^{*\top} \phi_i^1 + \mathbf{x}_i^{*\top} [\phi_i^2 - \phi_i^1] + \mathfrak{C} \|\phi_i^2 - \phi_i^1\| \\ &\quad + [\mathbf{y}_i^* - \mathbf{x}_i^*]^\top [\phi_i^2 - \phi_i^1] + [\mathbf{y}_i^* - \mathbf{x}_i^*]^\top \phi_i^1 \\ &\leq \mathbf{x}_i^{*\top} \phi_i^1 + \mathbf{x}_i^{*\top} [\phi_i^2 - \phi_i^1] + [\mathfrak{C} + D_i] \|\phi_i^2 - \phi_i^1\|. \end{aligned}$$

For the last inequality, we rely on the Cauchy-Schwarz inequality, $|\mathbf{y}_i^* - \mathbf{x}_i^*|^\top [\phi_i^2 - \phi_i^1]| \leq \|\mathbf{y}_i^* - \mathbf{x}_i^*\| \|\phi_i^2 - \phi_i^1\|$, and further used that D_i is the diameter of $\mathcal{S}_i$. Moreover, by definition $[\mathbf{y}_i^* - \mathbf{x}_i^*]^\top \phi_i^1 = \mathbf{y}_i^{*\top} \phi_i^1 - \mathbf{x}_i^{*\top} \phi_i^1 \leq 0$, since $\mathbf{y}_i^* \in \mathcal{S}_i$. Now, since $\mathfrak{C} + D_i \geq D_i$ is independent of ϕ_i^1, ϕ_i^2 , and $\mathbf{x}_i^*$, the proof is complete. $\blacksquare$

We are now ready to propose the first part of our algorithm. We allow agents to update their strategies at every time instant in a better response fashion. To that effect, with the set defined in (3.13), suppose

$$\pi_i^*(\phi_i) \in \mathcal{X}_i^*(\phi_i). \quad (3.14)$$

Notice here that $\boldsymbol{\pi}_i^*$ is related to the so-called *support function* of the convex set $\mathcal{S}_i$, which motivates the name of our algorithm. Now, given $\{\boldsymbol{\phi}(t)\}$, and using (3.14), agents update their strategies via

$$\begin{aligned}\mathbf{s}_i(t+1) &= [1 - \alpha_i] \mathbf{s}_i(t) + \alpha_i \boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i(t)), \\ &= \mathbf{s}_i(t) + \alpha_i [\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i(t)) - \mathbf{s}_i(t)], \quad \forall i \in \mathcal{A},\end{aligned}\tag{3.15}$$

from some initial $\{\mathbf{s}_i(0) \in \mathcal{S}_i\}_{i \in \mathcal{A}}$. In other words, the update of $\mathbf{s}_i(t)$ is a convex combination of $\mathbf{s}_i(t)$ and $\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i(t))$ using the parameter $\alpha_i \in (0, 1)$, for each $i \in \mathcal{A}$. Thus, $\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i(t))$ renders $\mathcal{S}_i$ invariant under (3.15), $\forall i \in \mathcal{A}$. Before specifying $\boldsymbol{\phi}_i(t)$, we bound the change of the Lyapunov function for any choice of bounded $\boldsymbol{\phi}(t)$.

Lemma 3.6.3. (Bound on Lyapunov function difference). *Consider the dynamics (3.15) from an initial condition $\{\mathbf{s}_i(0) \in \mathcal{S}_i\}_{i \in \mathcal{A}}$. Suppose $\{\boldsymbol{\phi}(t)\}_{t \in \mathbb{Z}_{\geq 0}}$ satisfies $\|\boldsymbol{\phi}_i(t)\| \leq M_i$ (for some $M_i \in \mathbb{R}_{>0}$), $\forall i \in \mathcal{A}$, $\forall t \in \mathbb{Z}_{\geq 0}$. Take any $\bar{D}_i \geq D_i$, $\forall i \in \mathcal{A}$. Then, $\forall t \in \mathbb{Z}_{\geq 0}$, with $\underline{\alpha} = \min_{i \in \mathcal{A}} \alpha_i \in (0, 1)$, it holds that*

$$\begin{aligned}V(\mathbf{s}(t+1), \boldsymbol{\phi}(t+1)) - V(\mathbf{s}(t), \boldsymbol{\phi}(t)) &\leq \\ &\sum_{i \in \mathcal{A}} \left[\frac{1 - \alpha_i}{\alpha_i} [\mathbf{s}_i(t+1) - \mathbf{s}_i(t)]^\top [\boldsymbol{\phi}_i(t+1) - \boldsymbol{\phi}_i(t)] \right. \\ &\quad \left. + \bar{D}_i \|\boldsymbol{\phi}_i(t+1) - \boldsymbol{\phi}_i(t)\| \right] - \underline{\alpha} V(\mathbf{s}(t), \boldsymbol{\phi}(t)).\end{aligned}\tag{3.16}$$

Proof. With a slight abuse of notation, we denote $\mathbf{s} = \mathbf{s}(t)$, $\mathbf{s}^+ = \mathbf{s}(t+1)$, $\boldsymbol{\phi} = \boldsymbol{\phi}(t)$, and $\boldsymbol{\phi}^+ = \boldsymbol{\phi}(t+1)$. From the definition of V in (3.12) and the dynamics in (3.15), we

have that

$$\begin{aligned}
& V(\mathbf{s}^+, \boldsymbol{\phi}^+) - V(\mathbf{s}, \boldsymbol{\phi}) \\
&= \sum_{i \in \mathcal{A}} \left[\max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i^+ - \max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i - \mathbf{s}_i^{+\top} \boldsymbol{\phi}_i^+ + \mathbf{s}_i^\top \boldsymbol{\phi}_i \right] \\
&= \sum_{i \in \mathcal{A}} \left[\max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i^+ - \max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i \right. \\
&\quad \left. - \mathbf{s}_i^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] - [\mathbf{s}_i^+ - \mathbf{s}_i]^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] - [\mathbf{s}_i^+ - \mathbf{s}_i]^\top \boldsymbol{\phi}_i \right].
\end{aligned}$$

Then, we can upper bound the time difference in V as

$$\begin{aligned}
& V(\mathbf{s}^+, \boldsymbol{\phi}^+) - V(\mathbf{s}, \boldsymbol{\phi}) \\
&\leq \sum_{i \in \mathcal{A}} \left[\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i)^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] + \overline{D}_i \|\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i\| \right. \\
&\quad \left. - \mathbf{s}_i^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] - \alpha_i [\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i) - \mathbf{s}_i]^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] \right. \\
&\quad \left. - \alpha_i [\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i) - \mathbf{s}_i]^\top \boldsymbol{\phi}_i \right].
\end{aligned}$$

The first two terms on the right-hand side of the previous inequality come from applying Lemma 3.6.2; and the last two terms similarly come from replacing (3.15) and rearranging terms. Then, combining like terms gives,

$$\begin{aligned}
& V(\mathbf{s}^+, \boldsymbol{\phi}^+) - V(\mathbf{s}, \boldsymbol{\phi}) \\
&\leq \sum_{i \in \mathcal{A}} \left[[\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i) - \mathbf{s}_i]^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] + \overline{D}_i \|\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i\| \right. \\
&\quad \left. - \alpha_i [\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i) - \mathbf{s}_i]^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] - \alpha_i [\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i) - \mathbf{s}_i]^\top \boldsymbol{\phi}_i \right] \\
&\leq \sum_{i \in \mathcal{A}} \left[[1 - \alpha_i] [\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i) - \mathbf{s}_i]^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] \right. \\
&\quad \left. + \overline{D}_i \|\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i\| \right] - \underline{\alpha} V(\mathbf{s}, \boldsymbol{\phi}).
\end{aligned}$$

For the last inequality, we have used the fact that $\sum_{i \in \mathcal{A}} [\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i) - \mathbf{s}_i]^\top \boldsymbol{\phi}_i = \sum_{i \in \mathcal{A}} [\max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i - \mathbf{s}_i^\top \boldsymbol{\phi}_i] = V(\mathbf{s}, \boldsymbol{\phi})$, and combined terms from the previous step. Finally, applying (3.15) in the last inequality completes the proof. $\blacksquare$

Recall that supergradients are possibly non-unique. Thus, an alternate definition of the NE [31] is given by

$$\mathbf{s}^* \in \mathcal{NE}(\overline{\mathcal{G}}_{\text{DR}}) \iff \mathbf{0} \in \mathcal{NC}_{\mathcal{S}_i}(\mathbf{s}_i^*) - \partial_{\mathbf{x}} \left[\overline{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i}^*) \right] \Big|_{\mathbf{s}_i^*}.$$

This motivates us to introduce the following min norm supergradient. First, let $\Pi_{\mathbf{x}}(\mathbf{x}, \mathbf{y})$ be the projection of $(\mathbf{x}, \mathbf{y})$ onto $\mathbf{x}$, the first components. Then, define $\forall i \in \mathcal{A}$,

$$\begin{aligned} \mathbf{v}_i(\mathbf{s}) &:= \Pi_{\boldsymbol{\zeta}_i} \left(\underset{\boldsymbol{\zeta}_i, \mathbf{z}}{\operatorname{argmin}} \|\cdot - \boldsymbol{\zeta}_i + \mathbf{z}\|^2 \right. \\ &\quad \left. \text{s.t. } \boldsymbol{\zeta}_i \in \partial_{\mathbf{x}} \left[\overline{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i}) \right] \Big|_{\mathbf{s}_i}, \mathbf{z} \in \mathcal{NC}_{\mathcal{S}_i}(\mathbf{s}_i) \right). \end{aligned} \quad (3.17)$$

Note that the previous formulation in (3.17) is well defined since $\|\cdot\|^2$ is strongly convex and lower bounded by 0; the set of supergradients is convex and compact; and the normal cone is a closed convex cone. Hence, (3.17) produces a unique optimizer. Further, we introduce a proximal term $\forall i \in \mathcal{A}$, with $\lambda_i \in \mathbb{R}_{>0}$,

$$\mathbf{w}_i(\mathbf{s}_i, t) := -\frac{1}{\lambda_i} [\mathbf{s}_i - \mathbf{s}_i(t-1)], \quad (3.18)$$

which is the derivative of the $(1/\lambda_i)$ -strongly concave function $-(1/\lambda_i)\|\mathbf{s}_i - \mathbf{s}_i(t-1)\|^2$. This penalizes the deviation of the strategy at time $t \in \mathbb{Z}_{\geq 0}$ from the previous strategy, and this penalty depends on the multiplier λ_i^{-1} . Now, using the definitions in (3.17), and (3.18), we propose the following choice of $\boldsymbol{\phi}_i(t)$ for (3.15), with $\mu_i > 0$,

$$\boldsymbol{\phi}_i(t) = \mu_i \mathbf{v}_i(\mathbf{s}(t)) + \mathbf{w}_i(\mathbf{s}_i(t), t). \quad (3.19)$$

From here, and under Assumption 3.6.1, it is easy to see that $\forall i \in \mathcal{A}$, $\|\phi_i(t)\| \leq \mu_i B_i + D_i/\lambda_i$, $\forall t \in \mathbb{Z}_{\geq 0}$. We formally define our algorithm dynamics next and discuss the particular choice of $\phi(t)$ in the remark that follows.

Definition 3.6.4. (*ISBRAG*). The dynamics obtained from the strategy update in (3.15), using the modified supergradient vector in (3.19) is referred to as *Inertial Supported Better Response Ascending superGradient* dynamics or *ISBRAG*. •

Remark 3.6.5. (*On the choice of $\phi_i(t)$*). From the definitions in (3.17), and (3.18), it is easy to realize that $\forall i \in \mathcal{A}$,

$$\begin{aligned} \phi_i(t) &= \partial_{\mathbf{x}} \left[\mu_i \bar{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i}(t)) \right] \Big|_{\mathbf{s}_i(t)} - \frac{1}{\lambda_i} [\mathbf{s}_i(t) - \mathbf{s}_i(t-1)] \\ &\in \partial_{\mathbf{x}} \left[\mu_i \bar{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i}(t)) - \frac{1}{2\lambda_i} \|\mathbf{x} - \mathbf{s}_i(t-1)\|^2 \right] \Big|_{\mathbf{s}_i(t)}. \end{aligned}$$

This produces the effect of allowing the agents to update their strategies at every time instant using the knowledge of the supergradient direction (scaled by μ_i) of their utility, along with an inertial direction so that they do not deviate from their previous strategies too much. •

Notice from (3.19) that ISBRAG uses a one time-step delayed state information to perform the current state update in (3.15). In such a case, it is useful to define an auxiliary state $\mathbf{p}_i(t) = \mathbf{s}_i(t-1)$, $\forall i \in \mathcal{A}$, $\forall t \in \mathbb{Z}_{\geq 0}$. Then, the equilibrium set for ISBRAG is given by

$$\mathcal{S}_{\text{eq}} := \{(\mathbf{s}, \mathbf{p}) \in \mathcal{S} \times \mathcal{S} \mid \mathbf{s}_i = \mathbf{p}_i \in \mathcal{X}_i^*(\mathbf{v}_i(\mathbf{s}_i)), \forall i \in \mathcal{A}\}. \quad (3.20)$$

Next, we establish V as a size function (see Definition 2.3.1) for a superset of the DRoNE set. This will be useful in proving convergence results in the sequel.

Lemma 3.6.6. (V is a size function). *Suppose Assumption 3.6.1 holds. Define $\bar{\mathcal{B}} :=$*

$\times_{i \in \mathcal{A}} \mathcal{B}_{B_i}(\mathbf{0})$ and let

$$\tilde{\phi}_i(\mathbf{s}_i, \mathbf{p}_i, \boldsymbol{\zeta}_i) = \mu_i \boldsymbol{\zeta}_i - \frac{1}{\lambda_i} [\mathbf{s}_i - \mathbf{p}_i], \quad \forall i \in \mathcal{A}; \quad (3.21)$$

where, $\forall i \in \mathcal{A}$, $\mathbf{s}_i, \mathbf{p}_i \in \mathcal{S}_i$, and $\boldsymbol{\zeta}_i \in \mathcal{B}_{B_i}(\mathbf{0})$. Define,

$$\mathcal{Z} := \left\{ (\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}) \in \mathcal{S} \times \mathcal{S} \times \overline{\mathcal{B}} \mid \right. \quad (3.22)$$

$$\left. \forall i \in \mathcal{A}, \mathbf{s}_i \in \mathcal{X}_i^*(\tilde{\phi}_i(\mathbf{s}_i, \mathbf{p}_i, \boldsymbol{\zeta}_i)) \right\}.$$

Then, $V(\mathbf{s}, \tilde{\phi}(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}))$ as a function of $(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta})$ is a size function for $\mathcal{Z}$. Moreover,

$$\left\{ (\mathbf{s}^*, \mathbf{s}^*, \boldsymbol{\zeta}^*) \mid \mathbf{s}^* \in \mathcal{NE}(\overline{\mathcal{G}}_{\text{DR}}), \boldsymbol{\zeta}_i^* = \mathbf{v}_i(\mathbf{s}^*), \forall i \in \mathcal{A} \right\} \subseteq \mathcal{Z}.$$

Proof. First recall from (3.12) and the discussion following it that V is Lipschitz and; hence, continuous. Moreover, recall that by definition, $V(\mathbf{s}, \boldsymbol{\phi}) \geq 0$, $\forall \mathbf{s} \in \mathcal{S}$, and $\boldsymbol{\phi}$. From (3.18) and (3.19), suppose $\mathbf{p} \in \mathcal{S}$ (with components $\mathbf{p}_i \in \mathcal{S}_i$, $i \in \mathcal{A}$) describes the state at the previous time step; then

$$\phi_i(\mathbf{s}, \mathbf{p}) = \mu_i \mathbf{v}_i(\mathbf{s}) - \frac{1}{\lambda_i} [\mathbf{s}_i - \mathbf{p}_i], \quad \forall i \in \mathcal{A}.$$

Since $\forall i \in \mathcal{A}$, $\mathbf{v}_i(\mathbf{s}) \in \partial_{\mathbf{x}} [\overline{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i})] \big|_{\mathbf{s}_i}$, we can consider the evolution of ISBRAG (Definition 3.6.4) with $\tilde{\phi}$ in (3.21) in the set $\mathcal{S} \times \mathcal{S} \times \overline{\mathcal{B}}$. Because of the assumption on the compactness of $\mathcal{S}_i$, and Assumption 3.6.1, $\forall i \in \mathcal{A}$, $\tilde{\phi}_i \in \mathcal{B}_{\mu_i B_i + D_i / \lambda_i}(\mathbf{0})$, $\boldsymbol{\zeta}_i \in \mathcal{B}_{B_i}(\mathbf{0})$, $\mathbf{s}_i \in \mathcal{S}_i$ and $\mathbf{p}_i \in \mathcal{S}_i$. Then, using these definitions and with a slight abuse of notation $V(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}) := V(\mathbf{s}, \tilde{\phi})$, we have that $V(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta})$ is continuous in all its arguments; since $\tilde{\phi}$ in (3.21) is continuous in its arguments. For the sake of brevity, define $\mathcal{Y} := \mathcal{S} \times \mathcal{S} \times \overline{\mathcal{B}}$, which is a compact set. Then, $V(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta})$ attains its maximum and minimum on $\mathcal{Y}$, and it is easy to deduce that there is a continuous extension of V (say $\tilde{V}$) that is proper. For

example,

$$\tilde{V}(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}) = \begin{cases} V(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}), & \text{if } (\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}) \in \mathcal{Y}; \\ d((\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}), \mathcal{Y}) + V(\mathbf{s}^*, \mathbf{p}^*, \boldsymbol{\zeta}^*) & \text{otherwise;} \end{cases}$$

where $(\mathbf{s}^*, \mathbf{p}^*, \boldsymbol{\zeta}^*) \in \operatorname{argmin}_{(\hat{\mathbf{s}}, \hat{\mathbf{p}}, \hat{\boldsymbol{\zeta}}) \in \mathcal{Y}} \|(\hat{\mathbf{s}}, \hat{\mathbf{p}}, \hat{\boldsymbol{\zeta}}) - (\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta})\|$.

Now, $\tilde{V}(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}) \geq 0$, $\forall (\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}) \in \mathcal{Y}$. Moreover, $\tilde{V}(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}) = 0$ iff $\forall i \in \mathcal{A}$, $\max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \tilde{\boldsymbol{\phi}}_i(\mathbf{s}_i, \mathbf{p}_i, \boldsymbol{\zeta}_i) = \mathbf{s}_i^\top \tilde{\boldsymbol{\phi}}_i(\mathbf{s}_i, \mathbf{p}_i, \boldsymbol{\zeta}_i)$, i.e. $\mathbf{s}_i \in \mathcal{X}_i^*(\tilde{\boldsymbol{\phi}}_i(\mathbf{s}_i, \mathbf{p}_i, \boldsymbol{\zeta}_i))$. Further, since $\tilde{V}$ is continuous, $\forall l \in \mathbb{R}_{\geq 0}$, the l -sublevel set $\{(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}) \mid \tilde{V}(\mathbf{s}, \mathbf{p}, \boldsymbol{\zeta}) \leq l\}$ is a closed subset of $\mathbb{R}^n$. Now, by construction, these l -sublevel sets are compact, since V attains its maximum in $\mathcal{Y}$. Thus, by Definition 2.3.1, $\tilde{V}$ is a size function for $\mathcal{Z}$. Notice from (3.15) and (3.19) that the set $\mathcal{Y}$ is further invariant under ISBRAG. Thus equivalently, V is a size function for $\mathcal{Z}$.

The last claim on subset relationship can be easily verified from (3.17) and the definition of a NE in Definition 3.4.1. This completes the proof. $\blacksquare$

Before providing convergence guarantees for our algorithm, we define a class of games for which the supergradients play well with the proximal terms.

Definition 3.6.7. (*Amicable supergradients*). The game $\overline{\mathcal{G}}_{\text{DR}}$ (satisfying Standing Assumption 3.4.2) is said to have *amicable supergradients* if the following holds. Consider an arbitrary but fixed $i \in \mathcal{A}$. Let $(\mathbf{s}_i, \mathbf{s}_{-i}) \in \mathcal{S}$. Then, $\exists d_i \in \mathbb{R}_{>0}$ such that $\forall (\bar{\mathbf{s}}_i, \bar{\mathbf{s}}_{-i}) \in \mathcal{B}_{d_i}(\mathbf{s}_i, \mathbf{s}_{-i})$, $\exists \boldsymbol{\zeta}_i^1 \in \partial_{\mathbf{x}}[\overline{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i})] \big|_{\mathbf{s}_i}$, $\exists \boldsymbol{\zeta}_i^2 \in \partial_{\mathbf{x}}[\overline{U}_i^{\text{DR}}(\mathbf{x}, \bar{\mathbf{s}}_{-i})] \big|_{\mathbf{s}_i}$, and $\exists \boldsymbol{\zeta}_i^3 \in \partial_{\mathbf{x}}[\overline{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i})] \big|_{\bar{\mathbf{s}}_i}$, such that,

$$[\boldsymbol{\zeta}_i^2 - \boldsymbol{\zeta}_i^1]^\top [\bar{\mathbf{s}}_i - \mathbf{s}_i] - [\boldsymbol{\zeta}_i^3 - \boldsymbol{\zeta}_i^1]^\top [\bar{\mathbf{s}}_i - \mathbf{s}_i] \leq c_i \|\bar{\mathbf{s}}_i - \mathbf{s}_i\|^2,$$

for some $c_i \in \mathbb{R}$. This $\{c_i\}_{i \in \mathcal{A}}$ is referred to as the *factors of amicability* of the supergradi-

ents.

•

The name *amicable* is chosen to represent the fact that the supergradients behave in a “friendly” manner with respect to the proximal term in (3.18). The significance of the previous definition will come into focus in the proof of the next theorem, where we will use the factors of concavity $\{\lambda_i^{-1}\}_{i \in \mathcal{A}}$ to offset the factors of amicability $\{c_i\}_{i \in \mathcal{A}}$ and produce monotonicity-like [32] behavior. In this regard, “amicability” can be seen to extend monotonicity. Next, we state and prove conditions under which ISBRAG converges.

Theorem 3.6.8. (Convergence of ISBRAG). *Suppose Assumption 3.6.1 holds. Moreover, suppose $\overline{\mathcal{G}}_{\text{DR}}$ has amicable supergradients with factor of amicability $\{c_i\}_{i \in \mathcal{A}}$. With $d_{\min} := \min_{i \in \mathcal{A}} d_i$, choose $\mu_i \in (0, \infty)$, $\forall i \in \mathcal{A}$ and*

$$\alpha_i \in \left(0, \min \left\{ \frac{d_{\min}}{D_i}, \frac{1}{2} \right\} \right], \quad \forall i \in \mathcal{A}. \quad (3.23)$$

Next, $\forall i \in \mathcal{A}$, choose $\mathfrak{N}_i \in (1, \infty)$ and suppose

$$\lambda_i \in \begin{cases} \left(\frac{1}{\mathfrak{N}_i \mu_i c_i}, \frac{1}{\mu_i c_i} \right), & \text{if } c_i > 0; \\ \left(\frac{1}{\mathfrak{N}_i \mu_i}, \infty \right), & \text{otherwise.} \end{cases} \quad (3.24)$$

Define the size function $\omega(\cdot) := d(\cdot, \mathcal{NE}(\overline{\mathcal{G}}_{\text{DR}}))$ as the distance from the NE set of $\overline{\mathcal{G}}_{\text{DR}}$. Denote $\bar{\alpha} := \max_{i \in \mathcal{A}} \alpha_i$, $\bar{\mu} := \max_{i \in \mathcal{A}} \mu_i$. Finally, let $\mathbf{s}(t)$ be a trajectory of the ISBRAG dynamics in Definition 3.6.4 from the initial condition $\{\mathbf{s}_i(0) = \mathbf{s}_i(-1) \in \mathcal{S}_i\}_{i \in \mathcal{A}}$. Then, there exist functions $\beta_1 \in \mathcal{KL}$, $\gamma_1 \in \mathcal{K}$, and a continuous function $\rho_1 : [0, 0.5] \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$

such that

$$\omega(\mathbf{s}(t)) \leq \max \left\{ \beta_1 \left(\omega(\mathbf{s}(0)), t \right), \gamma_1 \left(K \rho_1(\bar{\alpha}, \bar{\mu}) \right) \right\}, \quad (3.25)$$

where, the constant K depends only on $\{D_i, B_i, c_i, \mathfrak{N}_i\}_{i \in \mathcal{A}}$. Moreover, ρ_1 is strictly increasing w.r.t. both its arguments and $\rho_1(x, y) \rightarrow 0$ as $(x, y) \rightarrow 0$.

Proof. First, notice that because of Lemmas 2.3.2 and 3.6.6, there exists functions $\sigma_1, \sigma_2 \in \mathcal{K}_\infty$ such that

$$\sigma_1(\omega(\mathbf{s})) \leq V(\mathbf{s}) \leq \sigma_2(\omega(\mathbf{s})), \quad \forall \mathbf{s} \in \mathcal{S}. \quad (3.26)$$

This can be seen from the statement of Lemma 3.6.6 and realizing that the set $\{(\mathbf{s}, \boldsymbol{\zeta}) \mid \forall i \in \mathcal{A}, \mathbf{s}_i \in \mathcal{S}_i, \boldsymbol{\zeta}_i \in \partial_{\mathbf{x}_i}[\bar{U}_i^{\text{DR}}(\mathbf{x}_i, \mathbf{s}_{-i})]_{|\mathbf{s}_i} \}$ is invariant under ISBRAG.

Now, to bound the change in V , we first provide an upper bound on $[\mathbf{s}_i(t+1) - \mathbf{s}_i(t)]^\top [\boldsymbol{\phi}_i(t+1) - \boldsymbol{\phi}_i(t)]$, $\forall i \in \mathcal{A}$. With a slight abuse of notation, set $\mathbf{s}_i = \mathbf{s}_i(t)$, $\mathbf{s}_i^+ = \mathbf{s}_i(t+1)$, $\mathbf{s}_i^- = \mathbf{s}_i(t-1)$, $\mathbf{s}_{-i} = \mathbf{s}_{-i}(t)$, $\mathbf{s}_{-i}^+ = \mathbf{s}_{-i}(t+1)$, $\boldsymbol{\phi}_i^- = \boldsymbol{\phi}_i(t-1)$, $\boldsymbol{\phi}_i = \boldsymbol{\phi}_i(t)$, and $\boldsymbol{\phi}_i^+ = \boldsymbol{\phi}_i(t+1)$.

Consider an arbitrary but fixed agent $i \in \mathcal{A}$. By the property of supergradients, at time t , we have, $\forall \mathbf{x} \in \mathcal{S}_i$,

$$\begin{aligned} \mu_i \bar{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i}) - \frac{1}{2\lambda_i} \|\mathbf{x} - \mathbf{s}_i^-\|^2 &\leq \mu_i \bar{U}_i^{\text{DR}}(\mathbf{s}_i, \mathbf{s}_{-i}) \\ &\quad - \frac{1}{2\lambda_i} \|\mathbf{s}_i - \mathbf{s}_i^-\|^2 + \boldsymbol{\phi}_i^\top [\mathbf{x} - \mathbf{s}_i] - \frac{1}{2\lambda_i} \|\mathbf{x} - \mathbf{s}_i\|^2; \end{aligned} \quad (3.27)$$

and at time $t+1$, we have, $\forall \mathbf{x} \in \mathcal{S}_i$,

$$\begin{aligned} \mu_i \bar{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i}^+) - \frac{1}{2\lambda_i} \|\mathbf{x} - \mathbf{s}_i\|^2 &\leq \mu_i \bar{U}_i^{\text{DR}}(\mathbf{s}_i^+, \mathbf{s}_{-i}^+) \\ &\quad - \frac{1}{2\lambda_i} \|\mathbf{s}_i^+ - \mathbf{s}_i\|^2 + \boldsymbol{\phi}_i^{+\top} [\mathbf{x} - \mathbf{s}_i^+] - \frac{1}{2\lambda_i} \|\mathbf{x} - \mathbf{s}_i^+\|^2. \end{aligned} \quad (3.28)$$

Note that the last quadratic terms in both of the previous two equations come from the strong concavity of the proximal term $\|\cdot\|^2$ [53]. Substituting $\mathbf{x} = \mathbf{s}_i^+$ in (3.27), $\mathbf{x} = \mathbf{s}_i$ in (3.28); and combining the inequalities, we get

$$\begin{aligned}
& [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i]^\top [\mathbf{s}_i^+ - \mathbf{s}_i] \\
& \leq \frac{1}{2\lambda_i} \|\mathbf{s}_i^+ - \mathbf{s}_i^-\|^2 - \frac{1}{2\lambda_i} \|\mathbf{s}_i - \mathbf{s}_i^-\|^2 - \frac{1}{2\lambda_i} \|\mathbf{s}_i^+ - \mathbf{s}_i\|^2 \\
& \quad - \left[\frac{1}{2\lambda_i} + \frac{1}{2\lambda_i} \right] \|\mathbf{s}_i^+ - \mathbf{s}_i\|^2 + \mu_i \left[\bar{U}_i^{\text{DR}}(\mathbf{s}_i, \mathbf{s}_{-i}) \right. \\
& \quad \left. - \bar{U}_i^{\text{DR}}(\mathbf{s}_i^+, \mathbf{s}_{-i}) + \bar{U}_i^{\text{DR}}(\mathbf{s}_i^+, \mathbf{s}_{-i}^+) - \bar{U}_i^{\text{DR}}(\mathbf{s}_i, \mathbf{s}_{-i}^+) \right].
\end{aligned}$$

Noting that $\|\mathbf{s}_i^+ - \mathbf{s}_i^-\|^2 = \|\mathbf{s}_i^+ - \mathbf{s}_i\|^2 + 2[\mathbf{s}_i^+ - \mathbf{s}_i]^\top [\mathbf{s}_i - \mathbf{s}_i^-] + \|\mathbf{s}_i - \mathbf{s}_i^-\|^2$, and applying appropriate algebraic simplifications, leads to

$$\begin{aligned}
& [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i]^\top [\mathbf{s}_i^+ - \mathbf{s}_i] \\
& \leq \frac{1}{\lambda_i} [\mathbf{s}_i^+ - \mathbf{s}_i]^\top [\mathbf{s}_i - \mathbf{s}_i^-] - \frac{1}{\lambda_i} \|\mathbf{s}_i^+ - \mathbf{s}_i\|^2 \\
& \quad + \mu_i \left[\bar{U}_i^{\text{DR}}(\mathbf{s}_i, \mathbf{s}_{-i}) - \bar{U}_i^{\text{DR}}(\mathbf{s}_i^+, \mathbf{s}_{-i}) + \bar{U}_i^{\text{DR}}(\mathbf{s}_i^+, \mathbf{s}_{-i}^+) - \bar{U}_i^{\text{DR}}(\mathbf{s}_i, \mathbf{s}_{-i}^+) \right].
\end{aligned}$$

Now, suppose $\boldsymbol{\zeta}_i(\mathbf{s}_i^+, \mathbf{s}_{-i}) \in \partial_{\mathbf{x}} [\bar{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i})] \Big|_{\mathbf{s}_i^+}$, $\boldsymbol{\zeta}_i(\mathbf{s}_i, \mathbf{s}_{-i}^+) \in \partial_{\mathbf{x}} [\bar{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i}^+)] \Big|_{\mathbf{s}_i}$. Then using the appropriate definition of the supergradients (see Definition 2.4.1), we modify the

previous bound on $[\phi_i^+ - \phi_i]^\top [\mathbf{s}_i^+ - \mathbf{s}_i]$ as

$$\begin{aligned}
& [\phi_i^+ - \phi_i]^\top [\mathbf{s}_i^+ - \mathbf{s}_i] \\
& \leq \frac{1}{\lambda_i} [\mathbf{s}_i^+ - \mathbf{s}_i]^\top [\mathbf{s}_i - \mathbf{s}_i^-] - \frac{1}{\lambda_i} \|\mathbf{s}_i^+ - \mathbf{s}_i\|^2 \\
& \quad + \mu_i \left[\zeta_i(\mathbf{s}_i, \mathbf{s}_{-i}^+) - \zeta_i(\mathbf{s}_i^+, \mathbf{s}_{-i}) \right]^\top [\mathbf{s}_i^+ - \mathbf{s}_i] \\
& = \frac{1}{\lambda_i} [\mathbf{s}_i^+ - \mathbf{s}_i]^\top [\mathbf{s}_i - \mathbf{s}_i^-] - \frac{1}{\lambda_i} \|\mathbf{s}_i^+ - \mathbf{s}_i\|^2 \\
& \quad + \mu_i \left[\zeta_i(\mathbf{s}_i, \mathbf{s}_{-i}^+) - \zeta_i(\mathbf{s}_i, \mathbf{s}_{-i}) \right]^\top [\mathbf{s}_i^+ - \mathbf{s}_i] \\
& \quad - \mu_i \left[\zeta_i(\mathbf{s}_i^+, \mathbf{s}_{-i}) - \zeta_i(\mathbf{s}_i, \mathbf{s}_{-i}) \right]^\top [\mathbf{s}_i^+ - \mathbf{s}_i].
\end{aligned}$$

Here, for the last equality, we have used the linearity property of inner products. Then, from Definition 3.6.7,

$$\begin{aligned}
& [\phi_i^+ - \phi_i]^\top [\mathbf{s}_i^+ - \mathbf{s}_i] \leq \\
& \frac{1}{\lambda_i} [\mathbf{s}_i^+ - \mathbf{s}_i]^\top [\mathbf{s}_i - \mathbf{s}_i^-] - \frac{1}{\lambda_i} \|\mathbf{s}_i^+ - \mathbf{s}_i\|^2 + \mu_i c_i \|\mathbf{s}_i^+ - \mathbf{s}_i\|^2.
\end{aligned}$$

Indeed, we can use the inequality in Definition 3.6.7, since we have chosen $\alpha_i < d_{\min}/D_i$ to ensure that the opponents' strategies remain within the d_i -ball around the current strategy profile. Now, because of (3.24), $(-\frac{1}{\lambda_i} + \mu_i c_i) \|\mathbf{s}_i^+ - \mathbf{s}_i\|^2 \leq 0$. Then, substituting in the dynamics (3.15) in the previous inequality gives us

$$\begin{aligned}
& [\phi_i^+ - \phi_i]^\top [\mathbf{s}_i^+ - \mathbf{s}_i] \leq \frac{1}{\lambda_i} [\mathbf{s}_i^+ - \mathbf{s}_i]^\top [\mathbf{s}_i - \mathbf{s}_i^-] \\
& = \frac{\alpha_i^2}{\lambda_i} [\boldsymbol{\pi}_i^*(\phi_i) - \mathbf{s}_i]^\top [\boldsymbol{\pi}_i^*(\phi_i^-) - \mathbf{s}_i^-].
\end{aligned}$$

Recall that $\boldsymbol{\pi}_i^*(\phi_i), \boldsymbol{\pi}_i^*(\phi_i^-) \in \mathcal{S}_i$. Relying on the Cauchy-Schwarz inequality on the

previous step and utilizing the fact that D_i is the diameter of $\mathcal{S}_i$, we conclude that

$$\begin{aligned} [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i]^\top [\mathbf{s}_i^+ - \mathbf{s}_i] &\leq \\ \frac{\alpha_i^2}{\lambda_i} \|\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i) - \mathbf{s}_i\| \|\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i^-) - \mathbf{s}_i^-\| &\leq \alpha_i^2 \frac{D_i^2}{\lambda_i}. \end{aligned} \quad (3.29)$$

Finally, we provide an upper bound for $\|\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i\|$. Quickly, define $\forall i \in \mathcal{A}$,

$$\tilde{c}_i = \begin{cases} c_i, & \text{if } c_i > 0; \\ 1, & \text{otherwise;} \end{cases}$$

and notice that $\frac{1}{\lambda_i} < \mathfrak{N}_i \mu_i \tilde{c}_i$, $\forall i \in \mathcal{A}$. Then, using the properties of $\|\cdot\|$ and the triangle inequality,

$$\begin{aligned} \|\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i\| &\leq \mu_i \|\mathbf{v}_i^+ - \mathbf{v}_i\| + \frac{1}{\lambda_i} \|\mathbf{s}_i^+ - \mathbf{s}_i^-\| \\ &\leq 2\mu_i B_i + \frac{D_i}{\lambda_i} \leq (2B_i + D_i \mathfrak{N}_i \tilde{c}_i) \mu_i =: A_i \mu_i. \end{aligned} \quad (3.30)$$

Recall that $\overline{D}_i \geq D_i$, $\forall i \in \mathcal{A}$. Now, combining (3.16) from Lemma 3.6.3, (3.29), and (3.30), gives us,

$$\begin{aligned} V(\mathbf{s}(t+1), \boldsymbol{\phi}(t+1)) - V(\mathbf{s}(t), \boldsymbol{\phi}(t)) & \\ \leq \sum_{i \in \mathcal{A}} \left[[1 - \alpha_i] \alpha_i \frac{D_i^2}{\lambda_i} + \mu_i A_i \overline{D}_i \right] - \underline{\alpha} V(\mathbf{s}(t), \boldsymbol{\phi}(t)) & \\ \leq -\underline{\alpha} V(\mathbf{s}(t), \boldsymbol{\phi}(t)) + K \rho(\overline{\alpha}, \overline{\mu}) & \end{aligned} \quad (3.31)$$

where, $\forall i \in \mathcal{A}$,

$$K_i := \max \left\{ \frac{D_i^2}{\lambda_i}, (2B_i + D_i \mathfrak{N}_i \tilde{c}_i) \overline{D}_i \right\}, \quad (3.32)$$

the constant $K := \max_{i \in \mathcal{A}} nK_i$, and the function $\rho_1 : [0, 0.5] \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is defined as

$$\rho_1(x, y) = x - x^2 + y. \quad (3.33)$$

Observe that the last inequality in (3.31) holds since $x - x^2 + 1$ is strictly increasing in $[0, 0.5]$ and as such $\alpha_i - \alpha_i^2 + 1 \leq \bar{\alpha}_i - \bar{\alpha}_i^2 + 1$, $\forall i \in \mathcal{A}$. Moreover, from Lemma 3.6.3, $\underline{\alpha} > 0$. Thus, comparing (3.26) and (3.31) with (2.6a) and (2.6b), we conclude that the system is ISS with input $K\rho_1(\bar{\alpha}, \bar{\mu})$. Then, (3.25) follows immediately from Definition 2.3.3. Moreover, the properties of ρ_1 are immediate from its definition in (3.33). $\blacksquare$

From the previous result, we see that, given the right choice of parameters, ISBRAG converges to a region around the DRoNE set. We conclude this section by discussing the convergence bound in the following remark.

Remark 3.6.9. (*On the effect of parameters for ISBRAG*). From Theorem 3.6.8, it is clear that the agents' choices of $\{\alpha_i, \mu_i, \lambda_i\}_{i \in \mathcal{A}}$ affect not only the rate of convergence of ISBRAG (through β_1), but also the region to which it converges (through γ_1). This region can be made arbitrarily small due to the properties of γ_1 and ρ_1 in (3.25). From (3.31), it is easy to see that since $\underline{\alpha} = \min_{i \in \mathcal{A}} \alpha_i$, the rate of convergence depends on the agent that has the smallest step size. If this decreases, the rate of convergence decreases.

Moreover, $\underline{\alpha}$ affects the region to which ISBRAG converges as follows. From [54], we can characterize $\gamma_1 \in \mathcal{K}$ from Theorem 3.6.8 as

$$\gamma_1\left(K\rho(\bar{\alpha}, \bar{\mu})\right) = \sigma_1^{-1}\left(\frac{K}{\mathfrak{M}_{\underline{\alpha}}}\rho_1(\bar{\alpha}, \bar{\mu})\right), \quad (3.34)$$

where $\mathfrak{M}$ is any number in $(0, 1)$ and σ_1 comes from (3.26). Thus, using the relation in (3.33), agent $i \in \mathcal{A}$ can choose α_i and μ_i arbitrarily small and subsequently make $\gamma_1(K\rho_1(\bar{\alpha}, \bar{\mu}))$ arbitrarily small.

In light of this, it might be tempting to think that by making μ_i arbitrarily small,

agent $i \in \mathcal{A}$ diminishes the effect of its supergradient and is dominated by the inertial term (see Remark 3.6.5). However, due to the constraint in (3.24), reducing μ_i requires increasing λ_i which in turn requires decreasing $1/\lambda_i$ too. Hence the parameters μ_i and $1/\lambda_i$ work in tandem to counteract the effect of c_i and preserve supergradient information.

•

3.7 Distributed DRoNE Seeking

To implement the ISBRAG dynamics, each agent $i \in \mathcal{A}$ requires knowledge of $\mathbf{s}_{-i}(t)$, the current strategies of all other agents. In this section, we begin to adapt ISBRAG to a distributed communication network $\mathcal{G} = (\mathcal{A}, \mathcal{E})$ and solve Problem 3.4.8: (3) under Assumptions 3.4.5 and 3.4.6 on individual uncertainty. We tackle the problem in its full generality in Section 3.8.

In order to compensate for the lack of knowledge of others' strategies, we will allow the agents to estimate the same through a consensus-like protocol. This will induce an error (say $\Delta_i(t)$) in the computation of $\phi_i(t)$ (specifically in the computation of $\mathbf{v}_i(t)$ using (3.17)) for agent $i \in \mathcal{A}$, $\forall t \in \mathbb{Z}_{\geq 0}$. Let $\hat{\phi}(t)$, with $\hat{\phi}_i(t)$ components corresponding to $i \in \mathcal{A}$, denote the perturbed version of $\phi(t)$ at time $t \in \mathbb{Z}_{\geq 0}$. Then, we update the dynamics in (3.15) appropriately as, $\forall i \in \mathcal{A}$,

$$\hat{\phi}_i(t) = \mu_i [\mathbf{v}_i(\mathbf{s}(t)) + \Delta_i(t)] + \mathbf{w}_i(\mathbf{s}_i(t), t) \quad (3.35a)$$

$$= \phi_i(t) + \mu_i \Delta_i(t),$$

$$\hat{\pi}_i(t) = \pi_i^*(\hat{\phi}_i(t)) = \pi_i^*(\phi_i(t)) + \Delta'_i(t), \quad (3.35b)$$

$$\mathbf{s}_i(t+1) = \mathbf{s}_i(t) + \alpha_i [\hat{\pi}_i(t) - \mathbf{s}_i(t)] \quad (3.35c)$$

$$= \mathbf{s}_i(t) + \alpha_i [\pi_i^*(\phi_i(t)) - \mathbf{s}_i(t)] + \alpha_i \Delta'_i(t).$$

Here, $\Delta'_i(t)$ is the error in the strategy updates while $\Delta_i(t)$ is the error in the calculation of supergradients. Note that, if the agents estimate others' strategies and compute

$\widehat{\phi}_i(t)$ according to the estimated strategies, then, because of (3.17), $\mathbf{v}_i(\mathbf{s}(t)) + \Delta_i(t) \in \partial_{\mathbf{x}}[\overline{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i})]_{\mathbf{s}_i}$ and hence $\|\Delta_i(t)\| \leq 2B_i$, $\forall t \in \mathbb{Z}_{\geq 0}$. Moreover, because of (3.13), and (3.14), $\widehat{\pi}_i(t) = \pi_i^*(\phi_i(t)) + \Delta'_i(t) \in \mathcal{S}_i$. Hence, since $\pi_i^*(\phi_i(t)) \in \mathcal{S}_i$, $\|\Delta'_i(t)\| \leq D_i$. Thus, by properly choosing the $\{\alpha_i, \mu_i\}_{i \in \mathcal{A}}$ parameters, the effect of the disturbance can be mitigated accordingly. To show this formally, we first adapt Lemma 3.6.3 to account for the error induced by estimations.

Lemma 3.7.1. (Bound on Lyapunov function differences in the presence of disturbances). *Consider the dynamics (3.35c) with input sequence $\{\Delta(t), \Delta'(t)\}_{t \in \mathbb{Z}_{\geq 0}}$ from an initial condition $\{\mathbf{s}_i(0) \in \mathcal{S}_i\}_{i \in \mathcal{A}}$. Suppose $\{\phi(t)\}_{t \in \mathbb{Z}_{\geq 0}}$ satisfies $\|\phi_i(t)\| \leq M_i$ (for some $M_i \in \mathbb{R}_{>0}$), $\forall i \in \mathcal{A}$, $\forall t \in \mathbb{Z}_{\geq 0}$. Take any $\overline{D}_i > D_i$, $\forall i \in \mathcal{A}$. Denote $\overline{\alpha} := \max_{i \in \mathcal{A}} \alpha_i$, $\overline{\mu} := \max_{i \in \mathcal{A}} \mu_i$. Then, $\forall t \in \mathbb{Z}_{\geq 0}$, with $\underline{\alpha} = \min_{i \in \mathcal{A}} \alpha_i$,*

$$\begin{aligned} V(\mathbf{s}(t+1), \phi(t+1)) - V(\mathbf{s}(t), \phi(t)) \leq & \quad (3.36) \\ \sum_{i \in \mathcal{A}} \left[\frac{1 - \alpha_i}{\alpha_i} [\mathbf{s}_i(t+1) - \mathbf{s}_i(t)]^\top [\phi_i(t+1) - \phi_i(t)] \right. \\ & \left. + \overline{D}_i \|\phi_i(t+1) - \phi_i(t)\| \right] - \underline{\alpha} V(\mathbf{s}, \phi) + \rho_2(\overline{\alpha}, \overline{\mu}) \|\Delta'(t)\|. \end{aligned}$$

Here, the term $\rho_2(\overline{\alpha}, \overline{\mu})$ is characterized as

$$\rho_2(\overline{\alpha}, \overline{\mu}) = \max_{i \in \mathcal{A}} [A_i - B_i] \overline{\alpha} \overline{\mu} + \max_{i \in \mathcal{A}} A_i \overline{\mu}. \quad (3.37)$$

Further, A_i is as in (3.30) and B_i is as in Assumption 3.6.1.

Proof. As before, with a slight abuse of notation, we set $\mathbf{s} = \mathbf{s}(t)$, $\mathbf{s}^+ = \mathbf{s}(t+1)$, $\phi = \phi(t)$, $\phi^+ = \phi(t+1)$, $\widehat{\phi} = \widehat{\phi}(t)$, $\widehat{\phi}^+ = \widehat{\phi}(t+1)$, $\Delta = \Delta(t)$, and $\Delta' = \Delta'(t)$. Then, from

the definition of V in (3.12) and the dynamics in (3.35c), we have that

$$\begin{aligned}
& V(\mathbf{s}^+, \boldsymbol{\phi}^+) - V(\mathbf{s}, \boldsymbol{\phi}) \\
&= \sum_{i \in \mathcal{A}} \left[\max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i^+ - \max_{\mathbf{x}_i \in \mathcal{S}_i} \mathbf{x}_i^\top \boldsymbol{\phi}_i \right. \\
&\quad \left. - \mathbf{s}_i^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] - [\mathbf{s}_i^+ - \mathbf{s}_i]^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] - [\mathbf{s}_i^+ - \mathbf{s}_i]^\top \boldsymbol{\phi}_i \right] \\
&\leq \sum_{i \in \mathcal{A}} \left[\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i)^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] + \bar{D}_i \|\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i\| \right. \\
&\quad \left. - \mathbf{s}_i^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] - \alpha_i [\boldsymbol{\pi}_i^*(\hat{\boldsymbol{\phi}}_i) - \mathbf{s}_i]^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] \right. \\
&\quad \left. - \alpha_i [\boldsymbol{\pi}_i^*(\hat{\boldsymbol{\phi}}_i) - \mathbf{s}_i]^\top \boldsymbol{\phi}_i \right].
\end{aligned}$$

This chain of inequalities follow the same arguments as in the proof of Lemma 3.6.3 with the help of Lemma 3.6.2 and the dynamics (3.35) (specifically (3.35b) and (3.35c)). Now, by appropriately substituting in (3.35b) in the previous inequality, we get

$$\begin{aligned}
& V(\mathbf{s}^+, \boldsymbol{\phi}^+) - V(\mathbf{s}, \boldsymbol{\phi}) \\
&\leq \sum_{i \in \mathcal{A}} \left[[\boldsymbol{\pi}_i^*(\hat{\boldsymbol{\phi}}_i) - \boldsymbol{\Delta}'_i]^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] + \bar{D}_i \|\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i\| \right. \\
&\quad \left. - \mathbf{s}_i^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] - \alpha_i [\boldsymbol{\pi}_i^*(\hat{\boldsymbol{\phi}}_i) - \mathbf{s}_i]^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] \right. \\
&\quad \left. - \alpha_i [\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i) + \boldsymbol{\Delta}'_i - \mathbf{s}_i]^\top \boldsymbol{\phi}_i \right].
\end{aligned}$$

Now to upper bound this term, we use similar techniques as in the proof of Lemma 3.6.3. Specifically, notice that that $\sum_{i \in \mathcal{A}} [\boldsymbol{\pi}_i^*(\boldsymbol{\phi}_i) - \mathbf{s}_i]^\top \boldsymbol{\phi}_i = V(\mathbf{s}, \boldsymbol{\phi})$. Then by combining appropriate terms from the previous step, separating out the terms involving $\boldsymbol{\Delta}'_i$'s and further, using Cauchy-Schwarz inequalities we get $[-\boldsymbol{\Delta}'_i]^\top [\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i] \leq \|\boldsymbol{\Delta}'_i\| \|\boldsymbol{\phi}_i^+ - \boldsymbol{\phi}_i\|$ and

$[-\Delta'_i]^\top \phi_i \leq \|\Delta'_i\| \|\phi_i\|$. Putting all of this together gives us

$$\begin{aligned} & V(\mathbf{s}^+, \phi^+) - V(\mathbf{s}, \phi) \\ & \leq \sum_{i \in \mathcal{A}} \left[[1 - \alpha_i] [\pi_i^*(\hat{\phi}_i) - \mathbf{s}_i]^\top [\phi_i^+ - \phi_i] + \overline{D}_i \|\phi_i^+ - \phi_i\| \right] \\ & \quad + \sum_{i \in \mathcal{A}} \left[\|\Delta'_i\| \|\phi_i^+ - \phi_i\| + \alpha_i \|\Delta'_i\| \|\phi_i\| \right] - \underline{\alpha} V(\mathbf{s}, \phi). \end{aligned}$$

Finally, rearranging (3.35c) in the previous inequality gives us all the terms in (3.36) except the last term involving $\|\Delta'(t)\|$. To obtain the latter, we use the upper bounds from (3.30) and the discussion after (3.19); then, use similar arguments as in the proof of Theorem 3.6.8. This completes the proof. $\blacksquare$

Note that from (3.37), the gain ρ_2 in (3.36) can be made arbitrarily small by choosing arbitrarily small parameters $\{\alpha_i, \mu_i\}_{i \in \mathcal{A}}$. Thus, $\rho_2(\overline{\alpha}, \overline{\mu}) \|\Delta'(t)\|$ in (3.36) is a class- $\mathcal{K}$ function of $\|\Delta'(t)\|$. Hence, (3.36) is almost similar to the ISS requirement in (2.6). We can handle the extra terms like we did in the proof of Theorem 3.6.8 and show practical convergence when agents need to estimate others' strategies through some algorithm. We do this next.

Before proposing the distributed algorithm in full, we provide details about the dynamic consensus protocol that will be used to this end. In this manuscript, we adopt the protocol where the agents are allowed to communicate with each other during $T \in \mathbb{Z}_{>0}$ intermediate rounds to infer other agents' strategies. As discussed in [27], we employ a dynamic average consensus algorithm that produces a better tracking response than a

static consensus counterpart. The specific algorithm we use from [27] is detailed next,

$$v_i^c\left(t + \frac{\tau+1}{T}\right) = b_1 b_2 b_3 \sum_{j \in \mathcal{N}_i} a_{ij} \left[x_i^c\left(t + \frac{\tau}{T}\right) - x_j^c\left(t + \frac{\tau}{T}\right) \right] + v_i^c\left(t + \frac{\tau}{T}\right), \quad (3.38a)$$

$$z_i^c\left(t + \frac{\tau+1}{T}\right) = [1 - b_1 b_2] z_i^c\left(t + \frac{\tau}{T}\right) - b_1 v_i^c\left(t + \frac{\tau}{T}\right) - b_1 b_3 \sum_{j \in \mathcal{N}_i} a_{ij} \left[x_i^c\left(t + \frac{\tau}{T}\right) - x_j^c\left(t + \frac{\tau}{T}\right) \right], \quad (3.38b)$$

$$x_i^c\left(t + \frac{\tau}{T}\right) = z_i^c\left(t + \frac{\tau}{T}\right) + u_i\left(t + \frac{\tau}{T}\right). \quad (3.38c)$$

Here, $\{x_i^c, v_i^c, z_i^c\}_{i \in \mathcal{A}}$ are internal variables of the dynamics and $b_1, b_2, b_3 \in \mathbb{R}_{>0}$ are constants that can be chosen appropriately to ensure that (3.38) asymptotically tracks $[1/n] \sum_{i \in \mathcal{A}} u_i(t)$ with a small error. This error depends on the aforementioned constants b_1, b_2, b_3 , the eigenvalues of the graph Laplacian associated with $\mathcal{G}$, and (most importantly) an upper bound on $|u_i(t + [\tau + 1]/T) - u_i(t + \tau/T)|$. We refer the reader to [27, Theorem S2] for details on how to tune the parameters of (3.38) and minimize the error.

Now, in order to make the notations cleaner, we encapsulate the states in (3.38) into $\chi_i := [x_i^c, v_i^c, z_i^c]^\top$ for each agent $i \in \mathcal{A}$. This allows us to rewrite (3.38) as

$$\chi_i\left(t + \frac{\tau+1}{T}\right) = \mathbf{f}_i\left(\left\{\chi_j\left(t + \frac{\tau}{T}\right)\right\}_{j \in \mathcal{N}_i}, u_i\left(t + \frac{\tau}{T}\right)\right). \quad (3.39)$$

Observe that (3.39) is a fully distributed algorithm that requires agents to only pass information between neighbors. Moreover, for the sake of brevity, let $\Pi_x(\chi_i)$ denote the x_i^c component of χ_i . Now, for agent $i \in \mathcal{A}$ to approximate $\mathbf{s}_{-i}$, it has to run multiple copies of (3.39) across the components of $\mathbf{s}_{-i}$. To that end, let $\widehat{\mathbf{s}}_{ij} \in \mathbb{R}^{n_j}$ (with components $\widehat{s}_{ij,l}$) be agent i 's estimate of $\mathbf{s}_j$. We collect this into a stacked vector called $\widehat{\mathbf{s}}_{-i}$ for each $i \in \mathcal{A}$. Similarly, let χ_{ij}^l denote the dynamic consensus vector associated with $\widehat{s}_{ij,l}$. Combining all of this, we present the distributed algorithm that partially solves Problem 3.4.8: (3) next and discuss it in the remark that follows.

Definition 3.7.2. (*d-ISBRAG*). We refer to the following algorithm as *Distributed Inertial Supported Better Response Ascending superGradient dynamics* or *d-ISBRAG*.

$$\mathbf{s}_i(t+1) = \mathbf{s}_i(t) + \alpha_i \left[\boldsymbol{\pi}_i^* \left(\widehat{\boldsymbol{\phi}}_i \left(t + \frac{T-1}{T} \right) \right) - \mathbf{s}_i(t) \right], \quad (3.40a)$$

$$\widehat{\boldsymbol{\phi}}_i \left(t + \frac{\tau}{T} \right) = \mu_i \mathbf{v}_i \left(\mathbf{s}_i(t), \widehat{\mathbf{s}}_{-i} \left(t + \frac{\tau}{T} \right) \right) + \mathbf{w}_i(\mathbf{s}_i(t), t), \quad (3.40b)$$

$$\boldsymbol{\chi}_{ij}^l \left(t + \frac{\tau+1}{T} \right) = \mathbf{f}_i \left(\left\{ \boldsymbol{\chi}_{kj}^l \left(t + \frac{\tau}{T} \right) \right\}_{k \in \mathcal{N}_i}, 0 \right), \quad (3.40c)$$

$$\boldsymbol{\chi}_{ii}^l \left(t + \frac{\tau+1}{T} \right) = \mathbf{f}_i \left(\left\{ \boldsymbol{\chi}_{ki}^l \left(t + \frac{\tau}{T} \right) \right\}_{k \in \mathcal{N}_i}, n s_{i,l}(t) \right), \quad (3.40d)$$

$$\widehat{s}_{ij,l} \left(t + \frac{\tau}{T} \right) = \Pi_x \left(\boldsymbol{\chi}_{ij}^l \left(t + \frac{\tau}{T} \right) \right). \quad (3.40e)$$

The previous equation holds $\forall i, j \in \mathcal{A}, \forall t, \tau \in \mathbb{Z}_{\geq 0}$. The index l appropriately belongs to $\{1, \dots, n_i\}$. The $\mathbf{v}_i$ in (3.40b) is computed similarly as in (3.17) but with U_i^{DR} instead of $\overline{U}_i^{\text{DR}}$. •

Remark 3.7.3. (*On d-ISBRAG*). From (3.40), it is easy to notice the two time-scale approach in which d-ISBRAG proceeds. Each agent updates its strategy using (3.40a) at every $t \in \mathbb{Z}_{>0}$; while every agent runs the dynamic consensus protocol using (3.40c)-(3.40e) for T sub (steps) $\{t, t+1/T, \dots, t+[T-1]/T\}$. Moreover, agent $i \in \mathcal{A}$ performs the update in (3.40a) at $t+1$ by computing $\widehat{\boldsymbol{\phi}}_i$ using the most recent estimate of $\widehat{\mathbf{s}}_{-i}$ at $t+[T-1]/T$ (compare (3.40a) and (3.40b)).

Recall that $s_{i,l}$ is the l^{th} component of $\mathbf{s}_i$. Hence, regarding the consensus update, notice from (3.40c) and (3.40d) that agent $i \in \mathcal{A}$ contributes $n s_{i,l}$ to $\boldsymbol{\chi}_{ii}^l$ and 0 to all other $\boldsymbol{\chi}_{ij}^l$, $j \in \mathcal{A} \setminus \{i\}$. This in fact ensures that $\boldsymbol{\chi}_{ji}^l(t)$ asymptotically tracks $s_{i,l}(t)$, $\forall i, j \in \mathcal{A}$. •

We conclude this section by providing convergence guarantees for d-ISBRAG.

Theorem 3.7.4. (Convergence of d-ISBRAG). *Suppose Assumptions 3.4.5, 3.4.6, 3.4.7, and 3.6.1 hold. Moreover, suppose $\mathcal{G}_{\text{DR}}$ has amicable supergradients with factor of amicability $\{c_i\}_{i \in \mathcal{A}}$. With $d_{\min} := \min_{i \in \mathcal{A}} d_i$, choose $\mu_i \in (0, \infty)$, $\forall i \in \mathcal{A}$ and suppose $\forall i \in \mathcal{A}$,*

α_i satisfies (3.23) and λ_i satisfies (3.24). Define the size function $\omega(\cdot) := d(\cdot, \mathcal{NE}(\mathcal{G}_{\text{DR}}))$ as the distance from the NE set of $\mathcal{G}_{\text{DR}}$. Denote $\bar{\alpha} := \max_{i \in \mathcal{A}} \alpha_i$, $\bar{\mu} := \max_{i \in \mathcal{A}} \mu_i$. Suppose for (3.38), $v_i^c(0) = 0$, $z_i^c(0) \in \mathbb{R}$, $\forall i \in \mathcal{A}$. Further, let K_i be as in (3.32) with $K = \max_{i \in \mathcal{A}} nK_i$; ρ_1 be as in (3.33), and ρ_2 be as in (3.37). Finally, let $\mathbf{s}(t)$ be the solution to the dynamics in Definition 3.7.2 from initial condition $\{\mathbf{s}_i(0) = \mathbf{s}_i(-1) \in \mathcal{S}_i\}_{i \in \mathcal{A}}$, with appropriately chosen parameters b_1, b_2, b_3 , and T .

Then, there exists functions $\beta_2 \in \mathcal{KL}$, $\gamma_2 \in \mathcal{K}$ such that

$$\omega(\mathbf{s}(t)) \leq \max \left\{ \beta_2 \left(\omega(\mathbf{s}(0)), t \right), \right. \\ \left. \gamma_2 \left(K \rho_1(\bar{\alpha}, \bar{\mu}) + J \rho_2(\bar{\alpha}, \bar{\mu}) \right) \right\}, \quad (3.41)$$

where, J is such that $\max_{\mathbf{x} \in \mathcal{S}} \|\mathbf{x}\| \leq J$.

Proof. The proof follows the exact same arguments as in the proof of Theorem 3.6.8 with help from Lemma 3.7.1 instead of Lemma 3.6.3. For the final bound, notice from (3.35b) that $\|\Delta'\| \leq J$. This completes the proof. $\blacksquare$

3.8 When Samples are Shared

In this section, we solve Problem 3.4.8: (3) in its full generality, *i.e.* under Assumption 3.4.3 of shared samples. Now, the agents need to share information over the communication network $\mathcal{G}$ in order to estimate its supergradient directions. This is in contrast to the setup in Section 3.7, where every agent was capable of computing its own supergradients as long as it had access to other's strategies. The next result, which is a generalization of Danskin's theorem [55], gives a particular way of computing the supergradients for a class of concave functions.

Lemma 3.8.1. (Supergradient of the min of concave functions). *Let $\{f_i : \mathcal{D} \rightarrow \mathbb{R}\}_{i \in \mathcal{I}}$, (where $\mathcal{D} \subseteq \mathbb{R}^d$ is a convex set) be a set of concave functions. Define the concave function*

$g : \mathcal{D} \rightarrow \mathbb{R}$ as

$$g(\mathbf{x}) := \min_{i \in \mathcal{I}} f_i(\mathbf{x}).$$

Suppose for $\mathbf{x} \in \mathcal{D}$, $i^* \in \operatorname{argmin}_{i \in \mathcal{I}} f_i(\mathbf{x})$. Then,

$$\partial f_{i^*}(\mathbf{x}) \subseteq \partial g(\mathbf{x}). \quad (3.42)$$

Further, let f_i be differentiable $\forall i \in \mathcal{I}$, consider an $\mathbf{x} \in \mathcal{D}$, and let $\nabla f_i(\mathbf{x})$ be the corresponding derivative. Then, $\exists \mathcal{J} \subseteq \operatorname{argmin}_{i \in \mathcal{I}} f_i(\mathbf{x})$, with $|\mathcal{J}| = d + 1$, such that $\forall \zeta \in \partial g(\mathbf{x})$, $\zeta = \sum_{i \in \mathcal{J}} \lambda_i \nabla f_i(\mathbf{x})$, where $\lambda_i \geq 0$, $\forall i \in \mathcal{J}$ and $\sum_{i \in \mathcal{J}} \lambda_i = 1$.

Proof. We first prove the first claim. Let $\mathbf{x} \in \mathcal{D}$, $i^* \in \mathcal{I}$ be as in the hypothesis and let $\zeta \in \partial f_{i^*}(\mathbf{x})$. Then, $\forall \mathbf{y} \in \mathcal{S}$,

$$g(\mathbf{y}) \leq f_{i^*}(\mathbf{y}) \leq f_{i^*}(\mathbf{x}) + \zeta^\top [\mathbf{y} - \mathbf{x}] = g(\mathbf{x}) + \zeta^\top [\mathbf{y} - \mathbf{x}].$$

Here, the first inequality and the equality comes from the definition of g and the assumption on i^* . The second inequality comes from Definition 2.4.1. This proves (3.42).

Now, for the second part, by property of the supergradients and the hypothesis, $\partial g(\mathbf{x}) = \operatorname{co}(\{\nabla f_i(\mathbf{x})\}_{i \in \mathcal{I}})$, where $\operatorname{co}(\cdot)$ represents the convex hull. Then, the second part follows directly from Carathéodory's theorem from convex geometry. The proof is now complete. ■

Thus, based on Lemma 3.8.1, we argue that, in order to compute this supergradient, it is necessary to solve the inner optimization problem of the utility definition in (3.3). This is challenging since the optimization problem in (3.3) is infinite dimensional. In order to reformulate it into a tractable program, we use the method proposed in [2]. First, we make the following assumption regarding the effect of the random variable on an agent's

utility.

Assumption 3.8.2. (Utility is differentiable in own strategy and convex in the random variable). *For each $i \in \mathcal{A}$,*

1. $U_i(\cdot, \mathbf{s}_{-i}; \xi)$ is differentiable $\forall \mathbf{s}_{-i}, \xi$, and
2. $U_i(\mathbf{s}; \cdot)$ is convex and differentiable $\forall \mathbf{s}$.

Moreover, Ξ is convex and closed. •

Now, it is possible to use the very same arguments as in [2] to show that the utility defined in (3.3) can be rewritten as the solution of the following finite dimensional convex program,

$$\bar{U}_i^{\text{DR}}(\mathbf{s}) = \min_{\{\mathbf{x}_i^k \in \mathbb{R}^r\}_{k=1}^N} \frac{1}{N} \sum_{k=1}^N U_i(\mathbf{s}; \mathbf{h}(\hat{\xi}^{(k)}) - \mathbf{x}_i^k) \quad (3.43a)$$

$$\text{s.t. } \sum_{k=1}^N \|\mathbf{x}_i^k\|_1 \leq N\varepsilon_i; \quad (3.43b)$$

$$\hat{\xi}^{(k)} - \mathbf{x}_i^k \in \mathbf{h}(\Xi), \quad \forall k \in \{1, \dots, N\}. \quad (3.43c)$$

Note that even though this optimization problem (utility structure and decision variables) is local to each agent; as per Assumption 3.4.3, agent $i \in \mathcal{A}$ is only aware of $\{\mathbf{h}_i(\hat{\xi}^{(1)}), \dots, \mathbf{h}_i(\hat{\xi}^{(N)})\}$.

To address this, we make a few simplifying assumptions. First, we simplify the notations by letting $\hat{\mathbf{h}}^k := \mathbf{h}(\hat{\xi}^{(k)})$, $\forall k \in \{1, \dots, N\}$. Next, we limit Ξ by imposing box constraints on the decision variables as in the following.

Assumption 3.8.3. (Known bounds). *The uncertainty set Ξ is such that $\mathbf{h}(\Xi) = \{\xi \in \mathbb{R}^m \mid \underline{\mathbf{b}} \leq \xi \leq \bar{\mathbf{b}}\}$. The inequalities here are taken term-wise. Moreover, every agent knows $\underline{\mathbf{b}}$ and $\bar{\mathbf{b}}$.* •

Note that this assumption on global bounds on the uncertainty could be relaxed to agent's knowledge of component-wise bounds. If this was not the case, agents can gather this knowledge via some max consensus algorithm routine. Then, notice that since (3.43) is local to each agent $i \in \mathcal{A}$, the optimization problem remains unchanged by summing (3.43a) (and combining all the constraints in (3.43b) and (3.43c)) across all the agents. In what follows, we order the information available to agent $i \in \mathcal{A}$ as $[\widehat{h}_{p_i}^k, \dots, \widehat{h}_{q_i}^k]$, $\underline{\mathbf{b}}$, and $\overline{\mathbf{b}}$. Here, we choose appropriate $p_i \leq q_i \in \mathbb{Z}_{>0}$ to represent the starting and end indices for $i \in \mathcal{A}$, which results into

$$\widehat{\mathbf{h}}^k = [\dots, \widehat{h}_{p_i}^k, \dots, \widehat{h}_{q_i}^k, \dots]^\top, \quad \forall k \in \{1, \dots, m\}.$$

Then, incorporating all of these, we use the following optimization problem to compute the supergradients,

$$\begin{aligned} & \underset{\{\mathbf{y}_{ji}^k, z_{ji}\}_{i,j \in \mathcal{A}}}_{k=1}^N \text{ minimize } \frac{1}{N} \sum_{i \in \mathcal{A}} \sum_{k=1}^N U_i(\mathbf{s}_i, \mathbf{s}_{-i}; \mathbf{y}_{ii}^k), \text{ s.t.} \end{aligned} \quad (3.44a)$$

$$\sum_{k=1}^N \sum_{l=p_i}^{q_i} |\widehat{h}_l^k - y_{ii,l}^k| + \sum_{r \in \mathcal{N}_i} [z_{ii} - z_{ir}] \leq N\varepsilon_i, \quad \forall i \in \mathcal{A}; \quad (3.44b)$$

$$\sum_{k=1}^N \sum_{l=p_i}^{q_i} |\widehat{h}_l^k - y_{ji,l}^k| + \sum_{r \in \mathcal{N}_i} [z_{ji} - z_{jr}] \leq 0, \quad \forall i, j \in \mathcal{A}; \quad (3.44c)$$

$$\mathbf{y}_{ji}^k - \sum_{r \in \mathcal{N}_i} [\mathbf{y}_{jr}^k - \mathbf{y}_{ji}^k] = \mathbf{0}, \quad \forall i, j \in \mathcal{A}, \forall k \in \{1, \dots, N\}; \quad (3.44d)$$

$$\underline{\mathbf{b}} \leq \mathbf{y}_{ji}^k \leq \overline{\mathbf{b}}, \quad \forall k \in \{1, \dots, N\}, \quad \forall i, j \in \mathcal{A}. \quad (3.44e)$$

Here, $\widehat{h}_l^k$ (resp. $y_{ij,l}^k$) are the components of $\widehat{\mathbf{h}}^k$ (resp. $\mathbf{y}_{ij}^k$). The variable $\mathbf{y}_{ij}^k$ (resp. z_{ij}) can be thought of as agent j 's copy of $\mathbf{y}_i^k$ (similarly slack variable for i 's constraint). Then notice that (3.44) can be solved completely using local information because of Assumption 3.8.3. We state and prove our previous claim regarding the supergradients in

the next result.

Lemma 3.8.4. (Distributed optimization problem produces the required supergradients).

Suppose Assumptions 3.4.7, 3.8.2, and 3.8.3 hold; and let $\{\{\mathbf{y}_{ji}^{k*}, z_{ji}^*\}_{i,j \in \mathcal{A}}\}_{k=1}^N$ be a solution to (3.44). Then, with $\bar{\mathbf{y}}_i^* = \frac{1}{N} \sum_{k=1}^N \mathbf{y}_{ii}^{k*}$, $\forall i \in \mathcal{A}$; $\nabla_{\mathbf{x}}[U_i(\mathbf{x}, \mathbf{s}_{-i}; \bar{\mathbf{y}}_i^*)]|_{\mathbf{s}_i} \in \partial_{\mathbf{x}}[\bar{U}_i^{\text{DR}}(\mathbf{x}, \mathbf{s}_{-i})]|_{\mathbf{s}_i}$, $\forall i \in \mathcal{A}$.

Proof. First note that the description of $\bar{U}_i^{\text{DR}}$ in (3.43) and hence the equality between the right hand sides of (3.3) and (3.43) follows from Assumption 3.8.2 and [2]. Next it is easy to see that the optimization problem in (3.43) (combined across all agents) is equivalent to the following problem,

$$\underset{\{\{\mathbf{y}_i^k\}_{i \in \mathcal{A}}\}_{k=1}^N}{\text{minimize}} \quad \frac{1}{N} \sum_{i \in \mathcal{A}} \sum_{k=1}^N U_i(\mathbf{s}_i, \mathbf{s}_{-i}; \mathbf{y}_i^k) \quad (3.45a)$$

$$\text{s.t.} \quad \sum_{k=1}^N \sum_{l=1}^m |\hat{h}_l^k - y_{i,l}^k| \leq N\varepsilon_i, \quad \forall i \in \mathcal{A} \quad (3.45b)$$

$$\underline{\mathbf{b}} \leq \mathbf{y}_i^k \leq \bar{\mathbf{b}}, \quad \forall k \in \{1, \dots, N\}, \quad \forall i \in \mathcal{A}. \quad (3.45c)$$

This is because, we have used the auxiliary variables $\mathbf{y}_i^k$ to satisfy the equality constraint $\mathbf{y}_i^k = \hat{\mathbf{h}}^k - \mathbf{x}_i^k$, $\forall i, k$. Then, (3.45b) is the same as (3.43b) due to the definition of $\|\cdot\|_1$. Moreover, (3.45c) is the same as (3.43c) due to Assumption 3.8.3. Thus, optimization problems (3.43) and (3.45) are equivalent.

Next we introduce auxiliary variables to show that (3.45) and (3.44) are equivalent. To do this, we first label the feasible sets of (3.45) and (3.44) $\mathcal{F}$ and $\bar{\mathcal{F}}$ respectively; *i.e.* $\{\{\mathbf{y}_i^k\}_{i \in \mathcal{A}}\}_{k=1}^N \in \mathcal{F}$ and $\{\{\mathbf{y}_{ji}^k, z_{ji}\}_{i,j \in \mathcal{A}}\}_{k=1}^N \in \bar{\mathcal{F}}$. Now, since the cost functions in (3.45a) and (3.44a) are the same, it is enough to show the following two claims to show equivalence.

Claim i: $\forall \{\{\mathbf{y}_{ji}^k, z_{ji}\}_{i,j \in \mathcal{A}}\}_{k=1}^N \in \bar{\mathcal{F}}, \{\{\mathbf{y}_{ii}^k\}_{i \in \mathcal{A}}\}_{k=1}^N \in \mathcal{F}$.

To show this, consider a $\{\{\mathbf{y}_{ji}^k, z_{ji}\}_{i,j \in \mathcal{A}}\}_{k=1}^N \in \bar{\mathcal{F}}$. Since $\{\{\mathbf{y}_{ii}^k\}_{i \in \mathcal{A}}\}_{k=1}^N$ satisfies (3.44e), it also satisfies (3.45c). Moreover, because of (3.44d), and Assumption 3.4.7 on graph

connectivity, $\mathbf{y}_{ii}^k = \mathbf{y}_{ij}^k, \forall i, j \in \mathcal{A}$. We use Assumption 3.4.7 in the next argument also. Summing (3.44b) for $i \in \mathcal{A}$ and (3.44c) $\forall j \in \mathcal{A} \setminus \{i\}$ shows that $\{\{\mathbf{y}_{ii}^k\}_{i \in \mathcal{A}}\}_{k=1}^N$ satisfies (3.45b) because of the previous equality. This concludes the proof of this claim.

Claim ii: $\forall \{\{\mathbf{y}_i^k\}_{i \in \mathcal{A}}\}_{k=1}^N \in \mathcal{F}, \exists \{z_{ji}\}_{i,j \in \mathcal{A}}$ such that $\{\{\mathbf{y}_{ji}^k, z_{ji}\}_{i,j \in \mathcal{A}}\}_{k=1}^N \in \overline{\mathcal{F}}$ with $\mathbf{y}_i^k = \mathbf{y}_{ij}^k, \forall i, j \in \mathcal{A}$.

To show this, consider a $\{\{\mathbf{y}_i^k\}_{i \in \mathcal{A}}\}_{k=1}^N \in \mathcal{F}$ and take vectors $\{\{\mathbf{y}_{ji}^k\}_{i,j \in \mathcal{A}}\}_{k=1}^N$ such that $\mathbf{y}_i^k = \mathbf{y}_{ij}^k, \forall i, j \in \mathcal{A}$. Then, because of (3.45c) and Assumption 3.4.7, by construction, $\{\{\mathbf{y}_{ji}^k\}_{i,j \in \mathcal{A}}\}_{k=1}^N$ satisfies (3.44d) and (3.44e). Then the claim is easy to show because of the connectivity assumption. This concludes the proof of this other claim.

Combining these concludes that (3.45) and (3.44) are equivalent. Then the proof of the claim in this lemma follows from Lemma 3.8.1. The proof is now complete. $\blacksquare$

We would like to point out that even though the technique we used in the previous proof is similar to [56]; our setup is very different, since the agents do not have access to the cost function and constraints simultaneously. We would like to remind the reader that (3.44) is a convex program that can be solved entirely using local information. As such, agent $i \in \mathcal{A}$ handles the variables $\{\{\mathbf{y}_{ji}^k, z_{ji}\}_{j \in \mathcal{A}}\}_{k=1}^N$. There is a whole body of literature that provides iterative solutions to (3.44). In this work, we use the primal-dual subgradient method [53] using the augmented Lagrangian approach to compute the required supergradients for the d-ISBRAG updates. While it is possible to provide ISS properties for the distributed updates and perform the updates of such an algorithm in a synchronous manner with (3.40) we reserve that analysis for future work. To that extent, we condense the d-ISBRAG algorithm for shared samples into Algorithm 1. We formally state the convergence properties of this algorithm in the next result and conclude by remarking on the communication protocol later. Since this is a direct consequence of all the previous results, we skip a formal proof.

Theorem 3.8.5. (Convergence of d-ISBRAG with shared samples). *Suppose the hypothesis*

Algorithm 1. d-ISBRAG for shared samples

Require: Consensus time steps T_{con} , optimization time steps T_{opt}

- 1: Initialize $\mathbf{s}_i \leftarrow \mathbf{s}_i(0) \in \mathcal{S}_i, \forall i \in \mathcal{A}$
 - 2: Initialize $\hat{\mathbf{s}}_{ij} \leftarrow \text{random}(\mathcal{S}_j), \forall i \in \mathcal{A}, \forall j \in \mathcal{A} \setminus \{i\}$
 $\triangleright$ randomly initialize estimates
 - 3: **for** $t \in \mathbb{Z}_{\geq 0}$ **do**
 - 4: **for** $\tau \in 1, \dots, T_{\text{con}}$ **do**
 - 5: Update $\{\hat{\mathbf{s}}_{ij}\}_{j \in \mathcal{A} \setminus \{i\}}$ using steps (3.40c), (3.40d) of dynamic consensus
 - 6: **end for**
 - 7: **for** $t' \in 1, \dots, T_{\text{opt}}$ **do**
 - 8: Compute $\{\{\mathbf{y}_{ji}^{k*}, z_{ji}^*\}_{i,j \in \mathcal{A}}\}_{k=1}^N$ that is a solution to (3.44) in a distributed way
 - 9: **end for**
 - 10: **for** $i \in \mathcal{A}$ **do**
 - 11: $\bar{\mathbf{y}}_i^* \leftarrow \frac{1}{N} \sum_{k=1}^N \mathbf{y}_{ii}^{k*}$
 - 12: $\hat{\phi}_i \leftarrow \mu_i \nabla_{\mathbf{x}} [U_i(\mathbf{x}, \mathbf{s}_{-i}; \bar{\mathbf{y}}_i^*)] \big|_{\mathbf{s}_i} + \mathbf{w}_i(\mathbf{s}_i, t)$
 - 13: $\mathbf{s}_i \leftarrow \mathbf{s}_i + \alpha_i \left[\boldsymbol{\pi}_i^*(\hat{\phi}_i) - \mathbf{s}_i \right]$
 - 14: **end for**
 - 15: **end for**
-

of Theorem 3.7.4 holds and suppose $\mathbf{s}(t)$ is the sequence obtained from Algorithm 1 from initial condition $\{\mathbf{s}_i(0) = \mathbf{s}_i(-1) \in \mathcal{S}_i\}_{i \in \mathcal{A}}$. Then, the bound in (3.41) holds with $\omega(\cdot) := d(\cdot, \mathcal{NE}(\bar{\mathcal{G}}_{\text{DR}}))$. ■

3.9 Simulations

In this section, we illustrate the trajectory evolution of the algorithms that we proposed. For all our simulations we take 6 agents, *i.e.* $\mathcal{A} = \{1, \dots, 6\}$. All values are rounded off to 3 decimal places.

3.9.1 Effect of step-size α_i :

First we study the effect of $\{\alpha_i\}_{i \in \mathcal{A}}$ on the convergence of ISBRAG in Definition 3.7.2. We specifically want to show that our algorithm is able to converge near a NE even when the utilities are non-smooth. To that affect, we choose $\mathcal{S}_i = [0, 2], \forall i \in \mathcal{A}$. Then, we set

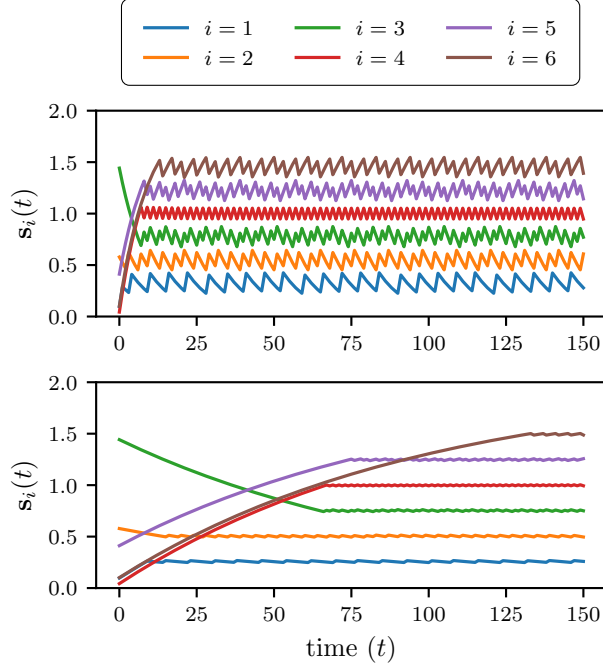


Figure 3.1. Effect of step-size parameters $\{\alpha_i\}_{i \in \mathcal{A}}$ on convergence of ISBRAG. The plots share a common legend. (Top) $\alpha_i = 0.1, \forall i \in \mathcal{A}$. (Bottom) $\alpha_i = 0.01, \forall i \in \mathcal{A}$.

the utilities of the agents as

$$U_i^{\text{DR}}(s_i, s_{-i}) = -|s_i - 0.25 \times i| \times \prod_{j \in \mathcal{A} \setminus \{i\}} s_j, \quad \forall i \in \mathcal{A}.$$

Note that we have denoted the strategies as $s_i \in \mathbb{R}$. Then clearly the strategy profile $s_1^* = 0.25$, $s_2^* = 0.5$, $s_3^* = 0.75$, $s_4^* = 1$, $s_5^* = 1.25$, and $s_6^* = 1.5$ is the unique NE. We simulate ISBRAG from an initial condition $s_1(0) = 0.097$, $s_2(0) = 0.578$, $s_3(0) = 1.442$, $s_4(0) = 0.043$, $s_5(0) = 0.412$, and $s_6(0) = 0.101$. Moreover, to isolate the effect of the step-size parameter, we fix $\mu_i = 0.5$, and $\lambda_i = 1, \forall i \in \mathcal{A}$. The solution trajectories for two cases: i) $\alpha_i = 0.1, \forall i \in \mathcal{A}$, and ii) $\alpha_i = 0.01, \forall i \in \mathcal{A}$ are shown in Figure 3.1. In accordance to the bound provided in Theorem 3.6.8, notice that with the higher the α_i value, the solutions converge faster, but they converge to a larger set around the NE. On the other hand, with a smaller α_i value, the solutions converge closer to the NE, but take more time-steps to get there.

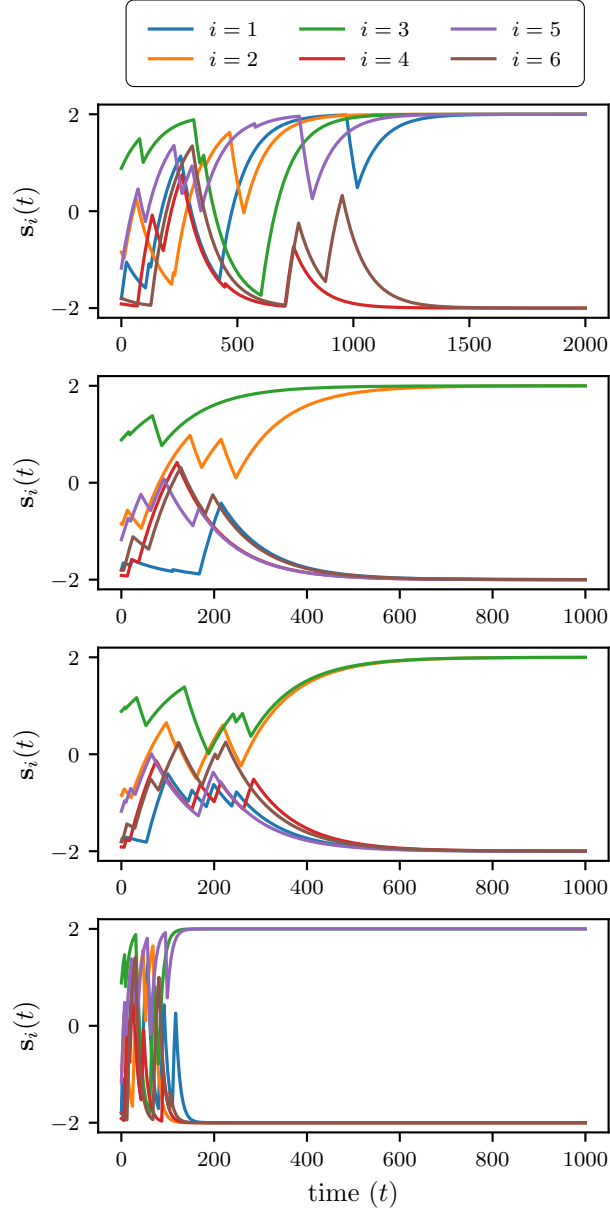


Figure 3.2. Convergence properties of d-ISBRAG. The plots share a common legend. Different plots correspond to different values of communication time steps T_{con} and step-size parameters $\{\alpha_i\}_{i \in \mathcal{A}}$. The figures from the top are arranged as follows. (First) $T_{\text{con}} = 10$, $\alpha_i = 0.01$, $\forall i \in \mathcal{A}$. (Second) $T_{\text{con}} = 50$, $\alpha_i = 0.01$, $\forall i \in \mathcal{A}$. (Third) $T_{\text{con}} = 100$, $\alpha_i = 0.01$, $\forall i \in \mathcal{A}$. (Fourth) $T_{\text{con}} = 100$, $\alpha_i = 0.1$, $\forall i \in \mathcal{A}$.

3.9.2 Performance of d-ISBRAG:

For this scenario, we test our algorithm on a non-monotone game and report (empirically) on how the parameters of the game affect the steady-state. Further, we allow the agents to communicate over a cyclic graph only, *i.e.* $\mathcal{E} = \{(1, 2), (2, 3), (3, 4), (4, 5), (5, 6), (6, 1)\}$. We choose $\mathcal{S}_i = [-2, 2]$, $\forall i \in \mathcal{A}$ and set the utilities of the agents as

$$\bar{U}_i^{\text{DR}}(s_i, s_{-i}) = \prod_{j \in \mathcal{A}} s_j, \quad \forall i \in \mathcal{A}.$$

It is easy to see that the Nash equilibrium is not unique. Specifically, any NE (other than $\mathbf{s}^* = \mathbf{0}$) must have $s_i^* \in \{-2, 2\}$, and there must be even number of agents with strategy -2 . The evolutions of d-ISBRAG, from an initial strategy profile of $s_1(0) = -1.805$, $s_2(0) = -0.844$, $s_3(0) = 0.884$, $s_4(0) = -1.913$, $s_5(0) = -1.176$, and $s_6(0) = -1.797$, with different choices of T_{con} and $\{\alpha_i\}_{i \in \mathcal{A}}$ are shown in Figure 3.2. Note that, the strategy we converge to is dependent on several of the algorithm parameters. Firstly, the first three figures from the top have same step-size parameters and differ in the number of consensus steps (T_{con}). As expected, the time steps needed to converge decreases as T_{con} increases. The last two graphs in Figure 3.2 share the same value of T_{con} and differ in the step-size parameters. The larger the α_i , the faster the convergence, but at the cost of high chattering-like behavior in the transients; while the smaller the α_i the longer it takes to converge with a reduced chattering in the transients. In any case, these trajectories do not sustain limiting oscillatory behavior (as opposed to the trajectories in Figure 3.1) since the utilities are smooth and the game admits proper pseudogradients.

3.10 Summary

In this work, we provide centralized and distributed algorithms that seek the Nash equilibria of a distributionally robust game (DRoNE's). First, we present conditions under

which these DRoNE's exist, and relate them to the NE of the associated stochastic game. This relation (which depends on the number of sample points available to the agents) predicts that as the number of data points increase, with high probability DRoNE's become better approximations of the NE's of the stochastic game. Second, we define and provide conditions on the parameters of the ISBRAG dynamics that ensure convergence to arbitrarily small regions around the DRoNE's. This comes at a cost of slower rate of convergence. To handle the distributed case, we provide methods in which the agents can approximate the supergradients of their (possibly) non-smooth utility by exchanging information between their neighbors. When the game has amicable supergradients, we introduce an inertial term that produces monotonicity-like behaviors. When the game has non-unique NE, the region around which DRoNE the algorithm converges to is highly dependent on the choice of algorithm parameters. In the future, we would like to extend the class of games compatible with this framework by introducing more general inertial terms. We will also handle simultaneous communication and estimate update among agents.

This chapter is a reprint of material that has been submitted for publication as “Distributionally Robust Nash Equilibrium Seeking with Partial Observations and Distributed Communication,” N. Mandal, and S. Martínez. The dissertation author was the primary investigator and author of this paper.

Part II

Application of Proposed Framework

Chapter 4

Task Allocation Among Self-Interested Agents

4.1 Introduction

A prototypical coordination problem aims to find an efficient assignment of group of agents to a collection of tasks. The tasks can range from abstract objectives to specific physical jobs, the nature of which may not be known. The agents may have heterogeneous capabilities, and react to different sets of incentives that are learned progressively. This necessitates of novel task-assignment algorithms that can adapt and react online as new information arises. Motivated by this, we study a discrete task allocation problem modeled as a game of self-interested agents with partial knowledge of their rewards. This requires addressing the problem's combinatorial nature, and designing provable-correct distributed dynamics that adapt to dynamic rewards revealed online. To the best of our knowledge, algorithms that combine all these features are not available in literature.

4.2 Literature review

The problem of task allocation in Cooperative Control with known rewards has been widely considered; see e.g. [57, 58, 59]. A centralized solution to this problem, where the number m of tasks and agents are equal and a task-agent matching is sought, is the

optimization-based Hungarian algorithm [60], and its distributed version [61]. The latter, which reproduces the Hungarian algorithm locally, requires tracking of the agents' identities associated with each task, has a time complexity of $O(m^3)$ and communication cost of $O(m \log m)$ (per communication round). Thus, the algorithm can be computationally and memory-intensive for large problems, and hard to adapt as new tasks are generated or their valuations change online. The work in [62] provides a tractable, sub-optimal solution to the same NP-hard problem, while the research in [63] showed that the sub-optimality can be resolved by restricting heterogeneous agents to be of certain types. In the same vein, the works in [64, 65, 66] consider submodular functions which allow rewards to take any non-negative value. However, submodular optimization can be applied in specific domains where the property naturally arises, such as in certain economics and distributed sensing problems. Alternatively, a well known approach to (unconstrained) task assignment problems is given by k -means clustering and the Lloyd's algorithm [67]. By interpreting that tasks are generated by a probability distribution, the approach can handle tasks generated dynamically [68, 69, 70, 71]. However, Lloyd's algorithm is sensitive to the initial task assignment for a small number of agents, and in general converges to local critical points.

We note that the related body of work in Operations Research [72, 73] and Economics [74] deal mostly with the hardness of the task allocation problems and are often uninterested in distributed implementations. Moreover, these methods fall short in addressing agent heterogeneity or the mismatch between the number of agents and the number of tasks. Probabilistic one-sided matching problems [75, 76], consider preferences among the resources and match each player with at most one resource. The proposed algorithms are centralized and do not consider in general the specific values of rewards for each resource. Instead, the Cooperative Control literature does not consider the hardest of task allocation problems and aims to develop distributed algorithms under various degrees of problem knowledge.

Game-theoretic models have also been proposed to find solutions to task allocation problems. For sensor networks, each agent is equipped with an appropriate utility function [11, 12, 77] and the optimal task allocation is related to the Nash equilibrium of this game. Any Nash-seeking [21] algorithm returns a solution; but often, these algorithms require strong assumptions on the utility functions and their derivatives. In particular, earlier works of task allocation in Cooperative Control [11, 12] assume complete and perfect information on agents' utilities, while in practice only imperfect information about tasks and other agents' capabilities is available. Subsequently, this has been mostly addressed via consensus algorithms [24, 22], and gossip-based algorithms [25], where each agent applies these methods to estimate all other agents' strategies and compute the gradient of its own utility. We note that none of these works considers a scenario where the agent's utility itself changes due to external factors; thus, the available algorithms are not adaptive with respect to changing environments. Potential games can be used in this regard; however, they do not work when the reward parameters are unknown (i.e. they require again perfect information). This encompasses [77], which characterizes the transient behavior for set covering games, and [78], which studies a general potential game approach for task allocation. However, in the latter case, the agents are yet homogeneous and tasks have the same rewards for all agents; which facilitates the analysis and facilitates handling imperfect information.

4.3 Contributions

In this chapter, we consider a task-assignment problem where a number of agents is to be matched to an unrestricted set of tasks. In the considered formulation, the number of tasks per agent is not constrained, yet the optimal assignment problem remains combinatorial as the number of tasks is discrete. To deal with arbitrary heterogeneous agents, we derive a game-theoretic partition problem formulation that favors task distribution.

We then relax the game into a weight game, one per task. We obtain characterizations of the NE of each game, their relationship, and identify conditions under which the NE leads to an optimal solution of the original assignment problem. Leveraging the relaxed formulation, and under a full-information assumption, we derive a projected best-response dynamics that is shown to converge to the NE (and an optimal task allocation) in finite time. The algorithm and its analysis provide a stepping stone for another, d-PBRAG, which is distributed, does not require the perfect information about other agents' utilities or their individual strategies, and converges to the optimal task allocation, also in finite time, as rewards are revealed online.

4.4 Preliminaries

Here, we recall the notations and the well-known concepts from Chapter 2 that are used to solve the problem of interest.

Notations

The sets of real numbers, non-negative real numbers, and non-negative integers are denoted as $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, and $\mathbb{Z}_{>0}$, respectively. For a set $\mathcal{S}$, $|\mathcal{S}|$ denotes its cardinality, $2^{\mathcal{S}}$ represents the class of all its subsets, $\mathcal{S}^n$ denotes the n Cartesian product of $\mathcal{S}$ with itself, and $\mathcal{S}^{n \times m}$ collects all $n \times m$ matrices whose $(i, j)^{\text{th}}$ entry lies in $\mathcal{S}$. Given $\mathbf{M} \in \mathcal{S}^{n \times m}$, m_i^j is its $(i, j)^{\text{th}}$ entry, and $\mathbf{m}_i^{\top} \in \mathcal{S}^m$ (resp. $\mathbf{m}^j \in \mathcal{S}^n$) its i^{th} row (resp. its j^{th} column). For $x \in \mathbb{R}$, $[x]_0^1 := \max\{0, \min\{x, 1\}\}$. For a set $\mathcal{S}$, define $\max^{(2)} \mathcal{S} := \max\{s \in \mathcal{S} \mid s \neq \max \mathcal{S}\}$. For $\mathbf{x} \in \mathbb{R}^n$ and $\mathcal{S} \subseteq \mathbb{R}^n$, $d(\mathbf{x}, \mathcal{S}) := \inf_{\mathbf{y} \in \mathcal{S}} \|\mathbf{x} - \mathbf{y}\|_1$ is the distance of the vector from the set.

Game theory

A *strategic form game* [10] is a tuple $\mathcal{G} := \langle \mathcal{A}, \{\mathcal{S}_i\}_{i \in \mathcal{A}}, \{\psi_i\}_{i \in \mathcal{A}} \rangle$ consisting of the following components:

1. a set of *players* (or *agents*) $\mathcal{A}$;

2. a set of *strategies* $s_i \in \mathcal{S}_i$ available to each $i \in \mathcal{A}$;
3. a set of *utility functions* $\psi_i : \times_{i \in \mathcal{A}} \mathcal{S}_i \rightarrow \mathbb{R}$ over the strategy profiles of all the agents.

In what follows, s_{-i} denotes the strategy profile of all players other than $i \in \mathcal{A}$. Next, we formally state the definition of the NE of a strategic form game.

Definition 4.4.1 (Nash equilibrium). The strategy profile $(\hat{s}_i, \hat{s}_{-i})$ is a Nash equilibrium (NE) of $\mathcal{G}$ if and only if

$$\psi_i(\hat{s}_i, \hat{s}_{-i}) \geq \psi_i(s_i, \hat{s}_{-i}), \quad \forall s_i \in \mathcal{S}_i, \quad \forall i \in \mathcal{A}.$$

$\mathcal{NE}(\mathcal{G})$ denotes the set of all Nash equilibria of $\mathcal{G}$. •

Graph theory

A *directed graph* [8] $\mathcal{G} := (\mathcal{A}, \mathcal{E})$, is a tuple consisting of (a) a set of *nodes* (here agents $\mathcal{A}$); (b) a set of *arcs* $\mathcal{E} \subseteq \mathcal{A} \times \mathcal{A}$ between the nodes. The set $\mathcal{N}_i := \{j \in \mathcal{A} \mid (j, i) \in \mathcal{E}\}$ denotes the (in) *neighbors* of node $i \in \mathcal{A}$ and $\overline{\mathcal{N}}_i := \mathcal{N}_i \cup \{i\}$. A *path* is an ordered set of non-repeating nodes such that each tuple of adjacent nodes belongs to $\mathcal{E}$. The graph $\mathcal{G}$ is said to be strongly connected if there exists a path from every node to every other node. The *diameter* of the graph $\text{diam}(\mathcal{G})$ is the length of the largest possible path between any two nodes.

4.5 Problem Formulation

A group of agents $\mathcal{A} := \{1, \dots, n\}$ is to complete a set of tasks $\mathcal{Q} := \{1, \dots, m\}$, where $n \neq m$ possibly, in a distributed manner. For this purpose, each agent $i \in \mathcal{A}$ encodes via $\phi_i : \mathcal{Q} \rightarrow \mathbb{R}_{\geq 0}$ the importance of each task (the higher $\phi_i(q)$ the larger the agent's capability/fondness on $q \in \mathcal{Q}$) and $r_i : \mathcal{Q} \rightarrow \mathbb{R}_{\geq 0}$ the reward for completing each task. For the sake of brevity, define $f_i(q) := r_i(q)\phi_i(q) \geq 0, \forall q \in \mathcal{Q}, \forall i \in \mathcal{A}$. Note that any function

of the form $f_i(q) = g_i(r_i(q), \phi_i(q)) \geq 0$ can be used to define the effective reward that the agent receives. We choose this particular structure because it scales the reward that agent $i \in \mathcal{A}$ receives for task $q \in \mathcal{Q}$ by its capability (or fondness) $\phi_i(q)$ making it so that the effective reward $f_i(q)$ is *zero* if either $r_i(q)$ or $\phi_i(q)$ is *zero*. Further, the resulting cost function in (4.1a) can be interpreted as a discrete counterpart of the expected utility of coverage control problems [70, 79] and a type of clustering metric for heterogeneous agents. An optimal task assignment is the solution to

$$\max_{\mathcal{P}=(\mathcal{V}_1, \dots, \mathcal{V}_n) \subseteq \mathcal{Q}^n} J(\mathcal{P}) := \sum_{i \in \mathcal{A}} \sum_{q \in \mathcal{V}_i} f_i(q), \quad (4.1a)$$

$$\text{s.t.} \quad \bigcup_{i \in \mathcal{A}} \mathcal{V}_i = \mathcal{Q}; \quad \mathcal{V}_i \cap \mathcal{V}_j = \emptyset, \text{ if } i \neq j. \quad (4.1b)$$

Here, $\mathcal{V}_i \subseteq \mathcal{Q}$ is the set of tasks assigned to agent $i \in \mathcal{A}$ and $\mathcal{P} = (\mathcal{V}_1, \dots, \mathcal{V}_n)$ is the ordered collection of sets that defines a partition of $\mathcal{Q}$ (as in (4.1b)). For the sake of completeness, if $\mathcal{V}_i = \emptyset$ for some $i \in \mathcal{A}$, then we let $\sum_{q \in \mathcal{V}_i} f_i(q) = 0$.

The group of agents is to compute an optimal partition of the task set $\mathcal{Q}$ on their own. Naturally, each agent $i \in \mathcal{A}$ aims to get the tasks $q \in \mathcal{Q}$ for which $f_i(q)$ is the largest. This motivates the following definition.

Definition 4.5.1 (Task specific dominating agent). An agent $i \in \mathcal{A}$ is said to be a *dominating agent for task* $q \in \mathcal{Q}$ (or *i dominates q*), if $f_i(q) \geq f_j(q)$, $\forall j \in \mathcal{A}$. If a task $q \in \mathcal{Q}$ has exactly one dominating agent, we say that there exists a *unique dominating agent for task* q . •

The collection of possible strategies for each agent is a combinatorial class, which grows exponentially with m . To address this problem, we first assume that each $i \in \mathcal{A}$ measures the utility of a subset of tasks $\mathcal{V}_i \subseteq \mathcal{Q}$ via

$$H_i(\mathcal{V}_i, \mathcal{V}_{-i}) := \sum_{q \in \mathcal{V}_i} \left[f_i(q) - \max_{\{j \in \mathcal{A} \mid j \neq i, q \in \mathcal{V}_j\}} f_j(q) \right]. \quad (4.2)$$

Here, again, if $\mathcal{V}_i = \emptyset$ for some $i \in \mathcal{A}$, then we let $\sum_{q \in \mathcal{V}_i} [\cdot] = 0$ and $\max_{q \in \mathcal{V}_i} f_i(q) = 0$. This leads to the partition game

$$\mathcal{G}_P := \langle \mathcal{A}, \{2^{\mathcal{Q}}\}_{i \in \mathcal{A}}, \{H_i\}_{i \in \mathcal{A}} \rangle,$$

where the strategy of each agent is to choose a subset $\mathcal{V}_i$ of $\mathcal{Q}$ to maximize H_i . In this way, agent strategies are no-longer required to form a valid partition, but the utility in (4.2) penalizes each agent for taking tasks that others have chosen.

Second, we further relax this game by reducing the decision of each agent $i \in \mathcal{A}$ regarding task $q \in \mathcal{Q}$ to the computation of a weight $w_i^q \in [0, 1]$. Briefly, this defines $\mathbf{W} \in [0, 1]^{n \times m}$ as the matrix whose $(i, q)^{\text{th}}$ entry is w_i^q . Thus, $\mathbf{w}_i^\top \in [0, 1]^m$ (resp. $\mathbf{w}^q \in [0, 1]^n$) represents the weights that agent $i \in \mathcal{A}$ (resp. for task $q \in \mathcal{Q}$) gives to each task (resp. given by each agent). Agent $i \in \mathcal{A}$ is equipped with the utility function:

$$U_i(\mathbf{w}_i, \mathbf{w}_{-i}) = \sum_{q \in \mathcal{Q}} \left[f_i(q) w_i^q - \max_{j \in \mathcal{A} \setminus \{i\}} f_j(q) w_j^q w_i^q \right], \quad (4.3)$$

which collectively define the weight game

$$\mathcal{G}_W := \langle \mathcal{A}, \mathbf{W}, \{U_i\}_{i \in \mathcal{A}} \rangle,$$

In this way, a product of weights in the second part of the sum in (4.3) relaxes the check on overlapping task in (4.2). In this chapter, we ignore the trivial case where all agents get the same payoff for a task, as stated in the following.

Assumption 4.5.2 (Non-trivial task assignment). *Not all agents are dominating for each*

task $q \in \mathcal{Q}$. •

Note that Assumption 4.5.2 is equivalent to stating that the diversity of preferences and agents is such that there will be a dominating agent per task according to the designed game. This is because the best effective reward $f_i(q)$ depends on each agent's preferences, $\phi_i(q)$, on each task q . As preferences can differ, this changes the effective reward that the agent receives.

The above framework allows us to deal with a case where the $f_i(q)$ are unknown to the agent, but where these values can be learned progressively by an external mechanism until convergence. More precisely, we assume the following.

Assumption 4.5.3 (Converging reward sequence). *For each $i \in \mathcal{A}$, $q \in \mathcal{Q}$, there exists a sequence $\{z_i^q(t)\}_{t \in \mathbb{Z}_{>0}}$ such that $z_i^q(t) \rightarrow f_i(q)$ as $t \rightarrow \infty$.* •

Remark 4.5.4 (On the choice of utilities). We are interested in solving the optimization problem (4.1) in a distributed manner while the $f_i(q)$'s are unknown. We do this by designing a multi-agent system game; i.e., by equipping agents with a suitable utility that is both easy to compute and results into useful properties. In this case, the NE of the game are related to the optimizers of the task allocation problem, which allows us to reduce the original combinatorial problem into an NE-seeking problem. •

In what follows, we first study the games when the reward parameters are known. Then we adapt the results for the case when only a converging reward sequence is available.

Now we formally state the goals of this work.

Problem 4.5.5. Given the aforementioned setup and the non-trivial task assignment assumption, find

1. a relationship between the NE of $\mathcal{G}_P$ and $\mathcal{G}_W$,
2. a relationship between the NE and optimal partitions according to (4.1),

3. a distributed algorithm that converges to the NE of the limiting weight game $\mathcal{G}_W$ under the converging reward sequence assumption. •

4.6 On Nash Equilibria and Optimal Partitions

We start by addressing the first two problems above. Thus, we first characterize the NE of the partition game $\mathcal{G}_P$.

Lemma 4.6.1 (Nash equilibria of $\mathcal{G}_P$). *The strategy $(\widehat{\mathcal{V}}_i, \widehat{\mathcal{V}}_{-i}) \in \mathcal{NE}(\mathcal{G}_P)$ if and only if:*

1. *for each $q \in \mathcal{Q}$, $\exists i \in \mathcal{A}$ dominating for q and $q \in \widehat{\mathcal{V}}_i$;*
2. *if j is not a dominating agent for task q , then $q \notin \widehat{\mathcal{V}}_j$.*

Proof. First, we show the necessity of Properties 1 and 2. Suppose $(\widehat{\mathcal{V}}_i, \widehat{\mathcal{V}}_{-i}) \in \mathcal{NE}(\mathcal{G}_P)$. We prove Property 1 by contradiction. assum $\exists q \in \mathcal{Q}$ such that $\forall i \in \mathcal{A}$ dominating for task q , $q \notin \widehat{\mathcal{V}}_i$. Pick such an agent i and take $\mathcal{V}_i = \widehat{\mathcal{V}}_i \cup \{q\}$. Then,

$$H_i(\mathcal{V}_i, \widehat{\mathcal{V}}_{-i}) - H_i(\widehat{\mathcal{V}}_i, \widehat{\mathcal{V}}_{-i}) = f_i(q) - \max_{k \neq i, q \in \widehat{\mathcal{V}}_k} f_k(q) > 0.$$

The inequality is strict since i is dominating and the max is over all agents that are not dominating for q (by assumption). This is a contradiction with $(\widehat{\mathcal{V}}_i, \widehat{\mathcal{V}}_{-i}) \in \mathcal{NE}(\mathcal{G}_P)$.

The necessity of Property 2 also follows from contradiction. Suppose $\exists q \in \mathcal{Q}$ and a $j \in \mathcal{A}$ not dominating for q with $q \in \widehat{\mathcal{V}}_j$. From Property 1, there is an $i \in \mathcal{A}$ dominating for q with $q \in \widehat{\mathcal{V}}_i$. Thus, with strategy $(\widehat{\mathcal{V}}_j, \widehat{\mathcal{V}}_{-j})$,

$$f_j(q) - \max_{k \neq j, q \in \widehat{\mathcal{V}}_k} f_k(q) = f_j(q) - \max_{k \in \mathcal{A}} f_k(q) < 0.$$

Now, consider $\mathcal{V}_j = \widehat{\mathcal{V}}_j \setminus \{q\}$. It follows that $H_j(\mathcal{V}_j, \widehat{\mathcal{V}}_{-j}) - H_j(\widehat{\mathcal{V}}_j, \widehat{\mathcal{V}}_{-j}) > 0$, contradicting $(\widehat{\mathcal{V}}_j, \widehat{\mathcal{V}}_{-j}) \in \mathcal{NE}(\mathcal{G}_P)$.

Now, we show sufficiency. Let $(\widehat{\mathcal{V}}_i, \widehat{\mathcal{V}}_{-i})$ satisfy Properties 1 and 2 and let $i \in \mathcal{A}$ be an arbitrary but fixed agent. Suppose that $\mathcal{V}_i \neq \widehat{\mathcal{V}}_i$ is any other strategy. Then, the proof follows from three cases:

Case (i): $\exists q \in \mathcal{V}_i$ such that $q \notin \widehat{\mathcal{V}}_i$ and i is dominating for q . Then, since $\exists j \in \mathcal{A}$ dominating for q with $q \in \widehat{\mathcal{V}}_j$,

$$f_i(q) - \max_{k \neq i, q \in \widehat{\mathcal{V}}_k} f_k(q) = f_i(q) - f_j(q) = 0.$$

Case (ii): $\exists q \in \mathcal{V}_i$ such that $q \notin \widehat{\mathcal{V}}_i$ and i does not dominate q . Then, as $\exists j \in \mathcal{A}$ dominating for q and $q \in \widehat{\mathcal{V}}_j$, we have

$$f_i(q) - \max_{k \neq i, q \in \widehat{\mathcal{V}}_k} f_k(q) = f_i(q) - f_j(q) < 0.$$

Case (iii): $\exists q \in \widehat{\mathcal{V}}_i$ such that $q \notin \mathcal{V}_i$. This can only happen if i dominates q (else, by Property 2, $q \notin \widehat{\mathcal{V}}_j$). Then,

$$f_i(q) - \max_{k \neq i, q \in \widehat{\mathcal{V}}_k} f_k(q) \geq 0.$$

From the above, it is easy to see that any deviation from $(\widehat{\mathcal{V}}_i, \widehat{\mathcal{V}}_{-i})$ will not result in an increase in utility for i since $H_i(\mathcal{V}_i, \widehat{\mathcal{V}}_{-i}) - H_i(\widehat{\mathcal{V}}_i, \widehat{\mathcal{V}}_{-i}) = f_i(q) - \max_{k \neq i, q \in \widehat{\mathcal{V}}_k} f_k(q)$. $\blacksquare$

From the previous result, at least one of the dominating agents will be assigned to a task by means of a NE strategy of $\mathcal{G}_{\mathbf{P}}$. However, this does not preclude that two dominating agents are assigned the same task. Next, we show that the NE of the relaxed game $\mathcal{G}_{\mathbf{W}}$ are equivalent to the NE of $\mathcal{G}_{\mathbf{P}}$.

Lemma 4.6.2 (Nash equilibria of $\mathcal{G}_{\mathbf{W}}$). *The strategy $(\widehat{\mathbf{w}}_i, \widehat{\mathbf{w}}_{-i}) \in \mathcal{NE}(\mathcal{G}_{\mathbf{W}})$ if and only if:*

1. *for each $q \in \mathcal{Q}$, $\exists i \in \mathcal{A}$ dominating for q and $\widehat{w}_i^q = 1$;*
2. *if j is not a dominating agent for task q , then $\widehat{w}_j^q = 0$.*

Proof. First, we show the necessity of all properties. Suppose $(\widehat{\mathbf{w}}_i, \widehat{\mathbf{w}}_{-i}) \in \mathcal{NE}(\mathcal{G}_W)$. We show Property 1 is necessary by contradiction. Consider an arbitrary $q \in \mathcal{Q}$ and suppose that for all dominating agents $i_q^* \in \mathcal{A}$ for task q , it holds that $\widehat{w}_{i_q^*}^q < 1$. In particular, for any such i_q^* , we have $\max_{j \neq \{i_q^*\}} f_j(q) \widehat{w}_j^q < f_{i_q^*}(q)$. Now consider the strategy $\mathbf{w}_{i_q^*}$, where $w_{i_q^*}^q = 1$ and $w_p^q = \widehat{w}_p^q, \forall p \neq q \in \mathcal{Q}$. Then,

$$U_{i_q^*}(\mathbf{w}_{i_q^*}, \widehat{\mathbf{w}}_{-i_q^*}) - U_{i_q^*}(\widehat{\mathbf{w}}_{i_q^*}, \widehat{\mathbf{w}}_{-i_q^*}) = \left[f_{i_q^*}(q) - \max_{j \neq i_q^*} f_j(q) \widehat{w}_j^q \right] [1 - \widehat{w}_{i_q^*}^q] > 0.$$

This leads to a contradiction with $(\widehat{\mathbf{w}}_i, \widehat{\mathbf{w}}_{-i}) \in \mathcal{NE}(\mathcal{G}_W)$.

We similarly show Property 2 is necessary by contradiction. Let $q \in \mathcal{Q}$ be an arbitrary task, and suppose that $\exists j \in \mathcal{A}$ which is not dominating for q but for which $\widehat{w}_j^q > 0$. Due to Property 1, let i_q^* be the dominating agent for q such that $\widehat{w}_{i_q^*}^q = 1$. Now define $\mathbf{w}_j$, with $w_j^q = 0$ and $w_p^q = \widehat{w}_p^q, \forall p \neq q \in \mathcal{Q}$. Then,

$$U_j(\mathbf{w}_j, \widehat{\mathbf{w}}_{-j}) - U_j(\widehat{\mathbf{w}}_j, \widehat{\mathbf{w}}_{-j}) = [f_j(q) - f_{i_q^*}(q)] [-\widehat{w}_j^q] > 0,$$

where the inequality is because both terms are negative. This contradicts $(\widehat{\mathbf{w}}_i, \widehat{\mathbf{w}}_{-i}) \in \mathcal{NE}(\mathcal{G}_W)$.

Next, we show sufficiency. Let $(\widehat{\mathbf{w}}_i, \widehat{\mathbf{w}}_{-i}) \in [0, 1]^{n \times m}$ be a candidate strategy satisfying Properties 1- 2 and let $i \in \mathcal{A}$. Take any other $\mathbf{w}_i \neq \widehat{\mathbf{w}}_i$ and a task $q \in \mathcal{Q}$. The proof follows from the following cases.

Case (i): i dominates q and $\widehat{w}_i^q = 1$. Then, by Definition 4.5.1,

$$\left[f_i(q) - \max_{j \neq i} f_j(q) \widehat{w}_j^q \right] w_i^q \leq \left[f_i(q) - \max_{j \neq i} f_j(q) \widehat{w}_j^q \right] \widehat{w}_i^q.$$

Case (ii): i is a dominating agent for task q and $\widehat{w}_i^q < 1$. Then, since $\exists j \in \mathcal{A}$

dominating for q and $\widehat{w}_j^q = 1$,

$$\begin{aligned} \left[f_i(q) - \max_{k \neq i} f_k(q) \widehat{w}_k^q \right] w_i^q &= [f_i(q) - f_j(q)] w_i^q \\ &= [f_i(q) - f_j(q)] \widehat{w}_i^q = \left[f_i(q) - \max_{k \neq i} f_k(q) \widehat{w}_k^q \right] \widehat{w}_i^q = 0, \end{aligned}$$

since $f_i(q) - f_j(q) = 0$.

Case (iii): i is not a dominating agent for task q (and hence $\widehat{w}_i^q = 0$). Again, $\exists j \in \mathcal{A}$ dominating for q and $\widehat{w}_j^q = 1$. Then,

$$\begin{aligned} \left[f_i(q) - \max_{k \neq i} f_k(q) \widehat{w}_k^q \right] w_i^q &= [f_i(q) - f_j(q)] w_i^q \\ &< [f_i(q) - f_j(q)] \widehat{w}_i^q = \left[f_i(q) - \max_{k \neq i} f_k(q) \widehat{w}_k^q \right] \widehat{w}_i^q, \end{aligned}$$

since $f_i(q) < f_j(q)$. Now, using these three cases, it is easy to see that any deviation from $(\widehat{\mathbf{w}}_i, \widehat{\mathbf{w}}_{-i})$ will not result in an increase the utility of i . ■

As a direct implication of Lemma 4.6.2, for any $\mathbf{W} \in [0, 1]^{n \times m}$ we can define $C : [0, 1]^{n \times m} \rightarrow (2^{\mathcal{Q}})^n$ as

$$C(\mathbf{W}) := (\text{tsupp}(\mathbf{w}_1), \dots, \text{tsupp}(\mathbf{w}_n)), \quad (4.4)$$

where $\text{tsupp}(\mathbf{w}_i) := \{q \in \mathcal{Q} \mid w_i^q = 1\}$, $\forall i \in \mathcal{A}$. Then,

$$\mathcal{NE}(\mathcal{G}_{\mathbf{P}}) = C(\mathcal{NE}(\mathcal{G}_{\mathbf{W}})). \quad (4.5)$$

Next, we relate the optimal partition and the NE of the two games through the following theorem.

Theorem 4.6.3 (Optimal partitions and NE). *Given the problem in (4.1), $\mathcal{O} \subseteq$*

$C(\mathcal{NE}(\mathcal{G}_W))$, where

$$\mathcal{O} := \{\mathcal{P}^* \mid \mathcal{P}^* \text{ is a solution to (4.1)}\}. \quad (4.6)$$

Proof. By (4.5), we can equivalently show that $\mathcal{O} \subseteq \mathcal{NE}(\mathcal{G}_P)$. Let $\mathcal{P}^* \in \mathcal{O}$. It is easy to see that $q \in \mathcal{V}_i^*$ only if $i \in \mathcal{A}$ is a dominating agent for task q . Moreover, if $j \in \mathcal{A}$ is not dominating for q , then $q \notin \mathcal{V}_j^*$. Then from Lemma 4.6.1, $\mathcal{P}^* \in \mathcal{NE}(\mathcal{G}_P)$. The rest follows from (4.5). $\blacksquare$

The above result states that if an agent $i \in \mathcal{A}$ is assigned tasks using the translated support of the NE of $\mathcal{G}_W$, this set is a superset of the optimizers of (4.1). The extra solutions arise when there are non-unique dominating agents for a task. When there are unique dominating agents, the next result shows there is a unique NE for $\mathcal{G}_W$. This follows from Lemma 4.6.2 immediately, so we skip a formal proof.

Corollary 4.6.4 (Uniqueness of NE). *Suppose that for each $q \in \mathcal{Q}$, i_q^* is the unique dominating agent for task q . Then $\mathcal{NE}(\mathcal{G}_W) = \{\widehat{\mathbf{W}}\}$ where, for each $q \in \mathcal{Q}$, $\widehat{\mathbf{W}}$ satisfies 1) $\widehat{w}_{i_q^*}^q = 1$, and 2) $\widehat{w}_j^q = 0$, $\forall j \neq i_q^*$. Further, $\mathcal{P}^* = C(\widehat{\mathbf{W}})$ is the unique solution to (4.1). $\blacksquare$*

In general, $\mathcal{NE}(\mathcal{G}_W)$ is a superset of the set of optimal partitions. The next example makes this clear.

Example 4.6.5 (Optimal partitions and NE). Let $\mathcal{A} = \{1, 2\}$ and $\mathcal{Q} = \{\mathbf{a}, \mathbf{b}\}$. Assume the values $f_1(\mathbf{a}) = f_2(\mathbf{a}) = 0.5$, $f_1(\mathbf{b}) = 0.7$ and $f_2(\mathbf{b}) = 0.3$. Then, $\mathcal{O} = \{(\{\mathbf{a}, \mathbf{b}\}, \emptyset), (\{\mathbf{b}\}, \{\mathbf{a}\})\}$; $\mathcal{NE}(\mathcal{G}_P) = \{(\{\mathbf{a}, \mathbf{b}\}, \emptyset), (\{\mathbf{a}, \mathbf{b}\}, \{\mathbf{a}\}), (\{\mathbf{b}\}, \{\mathbf{a}\})\}$; and

$$\mathcal{NE}(\mathcal{G}_W) = \left\{ \begin{bmatrix} 1 & 1 \\ \lambda & 0 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} \mu & 1 \\ 1 & 0 \end{bmatrix} \right\},$$

where λ, μ can independently take any value in $[0, 1]$. Thus, in this case $\mathcal{O} \subsetneq \mathcal{NE}(\mathcal{G}_P) = C(\mathcal{NE}(\mathcal{G}_W))$. Interestingly, note that there is an optimal partition in which agent 2 gets

no task. •

We conclude this section by including remarks on how our framework compares with standard potential games and congestion games. This interesting discussion is a result of the review process of the original paper and was not included due to space limitations of the journal.

Remark 4.6.6. (*Comparison with potential games*). We would also like to point out the following regarding the existence of a potential function. We are ultimately interested in solving the optimization problem in (4.1). Thus, given the game structure (and since the NE of a potential game are the maximizers of the potential function), the ideal candidate for a potential function should be,

$$H^1(\mathbf{w}) = \sum_{i \in \mathcal{A}} \sum_{q \in \mathcal{V}_i} f_i(q) w_i^q,$$

since this is the relaxed cost function related to the optimization problem in (4.1). This is ideal in the sense that we would like a NE of our game to correspond to a local maxima of this function. Now consider a situation where $\mathcal{A} = \{1, 2\}$, $\mathcal{Q} = \{\mathbf{a}\}$ and $f_1(\mathbf{a}) > f_2(\mathbf{a})$. Moreover, let $w_1^{\mathbf{a}} = w_2^{\mathbf{a}} = 1$. Then, for agent 2, any other strategy $\bar{w}_2^{\mathbf{a}} < 1$ results in $U_2(w_1^{\mathbf{a}}, \bar{w}_2^{\mathbf{a}}) > U_2(w_1^{\mathbf{a}}, w_2^{\mathbf{a}})$ while $H^1(w_1^{\mathbf{a}}, \bar{w}_2^{\mathbf{a}}) < H^1(w_1^{\mathbf{a}}, w_2^{\mathbf{a}})$. Thus, H^1 cannot be a generalized potential function let alone an exact potential function for this game.

Another natural choice for a potential function is the sum of the agents' utilities (which corresponds to the utilitarian social utility in game theory):

$$H^2(\mathbf{w}) = \sum_{i \in \mathcal{A}} U_i(\mathbf{w}).$$

We can use similar arguments as before to show that this cannot be a potential function either. Again, consider a situation where $\mathcal{A} = \{1, 2\}$, $\mathcal{Q} = \{\mathbf{a}\}$ and $f_1(\mathbf{a}) > f_2(\mathbf{a})$. Further, let $w_2^{\mathbf{a}} = 1$. Then, $U_1(w_1^{\mathbf{a}}, 1) = [f_1(\mathbf{a}) - f_2(\mathbf{a})]w_1^{\mathbf{a}}$ and $H^2(w_1^{\mathbf{a}}, 1) = f_2(\mathbf{a})[1 - w_1^{\mathbf{a}}]$. Then,

with $\bar{w}_1^a > w_1^a$, $U_1(\bar{w}_1^a, 1) > U_1(w_1^a, 1)$, but $H^2(\bar{w}_1^a, 1) < H^2(w_1^a, 1)$. Thus, H^2 cannot be a generalized potential function. Admittedly, there could be some other potential function ($\bar{H} \neq H^1, H^2$) for this game. But then, the Nash equilibria would correspond to the local maximizers of $\bar{H}$, which may not necessarily be the solution to (4.1). •

Remark 4.6.7. (*Comparison with congestion games*). We recall that potential functions are closely related to congestion games, where a group of agents competes with each other to allocate resources among themselves. The agents get a reward based on the resource they are allocated to and are also penalized based on the number of agents that are allocated to that same resource. The type of utilities U_i for our game can be given a similar interpretation. In [80], the author showed that every congestion game is an exact potential game and later in [81] the authors showed that any game that has an exact potential function is equivalent to some congestion game. In congestion games, each agent's utility generally depends on the resource allocated to them, and it decreases with the number of other agents that make use of the same resource. Thus, as the number of agents allocated to a given resource increases, the payoff of the assigned agents decreases. The latter is consistent with our setting; however, in congestion games different agents allocated to the same resource get the same reward, which is not consistent with our heterogeneity assumption. Moreover, the potential function for congestion games is generally given as a sum of agents' utilities, which, as we have argued above, cannot be a potential function for a heterogeneous game like ours. Hence, the framework of congestion games is related but we believe does not fully capture our scenario completely. •

This is a main reason why we have individually documented the relations between the optimizers of the task allocation problem, the Nash equilibria of the weight game. Next, we design a dynamical system using which the agents can figure out the optimal partition on their own.

4.7 Best Response Projected Gradient Ascent

From the previous section, we know that if the agents play the weight game $\mathcal{G}_W$, then the NE form a superset of the optimal task partition (with slight abuse of notation). Thus, here we let the agents update their weights (from any initial feasible weight) using the gradient of their utility while assuming the others do not change their weights. For such a dynamical system, we aim to relate its equilibria to the NE of $\mathcal{G}_W$ and hence also relate it to the set $\mathcal{O}$ of optimal solutions to (4.1). Now, from (4.3),

$$\frac{\partial}{\partial w_i^q} U_i = f_i(q) - \max_{j \in \mathcal{A} \setminus \{i\}} f_j(q) w_j^q =: u_i^q(\mathbf{w}^q). \quad (4.7)$$

Thus, the weights are updated using the following dynamics:

$$w_i^q(t+1) = [w_i^q(t) + \gamma_i^q u_i^q(\mathbf{w}^q(t))]_0^1, \quad (4.8)$$

with $\gamma_i^q \in \mathbb{R}_{>0}$, $\forall i \in \mathcal{A}, \forall q \in \mathcal{Q}$. We call this the projected best response ascending gradient dynamics (PBRAG). From (4.7) and (4.8), note that in order to compute the weight updates, each agent $i \in \mathcal{A}$ needs to know $f_j(q) w_j^q$, for all $j \neq i$. This requires that each agent must talk to every other agent to compute its own gradient. The equilibrium points of this dynamics is

$$\mathcal{S}_{\text{eq}} := \left\{ \mathbf{W} \in [0, 1]^{n \times m} \mid \begin{aligned} & \left[w_i^q + \gamma_i^q u_i^q(\mathbf{w}^q) \right]_0^1 = w_i^q, \\ & \forall i \in \mathcal{A}, \forall q \in \mathcal{Q} \end{aligned} \right\}. \quad (4.9)$$

For this, the following result can be stated immediately.

Lemma 4.7.1. $\overline{\mathbf{W}} \in \mathcal{S}_{\text{eq}}$ if and only if $\overline{\mathbf{W}} \in \mathcal{NE}(\mathcal{G}_W)$.

Proof. We prove this by showing that $\overline{\mathbf{W}} \in \mathcal{S}_{\text{eq}}$ if and only if $\overline{\mathbf{W}}$ follows Properties 1

and 2 of Lemma 4.6.2. Suppose that $\overline{\mathbf{W}} \in \mathcal{S}_{\text{eq}}$ and consider an arbitrary but fixed $q \in \mathcal{Q}$. We prove Property 1 by contradiction and assume that $\forall i \in \mathcal{A}$ dominating for q , $\overline{w}_i^q < 1$. Now, for any dominating agent $i_q^* \in \mathcal{A}$, $\max_{j \in \mathcal{A} \setminus \{i_q^*\}} f_j(q) \overline{w}_j^q < f_{i_q^*}(q)$. Thus, $u_{i_q^*}^q(\overline{\mathbf{w}}^q) > 0$. Since $\overline{w}_{i_q^*}^q < 1$ and $\gamma_{i_q^*}^q > 0$, this contradicts $\overline{\mathbf{W}} \in \mathcal{S}_{\text{eq}}$.

Next we prove Property 2 also by contradiction. Suppose that $\exists j \in \mathcal{A}$ not dominating for task q but $\overline{w}_j^q > 0$. Due to Property 1, let $i_q^* \in \mathcal{A}$ be the dominating agent for task q such that $\overline{w}_{i_q^*}^q = 1$. Then,

$$u_j^q(\overline{\mathbf{w}}^q) = f_j(q) - \max_{k \in \mathcal{A} \setminus \{j\}} f_k(q) \overline{w}_k^q = f_j(q) - f_{i_q^*}(q) < 0.$$

Again, as $\overline{w}_j^q > 0$ and $\gamma_j^q > 0$, this contradicts $\overline{\mathbf{W}} \in \mathcal{S}_{\text{eq}}$.

To show sufficiency, let $\overline{\mathbf{W}}$ satisfy Properties 1 and 2. Then it is easy to see that for each task $q \in \mathcal{Q}$, $u_i^q(\overline{\mathbf{w}}^q) \geq 0$ if $i \in \mathcal{A}$ dominates q with $\overline{w}_i^q = 1$, $u_i^q(\overline{\mathbf{w}}^q) = 0$ if $i \in \mathcal{A}$ dominates q with $\overline{w}_i^q < 1$, and $u_j^q(\overline{\mathbf{w}}^q) < 0$ if $j \in \mathcal{A}$ is not dominating for q (hence $\overline{w}_j^q = 0$). Then, $\overline{\mathbf{W}} \in \mathcal{S}_{\text{eq}}$ follows since $\gamma_j^q > 0$. $\blacksquare$

From Lemma 4.7.1, we can also infer that if there is a unique dominating agent, then the equilibrium set becomes a singleton and follows the same structure as in Corollary 4.6.4.

In what follows, we show that starting from any initial weights, the dynamics (4.8) converges to an equilibrium.

Theorem 4.7.2 (PBRAG converges to an equilibrium weight). *Suppose Assumption 4.5.2 holds. Consider the dynamics (4.8) with an initial condition $\mathbf{W}(0) \in [0, 1]^{n \times m}$ and let $\mathbf{W}(t)$ be the solution trajectory. Then $\lim_{t \rightarrow \infty} \mathbf{W}(t) = \overline{\mathbf{W}} \in \mathcal{S}_{\text{eq}}$.*

Proof. Notice that for the dynamics (4.8), the weight associated with each task evolves independently from the weights associated with other tasks. Thus, consider an arbitrary but fixed $q \in \mathcal{Q}$. Next, consider any $i \in \mathcal{A}$ that dominates q . From (4.7), $u_i^q(\mathbf{w}^q) \geq 0$, $\forall \mathbf{w}^q \in [0, 1]^n$. Thus, since $\gamma_i^q > 0$, $\forall i \in \mathcal{A}$, $w_i^q(t)$ is non-decreasing. Hence,

$w_i^q(t) \rightarrow \widehat{w}_i^q \in [0, 1]$ as $t \rightarrow \infty$, since $[0, 1]$ is compact. Now consider $\mathbf{I}^q := \{j \in \mathcal{A} \mid j \text{ dominates } q\}$, the set $\mathbf{X} := \{\mathbf{v} \in [0, 1]^{|\mathbf{I}^q|} \mid v_j = 1 \text{ for some } j \in \mathbf{I}^q\}$ and define the continuous function $V(\mathbf{w}^q) := d(\{w_i^q\}_{i \in \mathbf{I}^q}, \mathbf{X})$. It is clear that $V(\mathbf{w}^q(t+1)) \leq V(\mathbf{w}^q(t)) \forall t \in \mathbb{Z}_{>0}$. Applying the LaSalle invariance principle, there is convergence to the largest invariant set in $V(\mathbf{w}(t)) = V(\mathbf{w}(t+1))$ for all t . We argue this set is necessarily $\mathbf{X}$. Otherwise, invariance implies that $u_i^q(\mathbf{w}^q) = 0$ for any dominating agent $i \in \mathcal{A}$. However, this occurs if and only if $\exists i' \in \mathcal{A}, i \neq i'$, another dominating agent for task q such that $w_{i'}^q = 1$; otherwise, $u_i^q(\mathbf{w}^q) > 0$. Thus, $\{w_i^q(t)\}_{i \in \mathbf{I}^q} \rightarrow \mathbf{X}$ as $t \rightarrow \infty$. This along with previous discussion proves that $\widehat{w}_i^q$ follows Property 1 of Lemma 4.6.2.

Next consider any $j \in \mathcal{A}$ that is not dominating for q . From the previous part of the proof, we know that there is a $i \in \mathcal{A}$ dominating for q for which $w_i^q(t) \rightarrow 1$ and thus $f_i(q)w_i^q(t) \rightarrow f_i(q)$ as $t \rightarrow \infty$. This implies that $\exists \tau \in \mathbb{Z}_{>0}$ such that $\max_{k \in \mathcal{A} \setminus \{j\}} f_k(q)w_k^q(t) \geq f_j(q) + \nu$, for some $\nu > 0$ and $\forall t \geq \tau$. Then, as $u_j^q(\mathbf{w}^q(t)) \leq -\nu < 0$, $w_j^q(t)$ is a strictly decreasing sequence (after τ time steps). Thus, from the dynamics in (4.8), $w_j^q(t) \rightarrow \widehat{w}_j^q = 0$ as $t \rightarrow \infty$. Hence, $\widehat{w}_j^q$ follows Property 2 of Lemma 4.6.2. ■

When there is a unique dominating agent for each task, we can guarantee finite-time convergence to an optimal partition.

Theorem 4.7.3 (PBRAG converges in finite time). *Suppose Assumption 4.5.2 holds and suppose that for each $q \in \mathcal{Q}$, there exists a unique dominating agent, $i_q^* \in \mathcal{A}$. Define $\gamma := \min_{i \in \mathcal{A}, q \in \mathcal{Q}} \gamma_i^q > 0$, and let $\delta := \min_{q \in \mathcal{Q}} [f_{i_q^*}(q) - \max_{j \neq i_q^*} f_j(q)] > 0$. Consider the dynamics (4.8) starting from $\mathbf{W}(0) \in [0, 1]^{n \times m}$, with the solution trajectory $\mathbf{W}(t) \rightarrow \overline{\mathbf{W}} \in \mathcal{S}_{\text{eq}}$, as $t \rightarrow \infty$. Then $w_i^q(t) = \overline{w}_i^q, \forall i \in \mathcal{A}, \forall q \in \mathcal{Q}, \forall t \geq 2 \lceil (\gamma\delta)^{-1} \rceil$.*

Proof. From Lemma 4.7.1 and Corollary 4.6.4, it is clear that $\mathcal{S}_{\text{eq}}$ is a singleton set. Let $\overline{\mathbf{W}} \in \mathcal{S}_{\text{eq}}$ be the unique equilibrium point. From Theorem 4.7.2, we know that $\mathbf{W}(t) \rightarrow \overline{\mathbf{W}}$ as $t \rightarrow \infty$. From (4.7), $u_{i_q^*}^q(\mathbf{w}^q(t)) \geq \delta > 0, \forall t \in \mathbb{Z}_{>0}$ and hence from (4.8), $\forall t \in \mathbb{Z}_{>0}, w_{i_q^*}^q(t+1) \geq [w_{i_q^*}^q(t) + \gamma_{i_q^*}^q \delta]_0^1 \geq [w_{i_q^*}^q(0) + (t+1) \gamma_{i_q^*}^q \delta]_0^1$. The inequality holds

since $[\cdot]_0^1$ is a nondecreasing function. Thus, $w_{i_q^*}^q(t) = 1$, for all $t \geq \lceil (\gamma\delta)^{-1} \rceil \geq (\gamma\delta)^{-1} \geq [1 - w_{i_q^*}^q(0)][\gamma_{i_q^*}^q\delta]^{-1}$. Now define $\tau := \lceil (\gamma\delta)^{-1} \rceil$ and consider any $j \neq i_q^*$ and notice from (4.7) that $u_j^q(\mathbf{w}^q(t)) \leq -\delta < 0, \forall t \geq \tau$. Thus, $w_j^q(t) \leq [w_j^q(\tau) - t\gamma_j^q\delta]_0^1, \forall t \geq \tau$. The inequality again holds since $[\cdot]_0^1$ is non decreasing. So, $w_j^q(t) = 0$, for all $t \geq \tau + \lceil (\gamma\delta)^{-1} \rceil \geq \tau + [\gamma_j^q\delta]^{-1} \geq \tau + w_j^q(\tau)[\gamma_j^q\delta]^{-1}$. ■

Remark 4.7.4 (On the effect of step-size on convergence). By Theorem 4.7.2, (4.8) converges to an equilibrium weight when $\gamma_i^q > 0, \forall i \in \mathcal{A}, \forall q \in \mathcal{Q}$. Thus agents can choose any constant positive step size and guarantee convergence to a NE of the weight game. Further inspection of Theorem 4.7.3 leads to this interesting observation. Since $\delta^{-1} > 0$, the individual γ_i^q 's can be chosen in such a way that $0 < (\gamma\delta)^{-1} < 1$. Then $2\lceil (\gamma\delta)^{-1} \rceil = 2$. That is, by choosing a sufficiently large step size and communicating with every other agent, the agents can reach the NE in at most *two* time steps. •

In order to avoid all-to-all communication, it is possible to adapt (4.8) introducing a consensus subroutine. In the next section, we utilize this idea to handle decentralization together with unknown rewards.

4.8 Distributed Task Allocation

Finally, we provide a solution to Problem 4.5.5 (3). Recall that in Section 4.7, each agent $i \in \mathcal{A}$ computes $\max_{j \neq i} f_j(q)w_j^q$ using information from all other agents. Here, we introduce a communication graph $\mathcal{G} := (\mathcal{A}, \mathcal{E})$ with vertex set $\mathcal{A}$. The arc set $\mathcal{E}$ defines the connections between agents, with $(i, j) \in \mathcal{E}$ if and only if $i \in \mathcal{A}$ can send information to $j \in \mathcal{A}$. For the sake of brevity, let $d := \text{diam}(\mathcal{G})$.

Note that (from the proof of Theorem 4.7.2), if each agent $i \in \mathcal{A}$ uses a convex combination of the max and second unique max (*i.e.* $\lambda \max_{j \in \mathcal{A}} f_j(q) + (1 - \lambda) \max_{j \in \mathcal{A}}^{(2)} f_j(q)$, for some $\lambda \in (0, 1)$) instead of $\max_{j \neq i} f_j(q)w_j^q$, the outcome of the dynamics (4.8) remains similar. This is because the aforementioned convex combination penalizes any

non-dominating agent to reduce the weight to *zero* and encourages a dominating agent to increase its weight to *one* in a similar fashion to the utility in (4.3). Moreover, this convex combination is the same quantity for every agent and does not depend on the individual agent as the penalizing term in the utility in (4.3) does. This, in turn, is useful in providing a distributed PBRAG (d-PBRAG). Using a communication graph, the following result gives a distributed way to find the max and second unique max values.

Lemma 4.8.1 (Agreement on the two largest variables in a network). *Let $\mathcal{G}$ be a strongly connected graph and consider*

$$M_i^q(t+1) = \max_{j \in \mathcal{N}_i} M_j^q(t), \quad (4.10a)$$

$$S_i^q(t+1) = \max^{(2)} \left\{ \{S_j^q(t)\}_{j \in \mathcal{N}_i}, M_i^q(t), v_i^q \right\}, \quad (4.10b)$$

with initial condition $M_i^q(0) = S_i^q(0) = v_i^q \in \mathbb{R}_{\geq 0}$, $\forall i \in \mathcal{A}$, $\forall q \in \mathcal{Q}$. Then $\forall q \in \mathcal{Q}$ and $\forall i \in \mathcal{A}$;

1. $M_i^q(t) = \max\{v_j^q\}_{j \in \mathcal{A}}$, $\forall t \geq d$,
2. $S_i^q(t) = \max^{(2)}\{v_j^q\}_{j \in \mathcal{A}}$, $\forall t \geq 2d$.

Proof. We show this for an arbitrary but fixed $q \in \mathcal{Q}$. Let $i_q^* \in \operatorname{argmax}\{v_j^q\}_{j \in \mathcal{A}}$. Then, from (4.10a), $M_{i_q^*}^q(t) = v_{i_q^*}^q$, $\forall t \in \mathbb{Z}_{>0}$. Thus, $\forall i_q^* \in \operatorname{argmax}\{v_j^q\}_{j \in \mathcal{A}}$, $M_j^q(t) = v_{i_q^*}^q$, $\forall j \in \mathcal{N}_{i_q^*}$, $\forall t \geq 1$. Continuing this argument inductively proves Property 1 since $\mathcal{G}$ is strongly connected.

To show Property 2, we use Property 1. Now let $i_q^* \in \operatorname{argmax}^{(2)}\{v_j^q\}_{j \in \mathcal{A}}$. Then, from (4.10b), $S_{i_q^*}^q(t) = v_{i_q^*}^q$, $\forall t \geq d$. Thus, similarly, $\forall i_q^* \in \operatorname{argmax}^{(2)}\{v_j^q\}_{j \in \mathcal{A}}$, $S_j^q(t) = v_{i_q^*}^q$, $\forall j \in \mathcal{N}_{i_q^*}$, $\forall t \geq d+1$. Again, continuing this argument proves Property 2 since $\mathcal{G}$ is strongly connected. ■

To compute the gradient and update the weights w_i^q simultaneously, we propose

the following dynamics and discuss an intuition behind it in the remark that follows.

$$w_i^q(t+1) = \left[w_i^q(t) + \gamma_i^q(t) \left(z_i^q(t) - \frac{1}{2} (M_i^q(t) + S_i^q(t)) \right) \right]_0^1, \quad (4.11a)$$

$$M_i^q(t+1) = \sigma_{\text{sw}} \left(\max_{j \in \mathcal{N}_i} M_j^q(t), e_i^q(t+1), t+1, T \right), \quad (4.11b)$$

$$S_i^q(t+1) = \sigma_{\text{sw}} \left(\max^{(2)} \left\{ \{S_j^q(t)\}_{j \in \mathcal{N}_i}, M_i^q(t), e_i^q(t) \right\}, e_i^q(t+1), t+1, T \right), \quad (4.11c)$$

$$e_i^q(t+1) = \sigma_{\text{sw}} \left(e_i^q(t), z_i^q(t+1), t+1, T \right), \quad (4.11d)$$

for some $T \in \mathbb{R}_{\geq 0}$ and where σ_{sw} is the switching function

$$\sigma_{\text{sw}}(m, z, t, T) := \begin{cases} z, & \text{if } t \bmod T = 0, \\ m, & \text{otherwise.} \end{cases} \quad (4.12)$$

Remark 4.8.2 (d-PBRAG with agreement and periodic input injection). Note that $\forall i \in \mathcal{A}, \forall q \in \mathcal{Q}$, the weight update in (4.11a) uses the sequence $\{z_i^q(t)\}_{t \in \mathbb{Z}_{>0}}$ and a time-varying step-size $\gamma_i^q(t)$ instead of $f_i(q)$ and a constant step-size γ_i^q ; respectively, as in (4.8). The periodic switching function σ_{sw} ensures that $e_i^q(t)$ holds the value $z_i^q(t)$ for every T time-steps. This in turn allows (4.11b) and (4.11c) to run an agreement subroutine as (4.10) every T time-steps with $v_i^q = z_i^q(kT)$, for $k \in \mathbb{Z}_{>0}$. Thus, at every time-step which is a multiple of T , each agent believes that its own value is the maximum and corrects this belief over the next $T - 1$ time-steps. Then, as per the discussion preceding Lemma 4.8.1, each agent $i \in \mathcal{A}$ uses the convex combination $0.5(M_i^q(t) + S_i^q(t))$ in (4.11a) of its estimated max and second unique max value in lieu of $\max_{j \neq i} f_j(q)w_j^q$. This gives us a distributed way of assigning the task to the correct agent while reducing the information being shared. •

Theorem 4.8.3 (Asymptotic behavior of d-PBRAG). *Suppose Assumptions 4.5.2 and 4.5.3*

hold. Define

$$\Delta^q := \left(\max_{i \in \mathcal{A}} f_i(q) - \min_{i \in \mathcal{A}} f_i(q) \right) > 0, \forall q \in \mathcal{Q}. \quad (4.13)$$

Consider any $\varepsilon \in (0, 1)$. Suppose $\forall t \in \mathbb{Z}_{>0}$, $\gamma_i^q(t) = \alpha_i^q$, with $0 < \alpha_i^q \leq \varepsilon (2d \Delta^q)^{-1}$, $\forall i \in \mathcal{A}$, $\forall q \in \mathcal{Q}$. Next, define $\alpha := \min_{i \in \mathcal{A}, q \in \mathcal{Q}} \alpha_i^q$, $\mu^q := 0.5 (\max\{f_i(q)\}_{i \in \mathcal{A}} - \max^{(2)}\{f_i(q)\}_{i \in \mathcal{A}})$, and

$$\mu := (1 - \nu) \min_{q \in \mathcal{Q}} \mu^q > 0, \quad (4.14)$$

with $\nu \in (0, 1)$. Further, suppose $T > 2d + (\alpha \mu)^{-1} + 1$. Let $\mathbf{W}(t)$ be the solution trajectory to (4.11) starting from $\mathbf{W}(0) \in [0, 1]^{n \times m}$. Then $\exists \tau(\mathbf{W}(0)) \in \mathbb{Z}_{>0}$ such that $\forall t \geq \tau$,

1. $w_i^q(t) = 1$ if $i \in \mathcal{A}$ dominates $q \in \mathcal{Q}$;
2. $w_j^q(t) \leq \varepsilon$ if $j \in \mathcal{A}$ is not dominating for $q \in \mathcal{Q}$.

Thus $C(\mathbf{W}(t))$ converges in finite number of time steps.

Proof. First note that the bounds on α_i^q 's and T are valid because of Assumption 4.5.2. Then, we show the claims for an arbitrary but fixed $q \in \mathcal{Q}$.

Recall that because of Assumption 4.5.3, $\exists \tau_0 \in \mathbb{Z}_{>0}$ such that $\forall t, t' \geq \tau_0$, $z_i^q(t) - 0.5(z_i^q(t) + z_j^q(t')) < \Delta^q$, $\forall i, j \in \mathcal{A}$. Moreover τ_0 can be chosen such that $z_{i_q^*}^q(t) - 0.5(z_{i_q^*}^q(t) + z_j^q(t')) \geq (1 - \nu) \mu^q > 0$, for any $\nu \in (0, 1)$, if $i_q^* \in \operatorname{argmax}_{i \in \mathcal{A}} f_i(q)$ and $\forall j \in \mathcal{A}$ such that $j \notin \operatorname{argmax}_{i \in \mathcal{A}} f_i(q)$.

Now consider any $i_q^* \in \operatorname{argmax}_{i \in \mathcal{A}} f_i(q)$ and any $\nu \in (0, 1)$. Then from Remark 4.8.2, and the previous discussion, $\forall t \geq \tau_0$,

$$\alpha_{i_q^*}^q (z_{i_q^*}^q(t) - 0.5(M_{i_q^*}^q(t) + S_{i_q^*}^q(t))) \geq \alpha_{i_q^*}^q (1 - \nu) \mu^q \geq \alpha_{i_q^*}^q \mu.$$

This proves Property 1 of this theorem as $\alpha_{i_q^*}^q \mu > 0$.

Next consider any $j \notin \operatorname{argmax}_{i \in \mathcal{A}} f_i(q)$. Note from Remark 4.8.2 that $\forall t \geq \tau_0$ such that $t \in \{kT + 2d, \dots, 2kT - 1\}$ for some $k \in \mathbb{Z}_{>0}$, $w_j^q(t)$ strictly decreases, since,

$$z_j^q(t) - 0.5(M_j^q(t) + S_j^q(t)) \leq -(1 - \nu)\mu^q < 0.$$

Consider any $t \geq \tau_0$ such that $t \in \{kT + 2d, \dots, 2kT - 1\}$ with $k \in \mathbb{Z}_{>0}$. Then, since $w_j^q(kT + 2d - 1) \leq 1$,

$$w_j^q(t) \leq 1 - ((t \bmod T) - 2d)\alpha\mu,$$

and hence because of the bound on T , $w_j^q(2kT - 1) = 0$. Finally, consider any $t \geq \tau_0$ such that $t \in \{kT, \dots, kT + 2d - 1\}$ with $k \in \mathbb{Z}_{>0}$. Then, since $w_j^q(kT - 1) = 0$ (from previous arguments), we have $w_j^q(t) \leq (t \bmod T)\alpha_i^q \Delta^q$. Thus combining all these arguments proves Property 2.

The final claim follows from the previous ones and (4.4). ■

Note that the previous result does not guarantee that the weights converge. This stems from the fact that at periodic times, each agent believes that it gets the maximum reward for each task. Moreover, the previous result needs information about the limits of the converging sequences to provide bounds for the step sizes and the period of input injection. This can be avoided by allowing time-varying step sizes as stated next.

Theorem 4.8.4 (d-PBRAG converges to NE). *Suppose Assumptions 4.5.2 and 4.5.3 hold. Let $\mathbf{W}(t)$ be the solution trajectory of (4.11) from $\mathbf{W}(0) \in [0, 1]^{n \times m}$, with $T > 2d + 1$ and,*

$$\gamma_i^q(t) = \begin{cases} \alpha_i^q(k) > 0, & \text{if } t \in \{kT, \dots, kT + 2d - 1\}, \\ \beta_i^q(k) > 0, & \text{if } t \in \{kT + 2d, \dots, 2kT - 1\}, \end{cases}$$

$\forall i \in \mathcal{A}, \forall q \in \mathcal{Q}$, with $k \in \mathbb{Z}_{>0}$. Further, $\forall i \in \mathcal{A}, \forall q \in \mathcal{Q}$; take sequences $\alpha_i^q(k) \rightarrow 0$ as

Table 4.1. Approximate $f_i(q)$ values for Simulations

$i \in \mathcal{A} \backslash q \in \mathcal{Q}$	1	2	3	4
1	0.4536	0.4407	0.2881	0.0055
2	0.7504	0.2228	0.0411	0.2801
3	0.7656	0.0987	0.1381	0.2491
4	0.3023	0.2211	0.3334	0.2462

$i \in \mathcal{A} \backslash q \in \mathcal{Q}$	5	6	7	8
1	0.0049	0.2394	0.3152	0.2217
2	0.2374	0.0768	0.0852	0.1760
3	0.2969	0.1003	0.1471	0.6902
4	0.3033	0.4991	0.1231	0.5931

$k \rightarrow \infty$ and $\beta_i^q(k) \rightarrow \infty$ as $k \rightarrow \infty$. Then $\mathbf{W}(t) \rightarrow \overline{\mathbf{W}} \in \mathcal{NE}(\mathcal{G}_W)$ as $t \rightarrow \infty$.

Proof. From hypothesis, $\forall \varepsilon > 0, \exists K \in \mathbb{Z}_{>0}$, such that $\forall t \in \{kT, \dots, kT + 2d - 1\}$, with $k \geq K$, $\alpha_i^q(t) \leq \varepsilon (2d \Delta^q)^{-1}$, with Δ^q as in (4.13). Moreover, K can be chosen such that $\forall t \in \{kT + 2d, \dots, 2kT - 1\}$, with $k \geq K$, $T > 2d + ((1 - \nu)\mu \min_{i \in \mathcal{A}, q \in \mathcal{Q}} \alpha_i^q \beta_i^q(t))^{-1} + 1$ for any $\nu \in (0, 1)$ and with μ as in (4.14). Then this result is a consequence of applying similar arguments as in the proof of Theorem 4.8.3 and using (4.5). ■

We conclude this section by discussing some interesting observations about the parameters in (4.11) and some advantages.

Remark 4.8.5 (On the implementation of d-PBRAG). Note that even though $\text{diam}(\mathcal{G})$ is an internal property of the communication graph $\mathcal{G}$ and requires some structural knowledge of the same, the claims in Theorems 4.8.3 and 4.8.4 remain true if d is replaced with n . This is because $\text{diam}(\mathcal{G}) \leq n$. Moreover, these results can be extended to time-varying communication graphs with periodic connectivity because the agreement subroutine still works. Further, note that the conditions in Theorem 4.8.4 are only sufficient for convergence. In fact, $\beta_i^q(k)$ need not grow unbounded, but then knowledge of converging reward values are required for proper functioning of the algorithm. For example, if μ is large, small

values of $\beta_i^q(k)$ are sufficient to guarantee convergence; but if μ is small then $\beta_i^q(k)$ values have to be sufficiently large in order to guarantee that non-dominating agents are not assigned the task. Finally, notice that in order for the algorithm to work, each agent $i \in \mathcal{A}$ has to pass two values $(M_i^q(t), S_i^q(t))$ for each task $q \in \mathcal{Q}$ to its neighbors at each time step. This makes the local communication cost of this algorithm of the order of $O(m)$ per iteration time. •

Remark 4.8.6 (Advantages of d-PBRAG). Compared with Nash-seeking algorithms, where agents keep an estimate of all other players' strategies per task, d-PBRAG only requires agents to exchange two quantities with neighbors per iteration and task. This is advantageous from (i) a privacy-preservation, (ii) communication-efficiency, and (iii) convergence-rates perspectives (as it is well known, consensus algorithms rates' depend on the eigenvalues of the graph Laplacian, while max-consensus iterations are bounded by the diameter of the graph). •

4.9 Simulations

In this section, we verify our major claims and illustrate some interesting features of our algorithms.

4.9.1 Fast convergence of PBRAG with large step-size

Here, we simulate $n = 4$ agents to optimally allocate $m = 8$ tasks with $r_i(q), \phi_i(q) \sim \text{Unif}[0, 1], \forall i \in \mathcal{A}, q \in \mathcal{Q}$. In particular, Table 4.1 gives the approximate values of $f_i(q)$, $\forall i \in \mathcal{A}, q \in \mathcal{Q}$. For each $q \in \mathcal{Q}$, the teal cell represents $\max_{i \in \mathcal{A}} f_i(q)$.

We first verify the claim in Remark 4.7.4. Figure 4.1 shows the solution evolution using (4.8) from an initial $\mathbf{W}(0) = \mathbf{0}$. The optimal partition as in Figure 4.1 is given as $\mathcal{V}_1 = \{2, 7\}$, $\mathcal{V}_2 = \{4\}$, $\mathcal{V}_3 = \{1, 8\}$, $\mathcal{V}_4 = \{3, 5, 6\}$. Here, since the values of $f_i \in [0, 1]$,

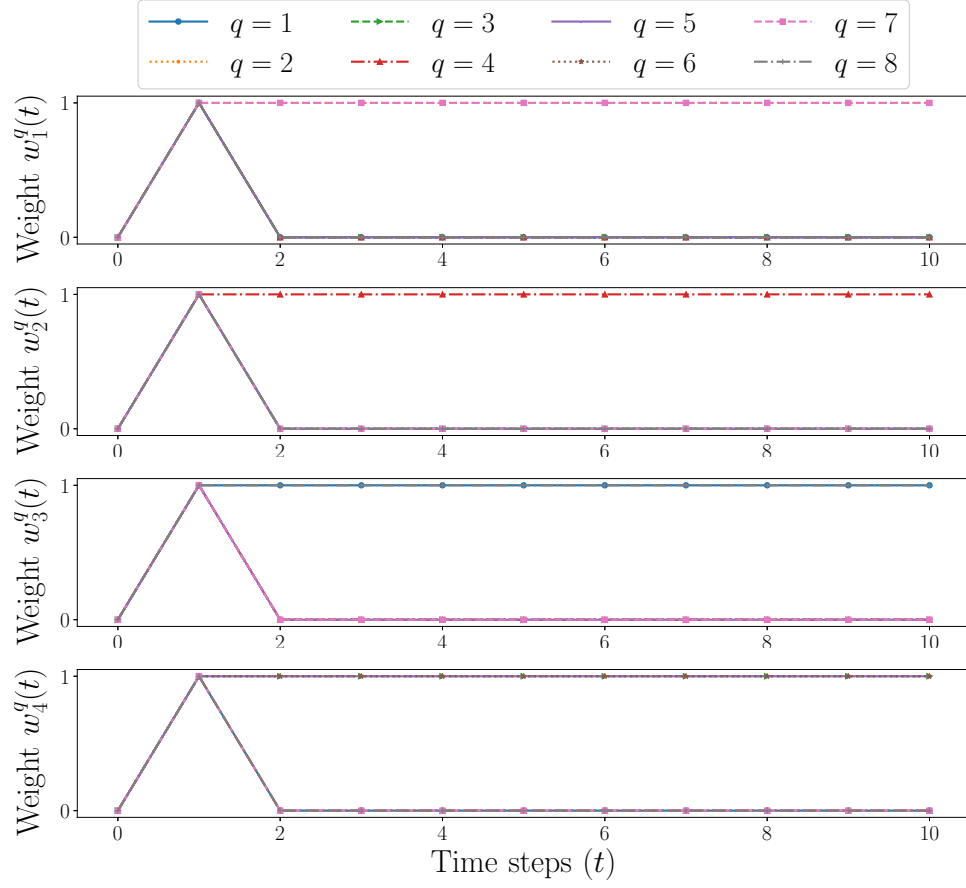


Figure 4.1. PBRAG using (4.8) and large step size γ . Plots share a common legend and time axis.

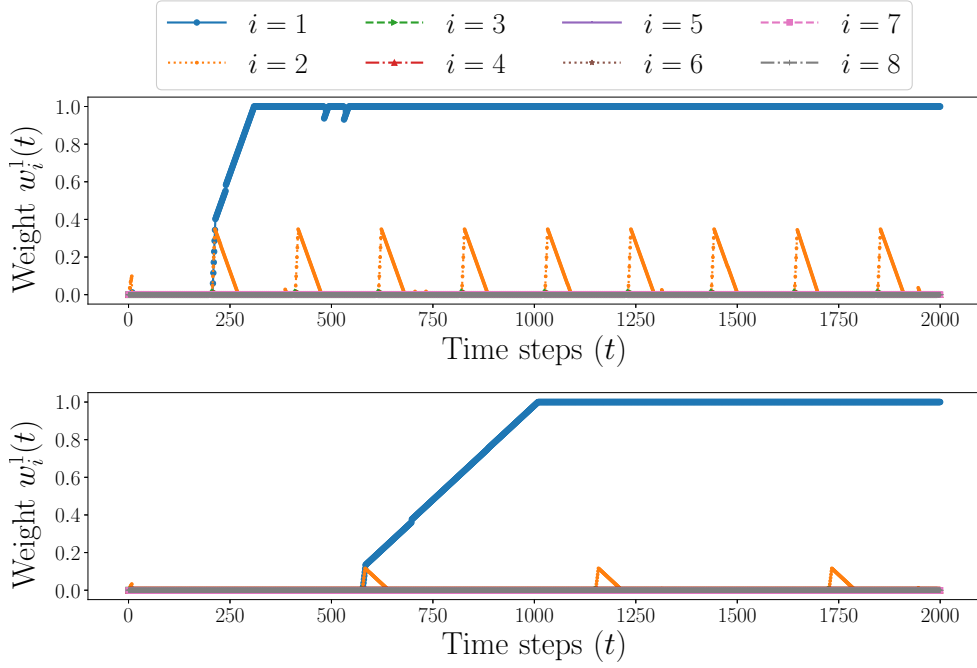


Figure 4.2. d-PBRAG for unknown rewards with constant step-size using (4.11) on cyclic communication graph. Step-size $\gamma_i^q(t)$ and time-period T were chosen as in Theorem 4.8.3 with different ε and $\nu = 0.1$. The plots share a common legend. (Top) $\varepsilon = 0.9$. (Bottom) $\varepsilon = 0.3$.

$\gamma_i^q \in O(10^6)$ was required to make the solutions converge in *two* time steps. For larger deviations in the values of f_i , much smaller values of γ_i^q 's can achieve similar effects.

4.9.2 Effect of constant step-size on d-PBRAG

Here, we deal with the claims in Theorem 4.8.3 for $n = 8$ agents optimally allocating $m = 1$ task. We take $f_1(1) = B$, $f_2(1) = 0.9B$, and $f_i(1) = 0.3B/i$, $\forall i \in \{3, \dots, 8\}$, with $B = 1000$. Thus agent 1 is the dominating agent. Further, we consider an unknown reward structure with $z_i^q(t) = f_i(q) + a_i^q \cos(b_i^q t) \exp(-c_i^q t)$, $\forall i \in \mathcal{A}$, $\forall q \in \mathcal{Q}$, where $a_i^q \sim \text{Unif}[0, f_i(q)]$, $b_i^q \sim \text{Unif}[0, 10]$, and $c_i^q \sim \text{Unif}[0, 1]$. We set the communication graph $\mathcal{G} = (\mathcal{A}, \mathcal{E})$ with $\mathcal{E} = \{(1, 2), (2, 3), (3, 4), (4, 1)\}$.

Figure 4.2 shows the solution evolution using (4.11) with constant step-size from an initial $\mathbf{W}(0) = \mathbf{0}$. It is interesting to note from Figure 4.2 that if ε is large, then $w_1^1(t)$ reaches 1 faster, but the weights of the non-dominating agents rise higher. On the other hand if ε is small, then the rise in the weights of the non-dominating agents is less but $w_1^1(t)$ reaches 1 slower. This is because ε affects the choice of T as well.

4.9.3 d-PBRAG with time-varying step-sizes

Here, we simulate $n = 4$ agents optimally allocating $m = 4$ tasks. We take the unknown reward structure as in Section 4.9.2 with $f_i(q)$ as in Table 4.1 (we only consider the tasks for $q \in \{1, 2, 3, 4\}$). Further, we use the distributed approach using (4.11) with time-varying step-sizes as described in Theorem 4.8.4. We also set $\mathcal{G}$ as in Section 4.9.2.

Figure 4.3 shows the solution evolution using (4.11) from an initial $\mathbf{W}(0) = \mathbf{0}$. The optimal partition as in Figure 4.3 is given as $\mathcal{V}_1 = \{2\}$, $\mathcal{V}_2 = \{4\}$, $\mathcal{V}_3 = \{1\}$, $\mathcal{V}_4 = \{3\}$. This is similar to the observation in Section 4.9.1 (restricted to $q \in \{1, 2, 3, 4\}$). Further, notice from Figure 4.3 that the weights of agents 3 and 4 take longer time to settle than agents 1 and 2. In general, the convergence rate of the algorithm depends on difficult-to-characterize properties of the unknown reward sequences.

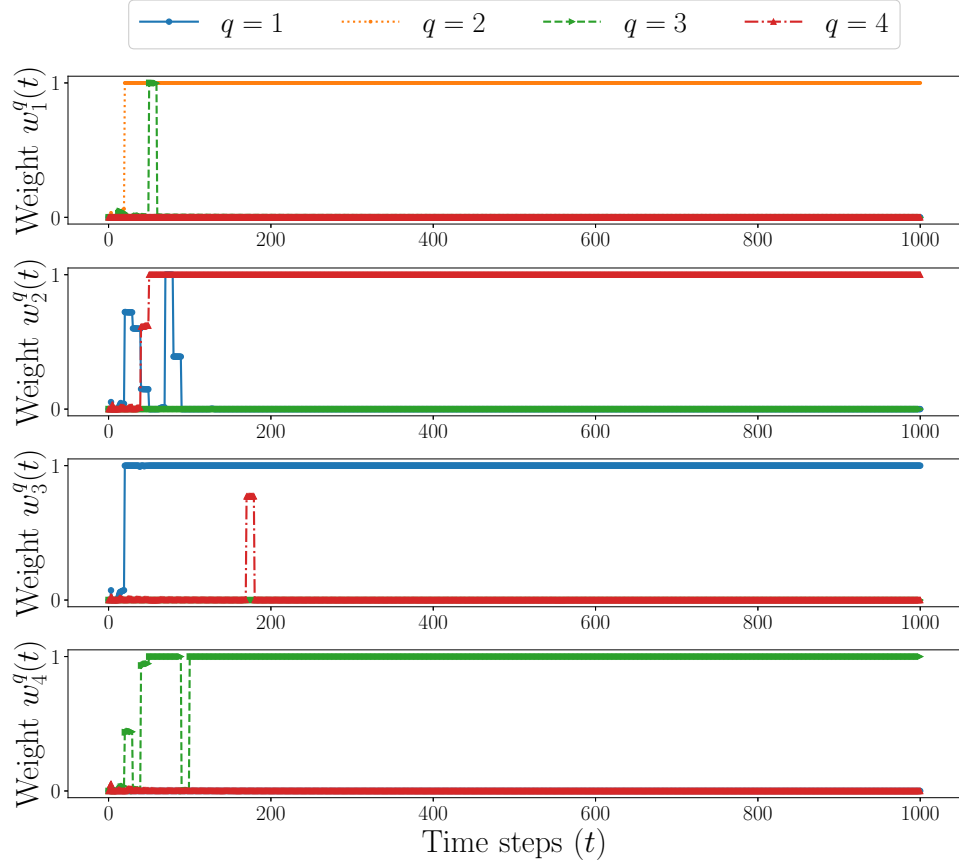


Figure 4.3. d-PBRAG for unknown rewards with time-varying step-size using (4.11) on cyclic communication graph. Plots share a common legend and time axis.

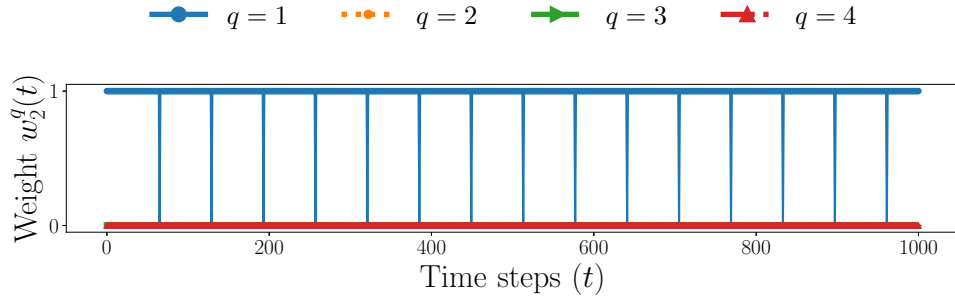


Figure 4.4. Distributed Hungarian implementation for the same scenario as in Figure 4.3. Plot is shown for one agent only.

We compare our algorithm with the distributed Hungarian algorithm in [61]. In order to incorporate the converging reward sequence, we restart the algorithm every n^3 time step (since it was shown in [61] that the algorithm converges in $O(n^3)$ time steps). In Figure 4.4 we show the evolution of agent 2 only (for the sake of space). As can be seen, distributed Hungarian keeps oscillating while d-PBRAG converges. The oscillations could be an artifact of restarting the algorithm; but, to the best of our knowledge, that is a reasonable approach to incorporate new information regarding the converging reward sequence.

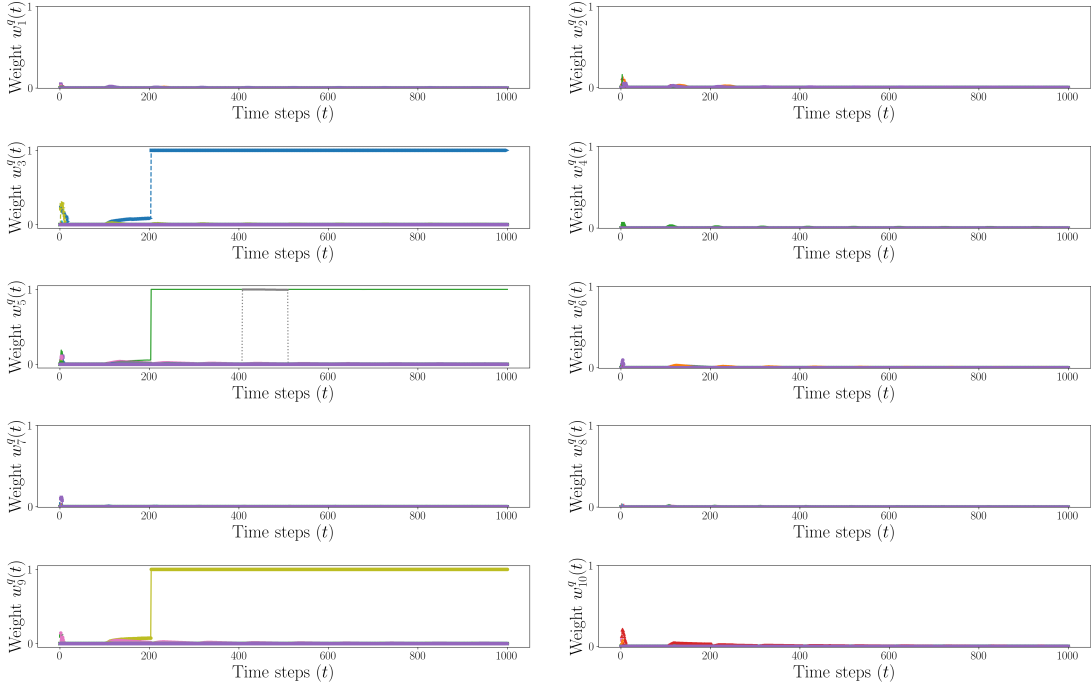


Figure 4.5. d-PBRAG for a large number of agents on a cyclic communication graph. Weight evolution of agents 1-10.

4.9.4 Comparison with Hungarian algorithm with large number of agents

This comparison was a result of the review process of the original paper that we omitted due to space limitations. The simulation is done for 50 agents and 50 tasks where

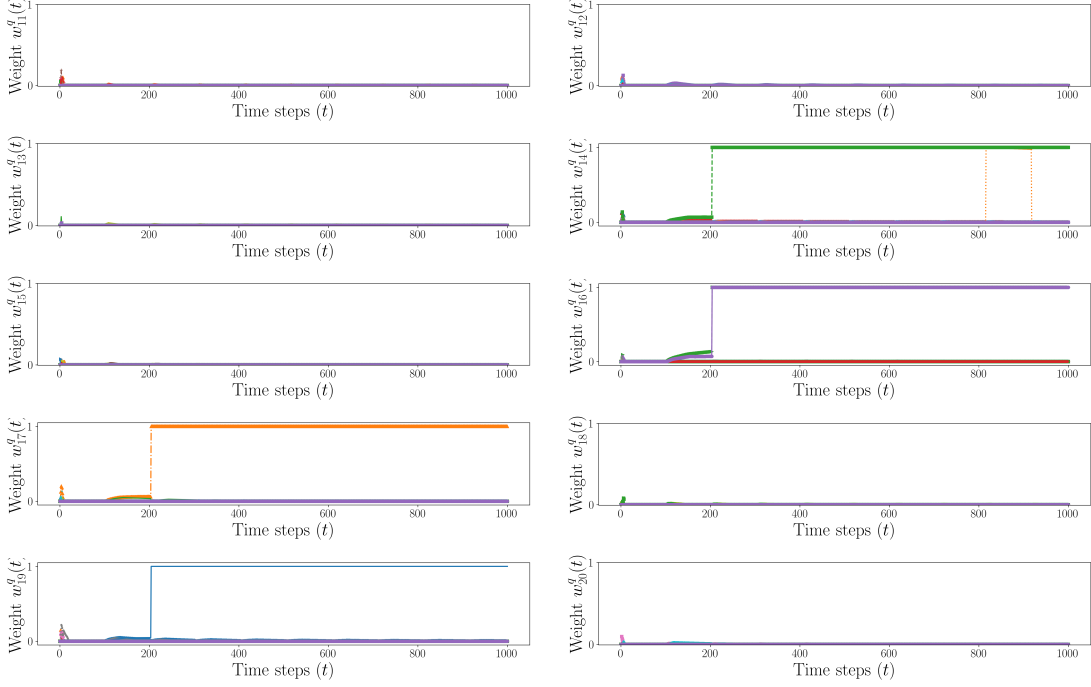


Figure 4.6. d-PBRAG for a large number of agents on a cyclic communication graph. Weight evolution of agents 11-20.

the actual reward values are unknown. Each agent $i \in \mathcal{A}$ gets an update regarding its reward for task $q \in \mathcal{Q}$ from the sequence $z_i^q(t) = f_i(q) + a_i^q \cos(b_i^q t) \exp(-c_i^q t)$, where $a_i^q \sim \text{Unif}[0, f_i(q)]$, $b_i^q \sim \text{Unif}[0, 10]$, and $c_i^q \sim \text{Unif}[0, 1]$. Here, $f_i(q) \in [0, \infty)$. The communication graph is chosen as a cycle graph, *i.e.* $\mathcal{G} = (\mathcal{A}, \mathcal{E})$ with $\mathcal{A} = \{1, 2, \dots, 50\}$, $\mathcal{E} = \{(1, 2), (2, 3), (3, 4), \dots, (49, 50), (50, 1)\}$ (so the diameter of the graph is 50).

The weight evolution of the agents using our algorithm are reported in Figures 4.6-4.9. Please note that we have broken up the plots into 5 different figures for the ease of presentation. We again observe that the simulations behave similarly to the ones for smaller networks. The main notable difference here is that, since the diameter of the communication graph is larger, the algorithm takes longer to converge. The algorithm practically reaches the optimal assignment at ≈ 600 time steps. Fluctuations observed after this point are due to changes in the estimates of the unknown rewards. In other words, for a fixed reward case, the algorithm would converge by 600 steps.

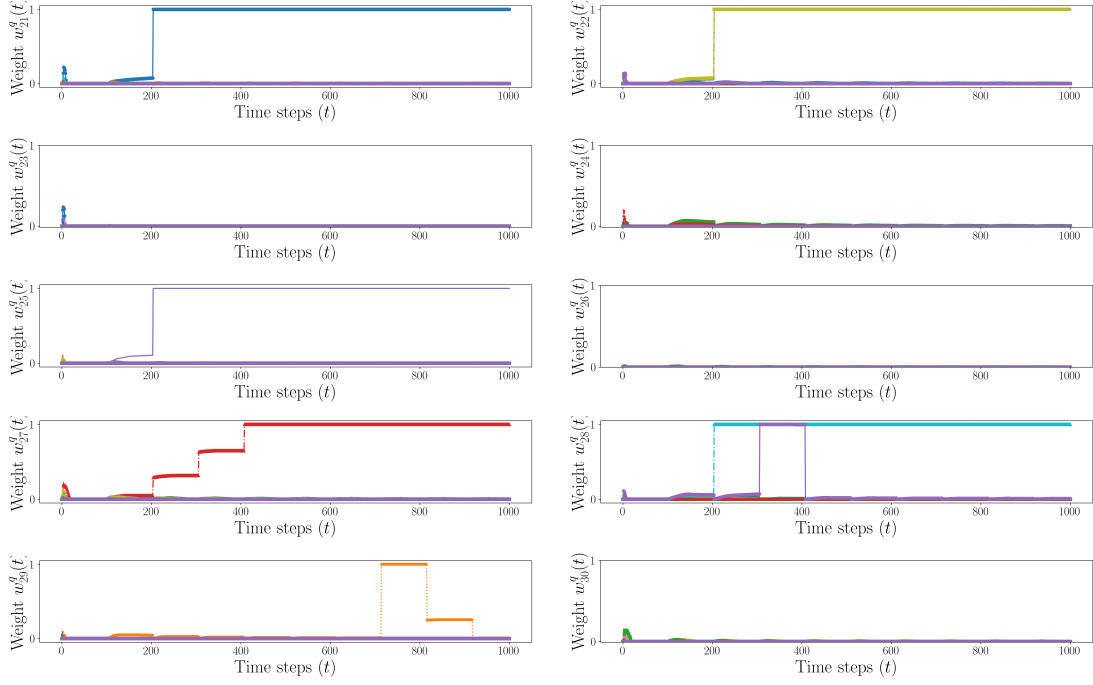


Figure 4.7. d-PBRAG for a large number of agents on a cyclic communication graph. Weight evolution of agents 21-30.

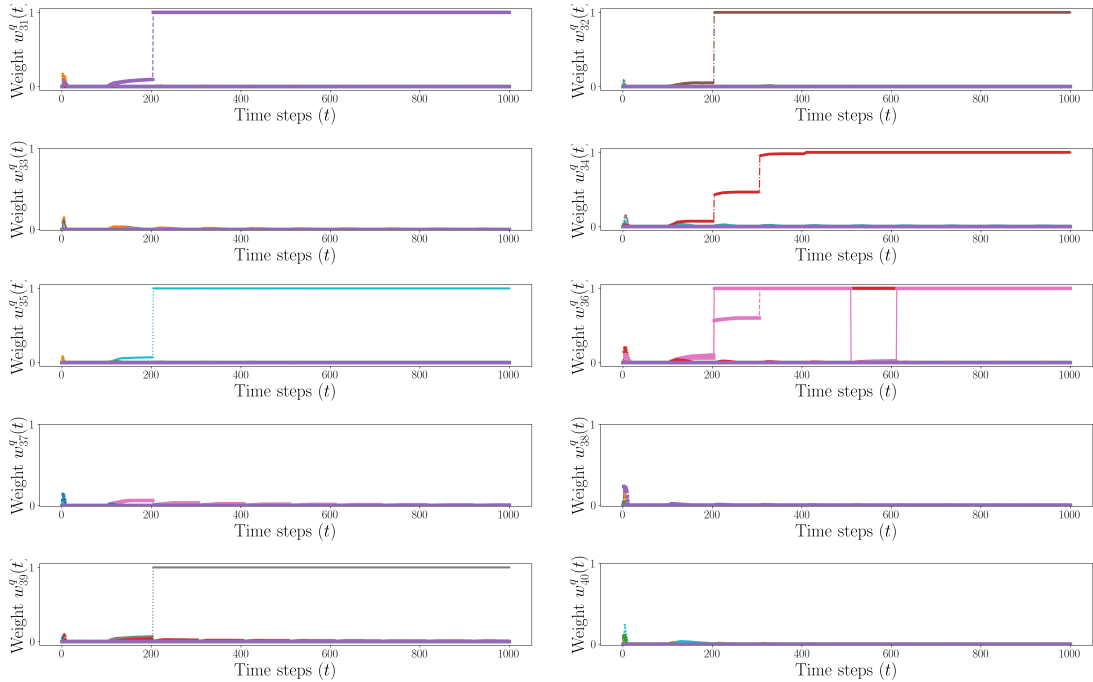


Figure 4.8. d-PBRAG for a large number of agents on a cyclic communication graph. Weight evolution of agents 31-40.

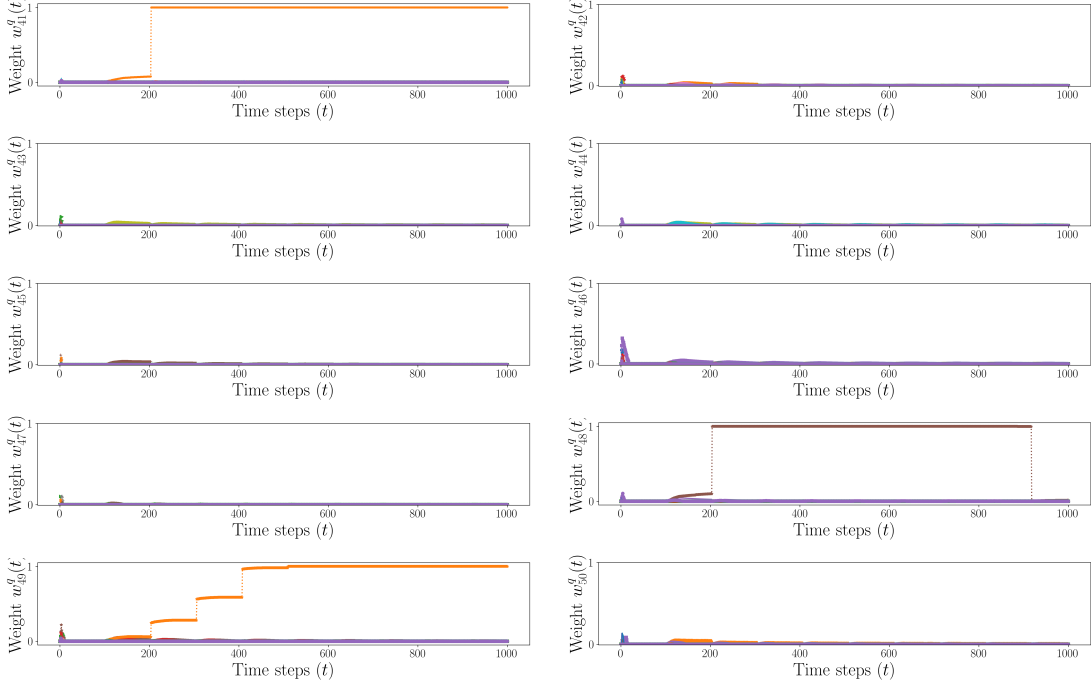


Figure 4.9. d-PBRAG for a large number of agents on a cyclic communication graph. Weight evolution of agents 41-50.

We performed a similar experiment with the distributed Hungarian algorithm which takes $O(|\mathcal{A}|^3)$ steps to converge, for a fixed reward scheme. This means that with our chosen number of agents, the algorithm would take ≈ 12500 steps to converge. Since this is a large order-of-magnitude estimate, we have evaluated in simulation the particular number of steps it takes in this example. Observe the time evolution of the assignments, up till ≈ 1250 time steps, in Figure 4.10. Here, 1 means that the task is assigned to the agent and 0 means that it is not. The simulations are solving the assignment with respect to *the static reward* that the agents perceived at $t = 0$. That is, this corresponds to a first assignment with initial task evaluations, which takes about 600 steps in our algorithm. Therefore, this algorithm will take much longer to converge in the time-varying rewards scenario that we have simulated above with our algorithm.



Figure 4.10. Assignment evolution of all agents using distributed Hungarian method. The agents are arranged in order from left to right and top to bottom.

4.10 Summary

In this chapter, we presented a game theoretic formulation of an optimal task allocation problem for a group of agents. By allowing agents to assign weights between *zero* and *one* for each task, we relaxed the combinatorial nature of the problem. This led to a partition and weight game, whose NE formed a superset of the optimal task partition. Then, we provided a distributed best-response projected gradient ascent by which convergence to the NE of the weight game was guaranteed.

Future work will consider constraints on number of tasks for each agent, and generalizing the setup to continuous space of tasks and classes of tasks. Further, analysis will be done on whether this setup enforces truth-telling among agents.

This chapter is a reprint of material as it appears in “Distributed task allocation for self-interested agents with partially unknown rewards,” N. Mandal, M. Khajenejad, and S. Martínez, *IEEE Transactions on Automatic Control*, 70(9):6284–6291, 2025. The dissertation author was the primary investigator and author of all of this paper.

Chapter 5

Behaviorally Heterogeneous Multi-Agent Exploration

5.1 Introduction

Exploring uncertain, hazardous environments is often handled by the coordination of multiple autonomous robots, with the goal of improving efficiency, adaptiveness, and robustness to individual failures. As shown in Figure 5.1, exploration frameworks are usually broken down into a) perception (robots gather position and environment knowledge using SLAM [82, 83]), b) reasoning (robots evaluate AoIs—areas of interest—using available knowledge), c) communication and assignment (robots discuss and assign tasks or AoIs among themselves), and d) navigation (robots plan and travel to selected AoIs). Robotic perception [84, 85], and navigation [86, 87] have been widely studied across multiple communities. With regards to decision making, it is evident that human teams with differing views [88] can fare better in heterogeneous complex scenarios. However, the adoption of human-like behaviors in robot coordination remains a nascent area of research. Our framework (outlined in Figure 5.1) aims to evaluate this question in the context of task allocation for unknown map exploration. We consider Behavioral Entropy [89], a human-inspired information metric having the potential of creating more powerful synergies.

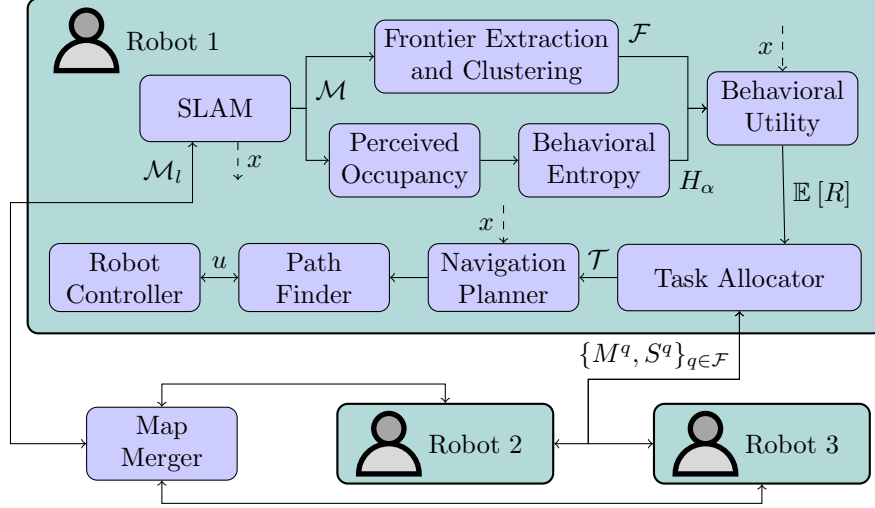


Figure 5.1. Proposed exploration framework with 3 robots in close proximity. Proximity determines the communication architecture. Each robot shares the same internal logic framework as in the teal box. The SLAM system provides a local map $\mathcal{M}_l$, a global map $\mathcal{M}$ (using an external map merger scheme) and robot localization x . Then, frontiers $\mathcal{F}$ (clustered using some scheme) and perceived occupancy are extracted from $\mathcal{M}$ and further Behavioral Entropy H_α is calculated for each cluster. This, along with the distance to the cluster, gives the sample Behavioral Utility $\mathbb{E}[R]$. Then, utilities are communicated with other robots to allocate the appropriate frontiers. The navigation planner takes these allocations and finds a traveling salesman tour among them. The path finder then finds a path to the frontier using RRT, which, in turn, sends global and local plans u to the controller.

5.2 Literature Review

There is a rich literature that applies SLAM techniques for unknown environment exploration using visual odometry [84] and LiDAR [85]; as well to handle navigation pipelines [86, 87] to safely reach an AoI. Here, we rely on the popular frontier-based exploration algorithm [90], which has been deployed successfully in challenging environments [91].

AoIs are typically evaluated in terms of their potential information gain. Often, Shannon entropy [92] is chosen as the metric of uncertainty. In [89], Behavioral Entropy (BE), a generalized entropy, was proposed to subjectively measure uncertainty using models of uncertainty perception [93] from the Behavioral Economics literature. It was shown [89, 94] that BE provides a more effective, yet flexible way to evaluate uncertainty in exploration tasks, leading to faster and better coverage compared to standard entropy. Additionally, being human-inspired makes it flexible and adaptable to various priorities and mapping schemes.

In order to efficiently assign AoIs among robots, we look towards the literature on task allocation [59]. Both centralized solutions [60] and distributed solutions [61] are known to be computationally and memory-intensive for large problems. Moreover, they are mostly addressed in static scenarios. Often authors [64] restrict the type of functions in order to develop a tractable solution to the NP-hard problem. Game-theoretic models have also been proposed to find solutions to task allocation problems, where each agent is equipped with an appropriate utility function [77], and the optimal task allocation is related to the Nash equilibrium of this game. Any Nash-seeking [21] algorithm returns a solution; but often, these algorithms require strong assumptions on the utility functions and their derivatives. Often, complete and perfect information on agents' utilities is required to run the algorithm updates, while in practice only imperfect information about tasks and other agents' capabilities is available. Further, to address the lack of information,

each agent applies consensus [22], or gossip-based algorithms [25] to estimate all other agents' strategies and compute the gradient of its own utility. There is still a need for scalable and distributed solutions that dynamically adapt to rewards revealed online with performance guarantees. This is studied in [95], but for deterministic reward sequences. Here, we deal with more realistic stochastic approximation sequences.

5.3 Contributions

We provide a novel distributed task allocation algorithm based on a game-theoretic interpretation of the classical task allocation problem. The utility of the robots is given by the expected value of a random variable with an unknown distribution. Compared to other works, we give probabilistic convergence guarantees for band estimations of this utility. We also give out-of-sample guarantees when the utility is approximated using i.i.d. samples of the random variable. Further, we extend the idea of BE to a team of agents. This allows us study a large variety of heterogeneous and homogeneous teams and give provable justification of the benefits of heterogeneity for exploration. We simulate various teams of robots mapping a complex environment and gather performance results based on time taken and path-length traveled. This allows us to provide heuristic guidelines for choosing the team composition in terms of behaviors for a multi-agent exploration team. We also show that our exploration scheme is superior against standard baselines and show scalability with different robot team sizes.

5.4 Preliminaries

Here, we recall the concepts from Chapter 2 that are used throughout this chapter.

Notations

The sets of real numbers, non-negative real numbers, and non-negative integers are denoted as $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, and $\mathbb{Z}_{>0}$, respectively. For a set $\mathcal{S}$, we denote by $|\mathcal{S}|$ its cardinality,

$2^{\mathcal{S}}$ its power set, by $\mathcal{S}^n$ its n Cartesian product, and by $\mathcal{S}^{n \times m}$ is the collection of $n \times m$ matrices whose $(i, j)^{\text{th}}$ entry lies in $\mathcal{S}$. For $x \in \mathbb{R}$, $[x]_0^1 := \max\{0, \min\{x, 1\}\}$. Given $\mathcal{S}$, we define $\max^{(2)} \mathcal{S} := \max\{s \in \mathcal{S} \mid s \neq \max \mathcal{S}\}$. The probability of an event (over a measurable space, which should be clear from the context) is $\Pr\{\cdot\}$.

Game theoretic notions

A *strategic form game* [10] is a tuple $\mathcal{G} := \langle \mathcal{A}, \{\mathcal{S}_i\}_{i \in \mathcal{A}}, \{\psi_i\}_{i \in \mathcal{A}} \rangle$ consisting of a set of *players* (or *agents* or *robots*) $\mathcal{A}$; a set of *strategies* $s_i \in \mathcal{S}_i$ available to each $i \in \mathcal{A}$; and a set of *utility functions* $\psi_i : \times_{i \in \mathcal{A}} \mathcal{S}_i \rightarrow \mathbb{R}$ over the strategy profiles of all the agents. Denote s_{-i} as the strategy profile of all players other than $i \in \mathcal{A}$. This helps to define the Nash equilibria next.

Definition 5.4.1 (Nash equilibrium). The strategy profile $(\hat{s}_i, \hat{s}_{-i})$ is a Nash equilibrium (NE) of $\mathcal{G}$ if and only if

$$\psi_i(\hat{s}_i, \hat{s}_{-i}) \geq \psi_i(s_i, \hat{s}_{-i}), \quad \forall s_i \in \mathcal{S}_i, \quad \forall i \in \mathcal{A}.$$

$\mathcal{NE}(\mathcal{G})$ denotes the set of all Nash equilibria of $\mathcal{G}$. •

Graph theoretic notions

A *directed graph* $\mathcal{G} := (\mathcal{A}, \mathcal{E})$, is a tuple consisting of a set of *nodes* (here robots, $\mathcal{A}$), and a set of *arcs* $\mathcal{E} \subseteq \mathcal{A} \times \mathcal{A}$ between the nodes; see [8]. The set $\mathcal{N}_i := \{j \in \mathcal{A} \mid (j, i) \in \mathcal{E}\}$ denotes the set of (in) *neighbors* of agent $i \in \mathcal{A}$ and $\overline{\mathcal{N}}_i := \mathcal{N}_i \cup \{i\}$. A *path* in $\mathcal{G}$ is an ordered set of non-repeated nodes such that each pair of adjacent nodes defines an edge. The graph $\mathcal{G}$ is said to be *strongly connected* if there exists a path from every node to every other node. The *diameter* of the graph $\text{diam}(\mathcal{G})$ is the length of the largest possible path between any two nodes.

Concentration of measure

Suppose that $\xi \in \mathbb{R}^m$ is a random variable with an unknown probability distribution, $\mathbb{P}^*$. Assume that $\{\widehat{\xi}(k) \in \mathbb{R}^m\}_{k=1}^N$ are i.i.d samples generated by this unknown distribution, $\mathbb{P}^*$. In addition, suppose that $\mathbb{P}^*$ is supported on $\Xi \subseteq \mathbb{R}^m$ and $\exists a > 1$ such that $\mathbb{E}_{\xi \sim \mathbb{P}^*} [\exp(\|\xi\|^a)] < \infty$ (i.e. $\mathbb{P}^*$ is light-tailed). Let $\mathcal{L}(\Xi)$ be the set of all probability distributions $\mathbb{Q}$ on Ξ with bounded first moment. Then, the 1-Wasserstein distance between two distributions $\mathbb{Q}_1, \mathbb{Q}_2 \in \mathcal{L}(\Xi)$ is defined as

$$d_W(\mathbb{Q}_1, \mathbb{Q}_2) := \inf_{\pi \in \Pi(\mathbb{Q}_1, \mathbb{Q}_2)} \int_{\Xi^2} \|\xi_1 - \xi_2\|_1 \pi(d\xi_1, d\xi_2),$$

where, $\Pi(\mathbb{Q}_1, \mathbb{Q}_2)$ is the set of joint probability distributions of ξ_1 and ξ_2 with marginals $\mathbb{Q}_1$ and $\mathbb{Q}_2$ respectively. Further,

$$\mathcal{B}_\varepsilon(\mathbb{P}) := \{\mathbb{Q} \in \mathcal{L}(\Xi) \mid d_W(\mathbb{Q}, \mathbb{P}) < \varepsilon\},$$

is the Wasserstein ball of radius ε around $\mathbb{P} \in \mathcal{L}(\Xi)$.

Recall that 2.2.1 characterizes the closeness of $\mathbb{P}^*$ to the sample average distribution $\widehat{\mathbb{P}}_N := \frac{1}{N} \sum_{k=1}^N \delta_{\widehat{\xi}(k)}$; is useful in distributional robust optimization [2].

5.5 Problem Formulation

Here, we first describe the constituents of our framework and then define the problem statement.

5.5.1 Exploration environment

A team of n robots $\mathcal{A} := \{1, \dots, n\}$ is to explore an unknown environment $\mathcal{X} \subset \mathbb{R}^2$ in an efficient and distributed fashion. The environment $\mathcal{X}$ is discretized into grid elements $\mathcal{D}$ that define an occupancy map $\mathcal{M}$. More precisely, $\mathcal{M}$ is defined in terms of a probability

function $f_{\text{occ}} : \mathcal{D} \rightarrow [0, 1]$, which assigns a probability of occupancy to each cell in $\mathcal{D}$ (*i.e.* $\mathcal{M} = f_{\text{occ}}(\mathcal{D})$). In what follows, we assume there is a subroutine that takes care of map merging (either centralized or distributed) and that $\mathcal{M}$ is available to all robots. Instead, we focus on the distributed exploration part of the approach. As the robots explore, they aim to reduce the volume of the unknown set $\{x \in \mathcal{D} \mid f_{\text{occ}}(x) \in (0, 0.5) \cup (0.5, 1)\}$ while increasing the volume of the known set $\{x \in \mathcal{D} \mid f_{\text{occ}}(x) \in \{0, 1\}\}$. Ideally, the team completes the exploration when $\mathcal{D}$ becomes known. In practice, limited observability due to the sensing noise and presence of obstacles hinders uncertainty reduction.

5.5.2 Robot Team

Each robot $i \in \mathcal{A}$ uses a Bayesian algorithm to update the occupancy map in its immediate environment as well as its position, $x_i \in \mathcal{D}$; see [96, 85]. To update the map, each robot $i \in \mathcal{A}$ identifies a set of frontiers $\mathcal{Q}_i$ from the current $\mathcal{M}$. We note that the size of $\mathcal{Q}_i \equiv \mathcal{Q}_i(t)$, $i \in \mathcal{A}$, may vary with time as robots discover frontiers to visit at time t (we omit the time argument when clear from the context). To aid in exploration, robot $i \in \mathcal{A}$ associates a reward of $\rho_i(q) := \mathbb{E}[R_i^q]$ for $q \in \mathcal{Q}_i$. Here R_i^q is a random variable with an unknown distribution, the expectation of which is to be estimated. We give further details of this procedure in Section 5.6. Randomness may appear either due to noisy sensor measurements or environmental uncertainty. Note that as these rewards are robot-dependent, they introduce heterogeneity, which will influence exploration.

5.5.3 Frontier assignment problem

Given $t \in \mathbb{Z}_{\geq 0}$, assume that $\{\mathcal{Q}_i(t)\}_{i \in \mathcal{A}}$ is the set of discovered frontiers at time t , and define $\mathcal{Q}(t) := \bigcup_{i \in \mathcal{A}} \mathcal{Q}_i(t)$. The optimal frontier assignment at time t is the solution

to

$$\max_{\mathcal{P}=(\mathcal{V}_1,\dots,\mathcal{V}_n)\subseteq \times_{i=1}^n \mathcal{Q}_i(t)} J(\mathcal{P}) := \sum_{i \in \mathcal{A}} \sum_{q \in \mathcal{V}_i} \rho_i(q) \quad (5.1a)$$

$$\text{s.t. } \bigcup_{i \in \mathcal{A}} \mathcal{V}_i = \mathcal{Q}(t); \quad \mathcal{V}_i \cap \mathcal{V}_j = \emptyset, \text{ if } i \neq j. \quad (5.1b)$$

Here, $\mathcal{V}_i \subseteq \mathcal{Q}_i(t)$ is the set of all frontiers assigned to robot $i \in \mathcal{A}$ and $\mathcal{P} = (\mathcal{V}_1, \dots, \mathcal{V}_n)$ is the ordered collection of sets that defines a partition of $\mathcal{Q}(t)$ (as in (5.1b)). The set $\mathcal{O}(\mathcal{A}, \mathcal{Q}, \{\rho_i(q)\}_{q \in \mathcal{Q}, i \in \mathcal{A}})$ collects the optimizers of (5.1). To ensure scalability to large environments, robots can only choose reachable frontiers within region $\mathcal{Q}_i \subseteq \mathcal{Q}_i$, around each robot's location. However, this might result into a robot $i \in \mathcal{A}$ running out of frontiers (*i.e.* $\mathcal{Q}_i = \emptyset$). When this happens, we allow the robot to expand its footprint radius until some frontiers become available (*i.e.* $\mathcal{Q}_i \neq \emptyset$) to explore. Thus, this adaptive technique trades off manageable computations with potential ineffectiveness.

After assignment, each robot keeps a list $\mathcal{T}_i$ of frontiers (a task buffer) that it is scheduled to visit. We assume that the task-assignment problem is re-solved after a robot completes the tasks in its buffer (*i.e.* $\mathcal{T}_i = \emptyset$). Thus, triggered by one robot at time t_e , all other robots will share with neighbors in a communication graph the frontiers that they have not been visited yet, as well as the new ones they have discovered, entering a frontier re-assignment phase. To do this, we will assume that the robot-communication graph becomes connected over time; see Assumption 5.5.1, so there is a time $t > t_e$ when the assignment problem (5.1) is resolved. The robot with an empty buffer then updates $\mathcal{T}_i$ using $\mathcal{O}(\mathcal{A}, \mathcal{Q}(t_e), \{\rho_i(q)\}_{q \in \mathcal{Q}, i \in \mathcal{A}})$. We will assume that the discovery of new frontiers happens at a sufficiently slow pace for the fast re-computation of the solution to (5.1). We focus on understanding the benefits of exploration with heterogeneous rewards, leaving the question of efficient dynamic task assignment under partial information for future work.

5.5.4 Communication network

The robots communicate with each other over a time-varying graph $\mathcal{G}(t) := (\mathcal{A}, \mathcal{E}(t))$. The arc set $\mathcal{E}(t)$ defines the connections between robots at time $t \in \mathbb{Z}_{\geq 0}$: in this way, $(i, j) \in \mathcal{E}(t)$ if and only if robot $i \in \mathcal{A}$ can send information to robot $j \in \mathcal{A}$ at time t , which can only occur when they are within a distance of each other. Further we assume the following.

Assumption 5.5.1. (Periodic fast communication). *For the communication network sequence $\{\mathcal{G}(t) = (\mathcal{A}, \mathcal{E}(t))\}_{t \in \mathbb{Z}_{\geq 0}}$, there exists a (known) $\tau \in \mathbb{Z}_{\geq 0}$ such that the graph $\bar{\mathcal{G}} := (\mathcal{A}, \bar{\mathcal{E}}(k))$, with $\bar{\mathcal{E}}(k) := \bigcup_{t=1}^{\tau} \mathcal{E}(k\tau + t)$, is strongly connected $\forall k \in \mathbb{Z}_{\geq 0}$. Further, robot communication happens at a faster time scale than robot movement.* •

5.5.5 Problem Statement

Given the aforementioned setup,

1. Provide a robust and distributed frontier allocation algorithm that converges to an stochastic optimal frontier assignment under time-varying communications.
2. Study the benefits of BE heterogeneity on exploration.
3. Verify the performance of the algorithms in complex, high-fidelity simulation environments and provide heuristic guidelines on choosing heterogeneous parameters in a mixed group autonomous agents, possibly including humans.

5.6 Distributed Frontier Assignment

Here, we first tackle the problem of efficient frontier allocation. This works for any evaluation scheme of frontiers. We provide the specific frontier utility metric in Section 5.7.

In order to assign frontiers in a distributed fashion, we generalize the distributed projected best-response ascending gradient (d-PBRAG) dynamics proposed in [95]. The

following definition from [95] helps us in characterizing the optimizers of (5.1), $\mathcal{O}(t)$ at a particular $t \in \mathbb{Z}_{\geq 0}$.

Definition 5.6.1. (*Frontier specific dominating agent*). A robot $i \in \mathcal{A}$ is said to *dominate frontier* $q \in \mathcal{Q}(t)$, if $q \in \mathcal{Q}_i(t)$ and $\rho_i(q) \geq \rho_j(q)$, $\forall j \in \mathcal{A}$ such that $q \in \mathcal{Q}_j(t)$. If a frontier $q \in \mathcal{Q}$ has exactly one dominating robot, we say that there exists a *unique dominating agent* for q . •

In [95] it was shown that if there is a unique dominating agent per $q \in \mathcal{Q}(t) = \bigcup_{i \in \mathcal{A}} \mathcal{Q}_i(t)$, then $\mathcal{O}(t)$ is in fact a singleton set. Moreover, it is also the unique Nash equilibrium of the game

$$\mathcal{G} := \langle \mathcal{A}, \{\mathbf{w}_i\}_{i \in \mathcal{A}}, \{U_i\}_{i \in \mathcal{A}} \rangle, \quad (5.2)$$

where robot $i \in \mathcal{A}$ is equipped with the utility function,

$$U_i(\mathbf{w}_i, \mathbf{w}_{-i}) = \sum_{q \in \mathcal{Q}_i(t)} \left[\rho_i(q) w_i^q - \max_{\substack{j \in \mathcal{A} \setminus \{i\} \\ q \in \mathcal{Q}_j}} \rho_j(q) w_j^q w_i^q \right], \quad (5.3)$$

and the strategies available to robot $i \in \mathcal{A}$ are $(w_i^q \in [0, 1])_{q \in \mathcal{Q}_i(t)}$, with $\mathbf{w}_i = (w_i^q)_{q \in \mathcal{Q}_i(t)}$. A weight $w_i^q = 1$ (resp. $w_i^q = 0$), loosely means that agent $i \in \mathcal{A}$ is assigned (resp. not assigned) the frontier $q \in \mathcal{Q}_i(t)$. Note that sufficiently small and constant perturbations to individual rewards can break down ties in allocation and create unique dominating agents per task, aiding in the frontier allocation problem. We quickly state and prove this in the next result.

Lemma 5.6.2. (Perturbation for unique dominating agent). *Let $\mathcal{O}_1 := \mathcal{O}(\mathcal{A}, \mathcal{Q}, \{\rho_i(q)\}_{q \in \mathcal{Q}, i \in \mathcal{A}})$ be the set of solutions the optimization problem in (5.1). Let $\varepsilon > 0$ and let $\mathcal{O}_2 :=$*

$\mathcal{O}(\mathcal{A}, \mathcal{Q}, \{\bar{\rho}_i(q)\}_{q \in \mathcal{Q}, i \in \mathcal{A}})$ be the set of solutions the optimization problem in (5.1) with

$$\bar{\rho}_i(q) = \rho_i(q) + \delta_i^q, \quad \delta_i^q \sim \text{Unif}[-\varepsilon, \varepsilon]. \quad (5.4)$$

Then, $\exists \bar{\varepsilon} > 0$, such that $\forall \varepsilon \leq \bar{\varepsilon}$, with probability one, $\forall q \in \mathcal{Q}$, there is a unique dominating agent for frontier q (hence $|\mathcal{O}_2| = 1$) and $\mathcal{O}_2 \subseteq \mathcal{O}_1$.

Proof. Consider an arbitrary but fixed $q \in \mathcal{Q}$. We prove the statement using the following cases.

Case (i): $\rho_i(q) = \rho$, $\forall i \in \mathcal{A}$. Then, since the δ_i^q 's are drawn independently from $\text{Unif}[-\varepsilon, \varepsilon]$, $\Pr\{\delta_i^q = \delta_j^q\} = 0$ for any choice of $j \neq i \in \mathcal{A}$. Then uniqueness of dominating agent holds from the definition of a dominating agent. Moreover, let $(\mathcal{V}_i, \dots, \mathcal{V}_n)$ be such that $q \in \mathcal{V}_i$ iff $i = \arg\max_{j \in \mathcal{A}} \bar{\rho}_j(q)$. Then it is easy to see that $\mathcal{O}_2 = (\mathcal{V}_i, \dots, \mathcal{V}_n) \subseteq \mathcal{O}_1$.

Case (ii): $\rho_i(q) \neq \rho_j(q)$ for some $i, j \in \mathcal{A}$. Then choose $\bar{\varepsilon} < 0.5 \left| \max_{i \in \mathcal{A}} \rho_i(q) - \max_{i \in \mathcal{A}}^{(2)} \rho_i(q) \right|$. By similar arguments as before, $\Pr\{\delta_i^q = \delta_j^q\} = 0$ for any choice of $j \neq i \in \mathcal{A}$. Then, again, uniqueness of dominating agent holds from the definition of a dominating agent. Moreover, the choice of $\bar{\varepsilon}$ makes it so that $\max_{i \in \mathcal{A}} \bar{\rho}_i(q) \subseteq \max_{i \in \mathcal{A}} \rho_i(q)$. Thus with $(\mathcal{V}_i, \dots, \mathcal{V}_n)$ such that $q \in \mathcal{V}_i$ iff $i = \arg\max_{j \in \mathcal{A}} \bar{\rho}_j(q)$; it is easy to see that $\mathcal{O}_2 = (\mathcal{V}_i, \dots, \mathcal{V}_n) \subseteq \mathcal{O}_1$. ■

In this work, however, we show in Section 5.7 how agent heterogeneity can naturally encode this feature.

Recall that $\rho_i(q) = \mathbb{E}[R_i^q]$ is the reward that robot $i \in \mathcal{A}$ gets for $q \in \mathcal{Q}_i$. We assume that the random variables $\{R_i^q\}_{i \in \mathcal{A}, q \in \mathcal{Q}}$ are independent of each other for all i, q . In most real world scenarios, robots do not have access to the actual distribution of rewards, and just receive samples of them. Using these, we assume that the robot can approximate its reward with some sequence $\{z_i^q(t)\}_{t \in \mathbb{Z}_{\geq 0}}$ (e.g. a sampled average value). With this in mind, we define the following.

Definition 5.6.3. (*Probabilistic band convergence*). We say that a sequence of random

variables $\{z_i^q(t)\}_{t \in \mathbb{Z}_{\geq 0}}$ converges in probability to a μ -band around $\rho_i(q) \in \mathbb{R}$, with $\mu \geq 0$, if $\forall \varepsilon > 0$, $\Pr \{|z_i^q(t) - \rho_i(q)| - \mu > \varepsilon\} \rightarrow 0$ as $t \rightarrow \infty$. $\bullet$

Note that if the reward $\rho_i(q)$ is some other function (and not the expected value) of the random variable R_i^q , then the previous definition is in line with the standard notion of concentration estimation of $\rho_i(q)$. Since we are interested in characterizing the convergence w.r.t. the sample size, we choose the expected value as the reward.

Recall that the robots communicate over a time-varying graph $\mathcal{G}(t) = (\mathcal{A}, \mathcal{E}(t))$. This allows us to allocate the frontiers in a distributed manner while providing an algorithm that seeks the NE of $\mathcal{G}$ in (5.2), and accounting for increasingly more accurate approximations $\{z_i^q(t)\}_{t \in \mathbb{Z}_{\geq 0}}$ of $\rho_i(q)$, for $i \in \mathcal{A}$. Thus, to compute their utility gradients and update strategies w_i^q simultaneously, the robots make use of the following dynamics given in [95]:

$$w_i^q(t+1) = \left[w_i^q(t) + \gamma_i^q(t) \left(z_i^q(t) - \frac{1}{2} (M_i^q(t) + S_i^q(t)) \right) \right]_0^1, \quad (5.5a)$$

$$M_i^q(t+1) = \sigma_{\text{sw}} \left(\max_{j \in \mathcal{N}_i} M_j^q(t), e_i^q(t+1), t+1, T \right), \quad (5.5b)$$

$$S_i^q(t+1) = \sigma_{\text{sw}} \left(\max^{(2)} \left\{ \{S_j^q(t)\}_{j \in \mathcal{N}_i}, M_i^q(t), e_i^q(t) \right\}, e_i^q(t+1), t+1, T \right), \quad (5.5c)$$

$$e_i^q(t+1) = \sigma_{\text{sw}} \left(e_i^q(t), z_i^q(t+1), t+1, T \right), \quad (5.5d)$$

for some $T \in \mathbb{R}_{\geq 0}$, and where σ_{sw} is the switching function

$$\sigma_{\text{sw}}(m, z, t, T) := \begin{cases} z, & \text{if } t \bmod T = 0, \\ m, & \text{otherwise.} \end{cases} \quad (5.6)$$

The previous dynamics combines a max consensus sub-routine with a periodic knowledge injection to steer the strategies of each agent towards the NE of $\mathcal{G}$.

Work in [95] considered deterministic rewards and a deterministic sequence converging to the true value of the rewards. It was shown that that by choosing the step size $\gamma_i^q(t)$ in a certain way, it is possible to send the weight $w_{i_q^*}^q(t)$ of the dominating agent (i_q^* for $q \in \mathcal{Q}_i$) to *one* while simultaneously reducing $w_j^q(t)$ (for $j \neq i_q^*$) to *zero*. This stems from the fact that the max consensus subroutine in (5.5b), (5.5c) stabilizes within a number of time steps dependent on the diameter of the communication graph and then the average $0.5(M_i^q(t) + S_i^q(t))$ increases (or decreases) the weights $w_i^q(t)$ appropriately. Going beyond the analysis provided in [95], we show next that this algorithm converges in probability to a solution of the task assignment problem (5.1) for a certain class of stochastic approximation sequences. To keep the notations clean, we present the arguments with $\mathcal{Q}_i(t) = \mathcal{Q}$, $\forall i \in \mathcal{A}$ (*i.e.* every robot has access to every frontier). The algorithm can be easily adapted to the partial frontier setup that this chapter deals with.

Theorem 5.6.4. (d-PBRAG convergence in probability). *Suppose the communication graph $\mathcal{G}(t)$ satisfies Assumption 5.5.1 with connectivity period τ . Suppose that each $q \in \mathcal{Q}$ has a unique dominating agent and let*

$$\mu^q < 0.5 (\max\{\rho_i^q\}_{i \in \mathcal{A}} - \max^{(2)}\{\rho_i^q\}_{i \in \mathcal{A}}), \quad \forall q \in \mathcal{Q}. \quad (5.7)$$

Further for each $i \in \mathcal{A}$, $q \in \mathcal{Q}$, assume that $z_i^q(t)$ converges in probability to a μ^q band around $\rho_i(q)$. Consider an initial condition $(\mathbf{w}_i(0) \in [0, 1]^{|\mathcal{Q}_i|})_{i \in \mathcal{A}}$, let $(\mathbf{w}_i(t))_{i \in \mathcal{A}}$ be the solution to (5.5) with $T > 2\tau + 1$, and

$$\gamma_i^q(t) = \begin{cases} \alpha_i^q(k) \geq 0, & \text{if } t \in \{kT, \dots, kT + 2\tau - 1\}, \\ \beta_i^q(k) > 0, & \text{if } t \in \{kT + 2\tau, \dots, kT + T - 1\}, \end{cases}$$

$\forall i \in \mathcal{A}$, $\forall q \in \mathcal{Q}$, with $k \in \mathbb{Z}_{>0}$. Further, for all $i \in \mathcal{A}$, $\forall q \in \mathcal{Q}$; take sequences $\alpha_i^q(k) \rightarrow 0$

as $k \rightarrow \infty$ and $\beta_i^q(k) \rightarrow \infty$ as $k \rightarrow \infty$. Then, for each $i \in \mathcal{A}$, and $\varepsilon > 0$,

$$\Pr \{ \|\mathbf{w}_i(t) - \bar{\mathbf{w}}_i\| > \varepsilon \} \rightarrow 0, \quad \text{as } t \rightarrow \infty, \quad (5.8)$$

where $(\{q \in \mathcal{Q}_i \mid \bar{w}_i^q = 1\})_{i \in \mathcal{A}} \subseteq \mathcal{O}(\mathcal{A}, \mathcal{Q}, \{\rho_i(q)\}_{q \in \mathcal{Q}, i \in \mathcal{A}})$ is a solution to (5.1), and $\bar{\mathbf{w}}_i := (w_i^q)_{q \in \mathcal{Q}_i}$.

Proof. Consider an arbitrary but fixed $q \in \mathcal{Q}$ and the dynamics (5). First, notice that the arguments in [95, Lemma 6.1] can be extended to periodically strongly connected graphs to conclude that $\forall k \in \mathbb{Z}_{>0}, \forall i \in \mathcal{A}, M_i^q(t) = \max_{i \in \mathcal{A}} z_i^q(kT), S_i^q(t) = \max_{i \in \mathcal{A}}^{(2)} z_i^q(kT), \forall t \in \{kT + 2\tau, \dots, (k+1)T - 1\} =: \mathcal{I}(k, \tau)$. Denote $\bar{\mathcal{I}}(T, \tau) := \{kT, \dots, kT + 2\tau - 1\}$. Now, Consider an $\varepsilon > 0$. Then, by the hypothesis, for any $\lambda > 0$, $\Pr \{ |z_i^q(t) - \rho_i^q| - \mu^q > \varepsilon \} < \lambda, \forall t \geq t_0$ for some t_0 . Then, with probability at least $1 - \lambda$, $\forall k \in \mathbb{Z}_{>0}$ such that $kT > t_0$, $\{w_i^q(t)\}_{t \in \mathcal{I}(k, \tau)}$ is increasing (resp. decreasing) if $i \in \operatorname{argmax}_{j \in \mathcal{A}} \rho_j^q =: i_q^*$ (resp. $i \notin \operatorname{argmax}_{j \in \mathcal{A}} \rho_j^q$). This is because of (7). Moreover, since $\beta_i^q(k) \rightarrow \infty, \forall k \in \mathbb{Z}_{>0}$ such that $kT > t_0$, $w_i^q(kT + T - 1) = 1$ (resp. $= 0$) if $i = i_q^*$ (resp. $i \neq i_q^*$). Finally, since $\alpha_i^q(k) \rightarrow 0, \min\{w_{i_q^*}^q(t)\}_{t \in \bar{\mathcal{I}}(k, \tau)} \rightarrow 1$ and $\max\{w_j^q(t)\}_{t \in \bar{\mathcal{I}}(k, \tau)} \rightarrow 0, \forall j \neq i_q^*$. Thus, $\Pr \{ \|\mathbf{w}_i(t) - \bar{\mathbf{w}}_i\| > \varepsilon \} < \lambda$ with $(\{q \in \mathcal{Q}_i \mid \bar{w}_i^q = 1\})_{i \in \mathcal{A}} \subseteq \mathcal{O}(\mathcal{A}, \mathcal{Q}, \{\rho_i(q)\}_{q \in \mathcal{Q}, i \in \mathcal{A}})$, since $q \in \mathcal{Q}$ has a unique dominating agent, completing the proof. Then, from [95, Theorem 6.4], it follows that starting from $(\mathbf{w}_i(t))_{i \in \mathcal{A}}$, there is $t_1 \geq t_0$ such that $\Pr \{ \|\mathbf{w}_i(t) - \bar{\mathbf{w}}_i\| > \varepsilon \} < \lambda, \forall t \geq t_1$. Moreover, $(\{q \in \mathcal{Q}_i \mid \bar{w}_i^q = 1\})_{i \in \mathcal{A}} \subseteq \mathcal{O}(\mathcal{A}, \mathcal{Q}, \{\rho_i(q)\}_{q \in \mathcal{Q}, i \in \mathcal{A}})$, since each $q \in \mathcal{Q}$ has a unique dominating agent, completing the proof. $\blacksquare$

Next, we provide further robustness guarantees for a particular estimator of the expected rewards. Let us denote the samples of the random variable R_i^q as $\{\hat{R}_i^q(k)\}_{k \in \{1, \dots, N_i\}}$. For each robot $i \in \mathcal{A}$ and each frontier $q \in \mathcal{Q}_i$, let the sample average distribution be $\hat{\mathbb{P}}_{N_i}^{i,q} := \frac{1}{N_i} \sum_{k=1}^{N_i} \delta_{\hat{R}_i^q(k)}$ where $\delta_{\hat{R}_i^q(k)}$ is the Dirac delta at $\hat{R}_i^q(k)$. In what follows, we assume

that the robots estimate the actual reward mean via the sequence of random variables

$$z_i^q(t) = \frac{1}{2} \left(\sup_{\mathbb{Q} \in \mathcal{B}_\varepsilon(\widehat{\mathbb{P}}_{N_i^t}^{i,q})} \mathbb{E}_{\xi \sim \mathbb{Q}}[\xi] + \inf_{\mathbb{Q} \in \mathcal{B}_\varepsilon(\widehat{\mathbb{P}}_{N_i^t}^{i,q})} \mathbb{E}_{\xi \sim \mathbb{Q}}[\xi] \right). \quad (5.9)$$

where the ε are chosen properly. Here, N_i^t represents the number of samples available to $i \in \mathcal{A}$ at time $t \in \mathbb{Z}_{\geq 0}$ and recall that $\mathcal{B}_\varepsilon(\widehat{\mathbb{P}}_{N_i^t}^{i,q})$ is the appropriate Wasserstein ball. Thus, the robot estimates its reward for a task as the average of the worst case expectations (in either direction) generated by the probability distributions that are at most ε distance from its sample average. Further, if N_i^t increases, the robot can get better estimates using a larger sample size. In the next result, we give distributionally robust guarantees on the convergence of the d-PBRAG algorithm.

Theorem 5.6.5. (Finite sample guarantees with distributionally robust d-PBRAG).

Suppose $\forall i \in \mathcal{A}$, $N_i^t \rightarrow K_i \in \mathbb{Z}_{\geq 0}$. Moreover, suppose $\theta \in (0, 1)$ is chosen such that with ε in (2.4), we have that, for each $q \in \mathcal{Q}$, $\exists i_q^* \in \mathcal{A}$ such that

$$\sup_{\mathbb{Q} \in \mathcal{B}_\varepsilon(\widehat{\mathbb{P}}_{N_j^t}^{j,q})} \mathbb{E}_{\xi \sim \mathbb{Q}}[\xi] < \inf_{\mathbb{Q} \in \mathcal{B}_\varepsilon(\widehat{\mathbb{P}}_{N_{i_q^*}^t}^{i_q^*,q})} \mathbb{E}_{\xi \sim \mathbb{Q}}[\xi], \quad \forall j \neq i_q^*. \quad (5.10)$$

Finally, consider an initial condition $(\mathbf{w}_i(0) \in [0, 1]^{|Q_i|})_{i \in \mathcal{A}}$ and let $(\mathbf{w}_i(t))_{i \in \mathcal{A}}$ be the solution to (5.5) with T and $\gamma_i^q(t)$ as in Theorem 5.6.4, and $z_i^q(t)$ as in (5.9). Then, with probability at least $(1 - \theta)$, $(\mathbf{w}_i(t))_{i \in \mathcal{A}}$ converge in probability to a 0 band around $(\overline{\mathbf{w}}_i)_{i \in \mathcal{A}}$; where, $(\{q \in \mathcal{Q}_i \mid \overline{w}_i^q = 1\})_{i \in \mathcal{A}} \subseteq \mathcal{O}(\mathcal{A}, \mathcal{Q}, \{\mathbb{E}[\xi_i^q]\}_{q \in \mathcal{Q}, i \in \mathcal{A}})$ is a solution to (5.1).

Proof. Suppose the hypothesis is true. From Theorem 2.2,

$$\Pr \left\{ \inf_{\mathbb{Q} \in \mathcal{B}_\varepsilon(\widehat{\mathbb{P}}_{N_i^t}^{i,q})} \mathbb{E}_{\xi \sim \mathbb{Q}}[\xi] \leq \mathbb{E}[R_i^q] \leq \sup_{\mathbb{Q} \in \mathcal{B}_\varepsilon(\widehat{\mathbb{P}}_{N_i^t}^{i,q})} \mathbb{E}_{\xi \sim \mathbb{Q}}[\xi] \right\} \geq 1 - \theta,$$

where $\mathbb{E}[R_i^q] = \mathbb{E}_{\xi \sim \mathbb{P}^*}[R_i^q]$. Moreover, using (10) the samples $z_i^q(t)$, generated using (9),

converges in probability to a μ^q band around $\mathbb{E}[R_i^q]$, with μ^q satisfying (7). The claim now follows from Theorem 4.3. ■

The previous result states that when the robots only have access to finite samples from the original distribution, then it is possible to give a confidence interval, using θ , with which the converging strategies of the game $\mathcal{G}$ is an optimal frontier partition. We end this section by saying that if $N_i^t \rightarrow \infty$ in Theorem 5.6.5, then θ can be made arbitrary close to *zero*.

5.7 Heterogeneous Behavioral Exploration

Here, we first introduce the notion of generalized BE [89] to determine a system of weighted rewards per robot, ρ_i , $i \in \mathcal{A}$. Later we show the benefits of such heterogeneous behavior for exploration.

Given a discretization $\mathcal{D}$ of the environment, let $p_k \in [0, 1]$ (resp. $(1 - p_k) \in [0, 1]$) be the probability that the cell $k \in \mathcal{D}$ is a free space (resp. an obstacle).

Each robot perceives this uncertainty differently, and we use *probability weighting functions* to further encode this heterogeneity. To do this, we employ the popular Prelec's weighing function [93], $\omega : [0, 1] \rightarrow [0, 1]$,

$$\omega(p) = e^{-\beta(-\log p)^\alpha}, \quad \alpha > 0, \beta > 0, \omega(0) = 0, \quad (5.11)$$

which transforms a probability vector $[p, 1 - p]^\top$ into a perceived probability vector $[\omega(p), \omega(1 - p)]^\top$.

In order to evaluate the utility of visiting a frontier $q \in \mathcal{D}$, each robot will use the notion of BE H_α [89]. This expands the notion of Shannon's entropy by using Prelec's weights ω from (5.11) as

$$H_\alpha(p) := -\omega(p) \log(\omega(p)) - \omega(1 - p) \log(\omega(1 - p)). \quad (5.12)$$

Moreover, from [89], setting $\beta = \exp((1 - \alpha) \log(\log(2)))$ guarantees satisfaction of entropy axioms [92, 97]. In [89], it was evaluated how low α model an “uncertainty averse” behavior in exploration, where the robot has a lower threshold for uncertainty. With high α values we get “uncertainty insensitive” behavior, implying that the robot considers more certain outcomes. The robot computes this entropy per cell in its sensing region around the frontier $q \in \mathcal{D}$ to get the utility of q . We give more details next.

Information gain and utility function

A sensor footprint $\mathcal{Q} \subset \mathcal{D}$ is obtained around each frontier $q \in \mathcal{F}_i(t)$. This is a collection of occupancy cells according to the sensor model. The robots can use any standard LiDAR model [96] to update the costmap in their sensing area. Suppose a robot $i \in \mathcal{A}$ has a behavioral parameter $\alpha_i \in \mathbb{R}_{>0}$. Moreover, let $x_i(t) \in \mathcal{D}$ be the position of the robot at time $t \in \mathbb{Z}_{\geq 0}$ and let $\eta(x, y)$ denote the path length between two points $x, y \in \mathcal{D}$. Then for a frontier $q \in \mathcal{Q}_i(t)$, the robot associates

$$r_i(q; \mathcal{M}) := \sum_{k \in \mathcal{Q}} H_{\alpha_i}(p_k), \quad \phi_i(q; \mathcal{M}) := \frac{1}{\eta(x_i(t), q)} \quad (5.13)$$

and $R_i^q = r_i(q; \mathcal{M})\phi_i(q; \mathcal{M})$. Note that the r_i is the behavioral information gain which is calculated as the reduction in entropy from observing the area $\mathcal{Q}$ around the frontier $q \in \mathcal{Q}_i(t)$. Further, $\phi_i(q; \mathcal{M})$ makes it so that the robots find closer frontiers more lucrative than the farther ones.

Recall the robot relies on its sensors and particle filter like models approximate the actual of occupancy (p_k) of a cell $k \in \mathcal{D}$ based on generating samples $\hat{p}_k$ of this occupancy value. Thus, the robot associates $\rho_i(q) = \mathbb{E}[R_i^q] = \mathbb{E}[r_i(q; \mathcal{M})\phi_i(q; \mathcal{M})]$ as the reward for visiting the frontier. Moreover, the robot can only calculate a sample average approximation of r_i . As such, this ties in with the uncertain reward scenario described in Section 5.6. In what follows, to simplify the presentation, we assume that the robots

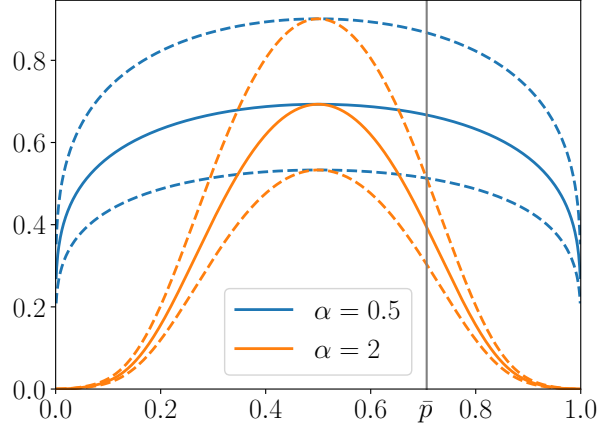


Figure 5.2. Scaled entropy vs. probability, *i.e.* $c.H_\alpha(p)$ vs. p . Upper (resp. lower) dotted lines have $c = d_M$ (resp. $c = d_m$). Bold line has $c = 1$.

have computed a good enough approximation of the frontier reward, $\rho_i(q)$, and focus our attention on the effect of heterogeneity induced by the behavioral parameters.

Mapping dynamics

Suppose that $\mathcal{D} = \{1, \dots, n\}$ is the set of finite cells that a robot can occupy. As each robot is only able to map an area around a frontier, then a team of $|\mathcal{A}|$ robots would only be able to map areas around $|\mathcal{A}|$ frontiers. Thus, assume that $n = |\mathcal{A}|$, in the subsequent discussion. Further, assume that the sensor footprint around cell $q \in \mathcal{D}$ obeys $\mathcal{Q} = \{q\}$, *i.e.* the robot can only map the cell it visits. We omit the discussion of more general footprints to reduce complexity yet highlight the effect of heterogeneity. Then, the mapping of the unknown environment obeys an algorithm of the form, from $x_i(0) \in \mathcal{D}$

$$x_i(t+1) = \underset{k \in \mathcal{D}}{\text{assign}} \frac{H_{\alpha_i}(p_k(t))}{\eta(x_i(t), k)}, \quad \forall i \in \mathcal{A}; \quad (5.14a)$$

$$p_k(t+1) = \begin{cases} \text{bel}(p_k(t)), & \text{if } k \in \{x_i(t+1)\}_{i \in \mathcal{A}}; \\ p_k(t), & \text{otherwise.} \end{cases} \quad (5.14b)$$

Here, $\text{assign}_{k \in \mathcal{D}} V(k)$ assigns the cell k with the highest value V (where, $V(k)$ is a vector of values associated with each cell grid) that is not the current position. For example, one way is to use the allocation problem (5.1) and ensure unique dominating agents.

In this framework, the belief update in (5.14b) corresponds to the act of mapping the cell $k \in \mathcal{D}$. This could be Bayesian update or some other form of update that incorporates noisy sensor measurements. Let $p_k^* \in \{0, 1\}$ be the ground truth occupancy of the cell k and assume that the mapping scheme is “good” in the sense of

$$|\text{bel}(p_k) - p_k^*| \leq c_M |p_k - p_k^*|, \quad (5.15)$$

for some $c_M \in [0, 1]$. Let the total entropy of the map be $\mathcal{H}(t) := \sum_{k \in \mathcal{D}} H_1(p_k(t))$, where H_1 is the BE with $\alpha = 1$ (*i.e.* Shannon’s entropy). This allows us to track the entropy reduction (from a good initial guess) under the exploration algorithm in the next result. The proof (which we skip) involves combining concave and differentiable properties to bound the change.

Lemma 5.7.1. (Convergence of the exploration algorithm). *Suppose $|p_k(0) - p_k^*| < 0.5$. Then, under (5.14) and (5.15),*

$$\begin{aligned} \mathcal{H}(t+1) - \mathcal{H}(t) \leq \sum_{i \in \mathcal{A}} \Big(\log(1 - p_{k_i^*}(t)) - \\ \log(p_{k_i^*}(t)) \Big) \left(\frac{1}{2} c_M^t - (p_{k_i^*}(t) - p_{k_i^*}^*) \right), \end{aligned} \quad (5.16)$$

where $k_i^* \in \arg\max_{k \in \mathcal{D}} H_{\alpha_i}(p_k(t)) / \eta(x_i(t), k)$. The right hand side of (5.16) is a negative quantity.

Proof. Since entropy is a concave function,

$$H_1(q) - H_1(p) \leq H_1'(p) \cdot (q - p), \quad \forall p, q \in (0, 1).$$

Further, since it is also differentiable,

$$H'_1(p) = \log(1 - p) - \log(p).$$

Thus, with $q = p_{k_i^*}(t + 1)$ and $p = p_{k_i^*}(t)$, we get

$$H_1(p_{k_i^*}(t + 1)) - H_1(p_{k_i^*}(t)) \leq \left(\log(1 - p_{k_i^*}(t)) - \log(p_{k_i^*}(t)) \right) \left(p_{k_i^*}(t + 1) - p_{k_i^*}(t) \right)$$

Moreover, from (13b) and (14) we have that

$$|p_{k_i^*}(t + 1) - p_{k_i^*}^*| = |\text{bel}(p_{k_i^*}(t)) - p_{k_i^*}^*| \leq c_M |p_{k_i^*}(t) - p_{k_i^*}^*| \leq \dots \leq c_M^t |p_{k_i^*}(0) - p_{k_i^*}^*| \leq \frac{1}{2} c_M^t.$$

The last inequality is because $|p_k(0) - p_k^*| < 0.5$. Thus $|p_{k_i^*}(t) - p_{k_i^*}^*| < 0.5, \forall t \in \mathbb{Z}_{\geq 0}$. Thus,

$$\begin{aligned} -\frac{1}{2} c_M^t &\leq p_{k_i^*}(t + 1) - p_{k_i^*}^* \leq \frac{1}{2} c_M^t, \\ \implies -\frac{1}{2} c_M^t - (p_{k_i^*}(t) - p_{k_i^*}^*) &\leq p_{k_i^*}(t + 1) - p_{k_i^*}(t) \leq \frac{1}{2} c_M^t - (p_{k_i^*}(t) - p_{k_i^*}^*) \end{aligned}$$

Now, there can be two cases.

Case (i): $p_{k_i^*}(t) > 0.5$. Then, $p_{k_i^*}(t+1) > p_{k_i^*}(t)$ and $(\log(1 - p_{k_i^*}(t)) - \log(p_{k_i^*}(t))) <$

0. Then,

$$\begin{aligned} &\left(\log(1 - p_{k_i^*}(t)) - \log(p_{k_i^*}(t)) \right) \left(p_{k_i^*}(t + 1) - p_{k_i^*}(t) \right) \\ &\leq \left(\log(1 - p_{k_i^*}(t)) - \log(p_{k_i^*}(t)) \right) \left(-\frac{1}{2} c_M^t - (p_{k_i^*}(t) - p_{k_i^*}^*) \right) \end{aligned}$$

Case (ii): $p_{k_i^*}(t) < 0.5$. Then, $p_{k_i^*}(t + 1) < p_{k_i^*}(t)$ and $(\log(1 - p_{k_i^*}(t)) - \log(p_{k_i^*}(t))) > 0$.

Then,

$$\begin{aligned} & \left(\log(1 - p_{k_i^*}^*(t)) - \log(p_{k_i^*}^*(t)) \right) \left(p_{k_i^*}^*(t+1) - p_{k_i^*}^*(t) \right) \\ & \leq \left(\log(1 - p_{k_i^*}^*(t)) - \log(p_{k_i^*}^*(t)) \right) \left(\frac{1}{2} c_M^t - (p_{k_i^*}^*(t) - p_{k_i^*}^{**}(t)) \right) \end{aligned}$$

Combining these gives the result. ■

We finish the section by analyzing how a multi-robot system with different α_i is beneficial for mapping of an unknown area. Essentially, under (5.14) and (5.15), each robot is guaranteed to visit a distinct location of $\mathcal{D}$ and continue the mapping process. This is elaborated in the remark later.

Proposition 5.7.2 (On the effect of heterogeneity). *Suppose that $\{\alpha_i\}_{i \in \mathcal{A}}$ is a set of distinct elements and define $d_M := \max_{i,j,k,l \in \mathcal{D}} \frac{\eta(i,j)}{\eta(k,l)}$, $d_m := \min_{i,j,k,l \in \mathcal{D}} \frac{\eta(i,j)}{\eta(k,l)}$. Further, for any two $i \neq j \in \mathcal{A}$, let $\bar{p}_{i,j} := \min \left(\{p \in (0,1) \mid d_M H_{\alpha_i}(p) = d_m H_{\alpha_j}(p)\} \cup \{p \in (0,1) \mid d_m H_{\alpha_i}(p) = d_M H_{\alpha_j}(p)\} \right)$, and define $\bar{p} := \min_{i,j \in \mathcal{A}; i \neq j} \bar{p}_{i,j}$. Suppose that for each $k \in \mathcal{D}$, $p_k(0) \in [0,1] \setminus [\bar{p}, 1 - \bar{p}]$. Let $(x_i(t))_{i \in \mathcal{A}}$ be the solution to (5.14) and (5.15) from any $x_i(0) \in \mathcal{D}$, for $i \in \mathcal{A}$. Then, for all $t \geq 0$, $x_i(t) \neq x_j(t)$ if $i \neq j \in \mathcal{A}$.*

Proof. Consider Figure 5.2, where we illustrate the allocation is distinct for two robots. First, note from Figure 5.2 that when $d_m = d_M = 1$, the entropy values for different α_i 's are the bold lines. Observe that, no matter what the p values are, because of the separation in the bold line entropy values, there is a distinct allocation of cells among the robots. Now, in general when $d_m \leq 1 \leq d_M$, the separation appears after $\bar{p}$, as shown in Figure 5.2. Thus the same argument holds. Finally note that (14) makes $([0,1] \setminus [\bar{p}, 1 - \bar{p}])^{|\mathcal{D}|}$ invariant under (13). Thus the argument can be made recursively. This completes the proof. ■

Remark 5.7.3 (Interpretation of the effect of heterogeneity). The range of occupancy values where a unique assignment is made depends on the choice of α_i 's and the distances

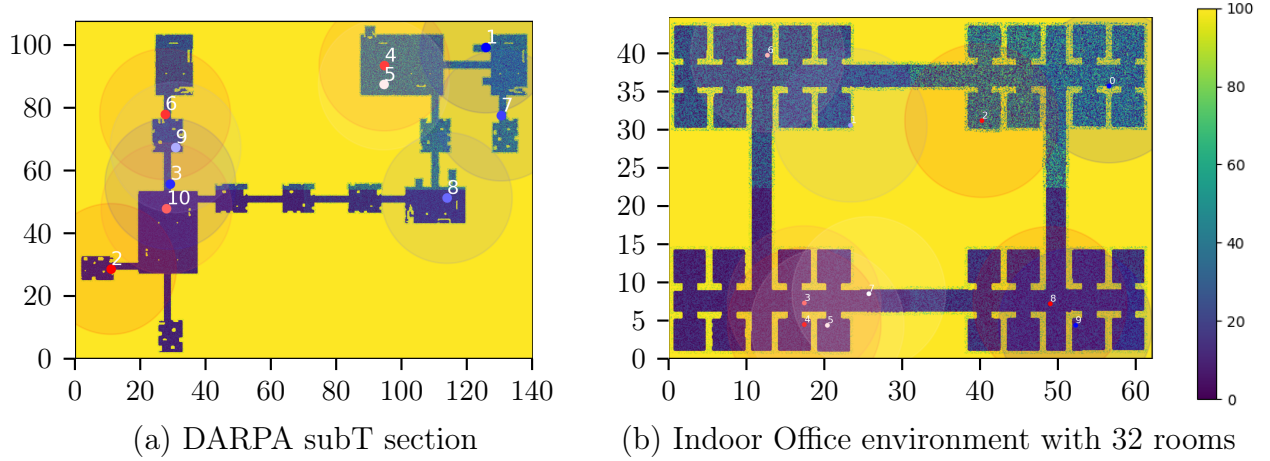


Figure 5.3. Indoor maps used for simulations. The plots use the same colorbar. The robot positions $x_i(0)$ is represented by dots and the circle around each robot shows its initial region of frontier consideration.

between the grid cells (see Figure 5.2 and Proposition 5.7.2). If d_m is much smaller than d_M , then a much larger value of α_i is required to make $[\bar{p}, 1 - \bar{p}]$ small. Mathematically, this is possible to do, but it introduces numerical issues as the entropy values become very small near the boundaries. As a result of Proposition 5.7.2, a proper choice of α_i 's leads to a unique cell allocation for robots and hence improve the efficiency of exploration. •

5.8 Simulations and Results

In this section, we implement the heterogeneous mapping algorithm on Python environments. First, we detail the simulation environments and the algorithm.

5.8.1 Environment Setup

We consider a 2D rectangular environments discretized by 0.1×0.1 unit square grid cells. We consider two types of environments, namely indoor environments (see Figure 5.3 for a map layouts) and outdoor environment (see Figure 5.4 for a map layout) to test our algorithm. The ground truth map $\mathcal{M}^*$ consists solely of free-space (0 occupancy value) and obstacles (100 occupancy value). The map in Figure 5.3(a) was used in the DARPA

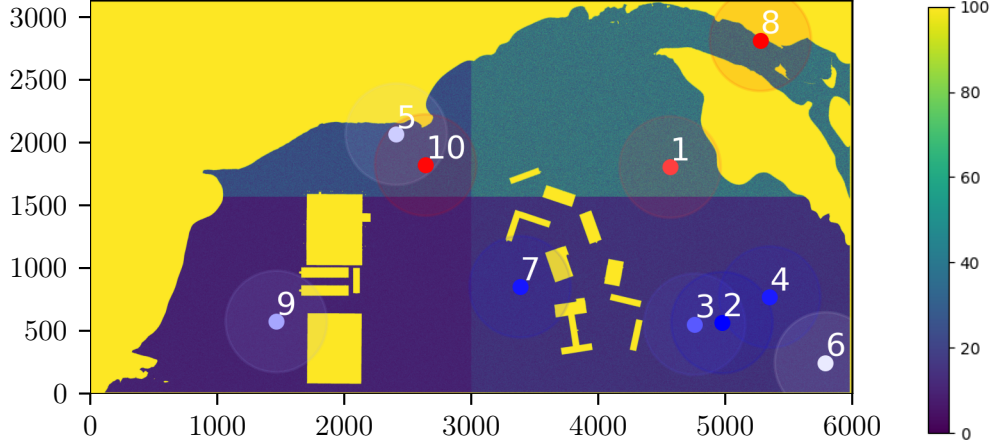


Figure 5.4. Outdoor camp map used for simulations. The robot positions $x_i(0)$ is represented by dots and the circle around each robot shows its initial region of frontier consideration.

Subterreanean challenge [98]. In Figures 5.3 and 5.4, the blue/dark areas (0) represent free space, the yellow areas (100) represent known obstacles and other values indicate uncertain areas. We construct the initial occupancy map $\mathcal{M}$ by adding noise to each cell value of the ground truth $\mathcal{M}^*$ (depending on which map we are using). The environment is divided into four quadrants which contain different levels of noise, sampled from a uniform distribution with different intervals from $[0, 50]$ (top-left quadrant), to $[0, 80]$ (top-right quadrant), and others with $[0, 30]$ (bottom-right) and $[0, 20]$ (bottom-left) (see Figures 5.3 and 5.4 for reference). This is to simulate the fact that different sections of the ground truth maps $\mathcal{M}^*$ can have different initial uncertainties. We do not add noise to the extreme areas of the map so that the robot remains within map boundaries. The initial robot positions are sampled uniformly from the free space in $\mathcal{M}^*$. One such sample initial condition is shown in Figures 5.3 and 5.4, where the robot positions $x_i(0)$ is represented by dots beside the robot ID (i). In order to simulate heterogeneity among robots we assign their α values

Algorithm 2. Behavioral explorer

```
1: Input :  $\mathcal{M}, x_i, K$  ; Output :  $f_{\text{occ}}(\mathcal{D})$ 
2:  $\text{taskBuffer}_i \leftarrow \emptyset$ 
3:  $\mathcal{Q}_i \leftarrow \text{getFrontier}(\mathcal{M}, K)$ 
4: while  $\mathcal{Q}_i \neq \emptyset$  do
5:   if  $\text{taskBuffer}_i = \emptyset$  then
6:      $\{w_i^q\}_{q \in \mathcal{Q}} \leftarrow \text{communicate}(\mathcal{Q}_i, \mathcal{N}_i)$ 
7:      $\text{tour} \leftarrow \text{pick}(\{q \mid w_i^q = 1\}, N)$ 
8:      $\text{tour} \leftarrow \text{travelingSalesman}(\text{tour})$ 
9:      $\text{taskBuffer}_i \leftarrow \text{tour}$ 
10:  end if
11:   $\text{path} \leftarrow \text{RRT}(x_i, \text{taskBuffer}_i[0])$ 
12:  Move to  $\text{taskBuffer}_i[0]$  following path and map along the way
13:   $x_i \leftarrow \text{taskBuffer}_i[0]$ 
14:  Remove  $\text{taskBuffer}_i[0]$ 
15:   $\mathcal{Q}_i \leftarrow \text{getFrontier}(\mathcal{M}, K)$ 
16: end while
```

from the following density function

$$\text{PWC}[a, b] := \begin{cases} \frac{1}{b-a} \mathbf{1}_{[a, b]}, & \text{if } 1 \notin (a, b); \\ \frac{1}{2(1-a)} \mathbf{1}_{[a, 1]} + \frac{1}{2(b-1)} \mathbf{1}_{[1, b]}, & \text{if } 1 \in (a, b); \end{cases}$$

where $\mathbf{1}_{\mathcal{S}}$ is the indicator function for the set $\mathcal{S}$. We use different ranges of (a, b) as simulation parameters to check mapping performance. For instance, if $1 < a < b$ (resp. $a < b < 1$), the all robots are uncertainty insensitive (resp. averse); and if $a < 1 < b$, we have mixed behavior. Moreover, if $a = b = 1$, then $\alpha_i = 1, \forall i \in \mathcal{A}$. This corresponds to all the agents using the regular Shannon's entropy for exploration. To that extent, in Figures 5.3 and 5.4, we color robot $i \in \mathcal{A}$ red if $\alpha_i > 1$ and blue if $\alpha_i < 1$. We use 10 robots randomly selected from $\text{PWC}[a, b]$ to perform simulations.

Algorithm 3. communicate

- 1: Input : $\mathcal{Q}_i, \mathcal{N}_i$; Output : $\{w_i^q\}_{q \in \mathcal{Q}}$
 - 2: $w_i^q \leftarrow 1, \forall q \in \mathcal{Q}$ ▷ initialize all weights to 1
 - 3: $z_i^q \leftarrow r(q)\phi(q), \forall q \in \mathcal{Q}$ ▷ utility using (5.13)
 - 4: Run (5.5) sufficiently long using $\mathcal{Q}_i, \mathcal{N}_i$ ▷ d-PBRAG
-

5.8.2 Exploration pipeline

The formal description of the exploration pipeline is given in Algorithm 2. Each robot $i \in \mathcal{A}$ runs Algorithm 2 separately and addresses the communication call among its neighbors $\mathcal{N}_i$ whenever needed. The neighbors of a robot at any time is the set of robots that have common frontiers. Each robot initializes an empty task buffer to keep track of the frontiers they visit.

First, robot $i \in \mathcal{A}$ updates a circular area $\mathcal{Q} = \mathbf{B}_r^{x_i}$ in $\mathcal{D}$, with a radius r and centered around its current position x_i . We use three different mapping radii $r \in \{2, 3, 4\}$ to capture different sensor ranges. Three levels of mapping noise $\sigma^m \in \{0, 1, 2\}$ are employed with the following meaning:

- a) In perfect mapping ($\sigma^m = 0$) each cell in $\mathcal{Q}$ is updated to the ground truth.
- b) In imperfect mapping ($\sigma^m = 1$) the occupancy values in $\mathcal{Q}$ are reduced (resp. increased) randomly with a random number generated from $\text{Unif}[0, 35]$ for each cell in $\mathcal{Q}$ that is a free-space (resp. obstacle).
- c) In highly imperfect mapping ($\sigma^m = 2$), occupancy values are updated with random numbers generated from $\text{Unif}[0, 15]$.

Second, frontiers $\mathcal{Q}_i$ are obtained as those cells in robot i 's map $\mathcal{M}_i$ with occupancy values < 2 and a non-negative gradient w.r.t. to the cells around them; see [89]. Each robot only considers frontiers in a certain radius around it. We set this frontier radius initially

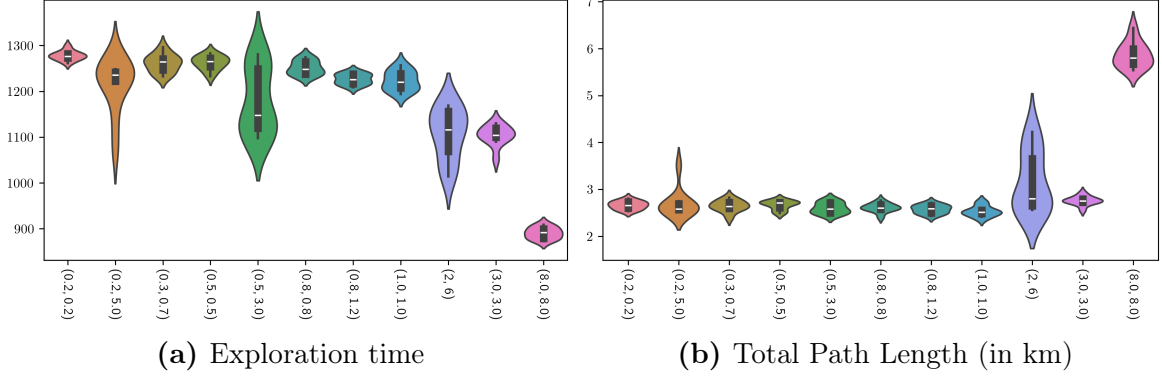


Figure 5.5. Simulation metric for $r = 2$, $\sigma^m = 2$. X-axis shows different α_i ranges (a, b) .

to 10 times the sensing radius; see Figure 5.4. The robot can increase the frontier radius if it is not able to see any frontier within this radius. This helps in reducing compute time and also communication cost.

Robots with empty task buffers, communicates with its neighbors to fill up their task buffer. For each $q \in \mathcal{F}_i$, we calculate the information gain using $\mathcal{Q} = \mathbf{B}_r^q$. That is, for each cell in $\mathcal{Q}$, the information gain per cell is given by the BE in these cells and the total information gain ν_i is the sum of these values. Then, for each $q \in \mathcal{Q}_i$, (5.13) gives its utility. Using this utility, the robots run rounds of the d-PBRAG algorithm, with step-size parameters of the form $\alpha_i^q(k) = k^{-1}$ and $\beta_i^q(k) = k$, sufficiently long, to allocate the available frontiers on their map among themselves. At the end of the communication rounds, robot i considers the frontiers $q \in \mathcal{Q}_i$ that have weights $w_i^q = 1$ and chooses at most N (we set $N = 14$) frontiers. The robot then finds a traveling salesman tour (TST) among these N frontiers.

Finally, the robot moves from x_i to the first frontier q_0 in the TST using a path computed using the RRT planner. The robot also maps (according to first step) all possible circular regions within its sensor range along the path. The robot then removes q_0 from the TST and continues the algorithm.

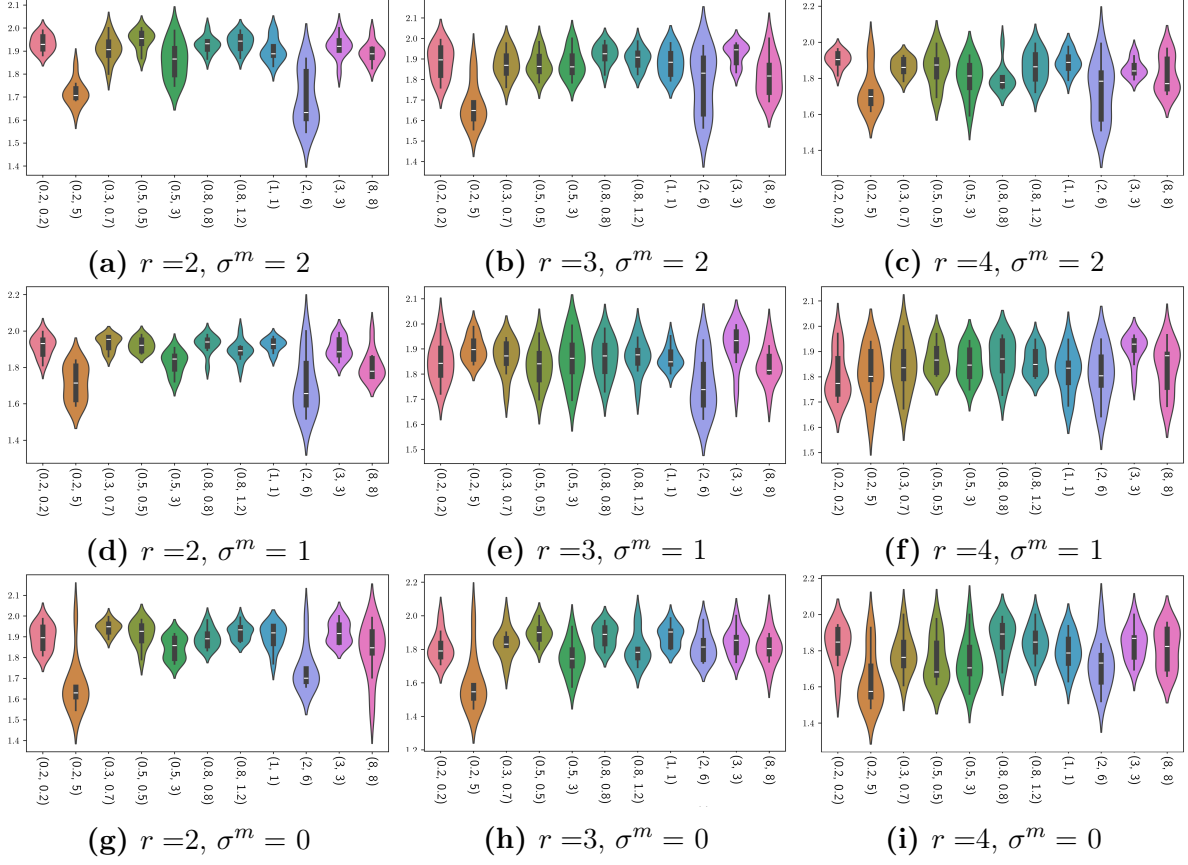


Figure 5.6. Cost for DARPA subT map. The rows from top to bottom have $\sigma^m = 2, 1, 0$ resp. The columns from left to right have $r = 2, 3, 4$ resp.

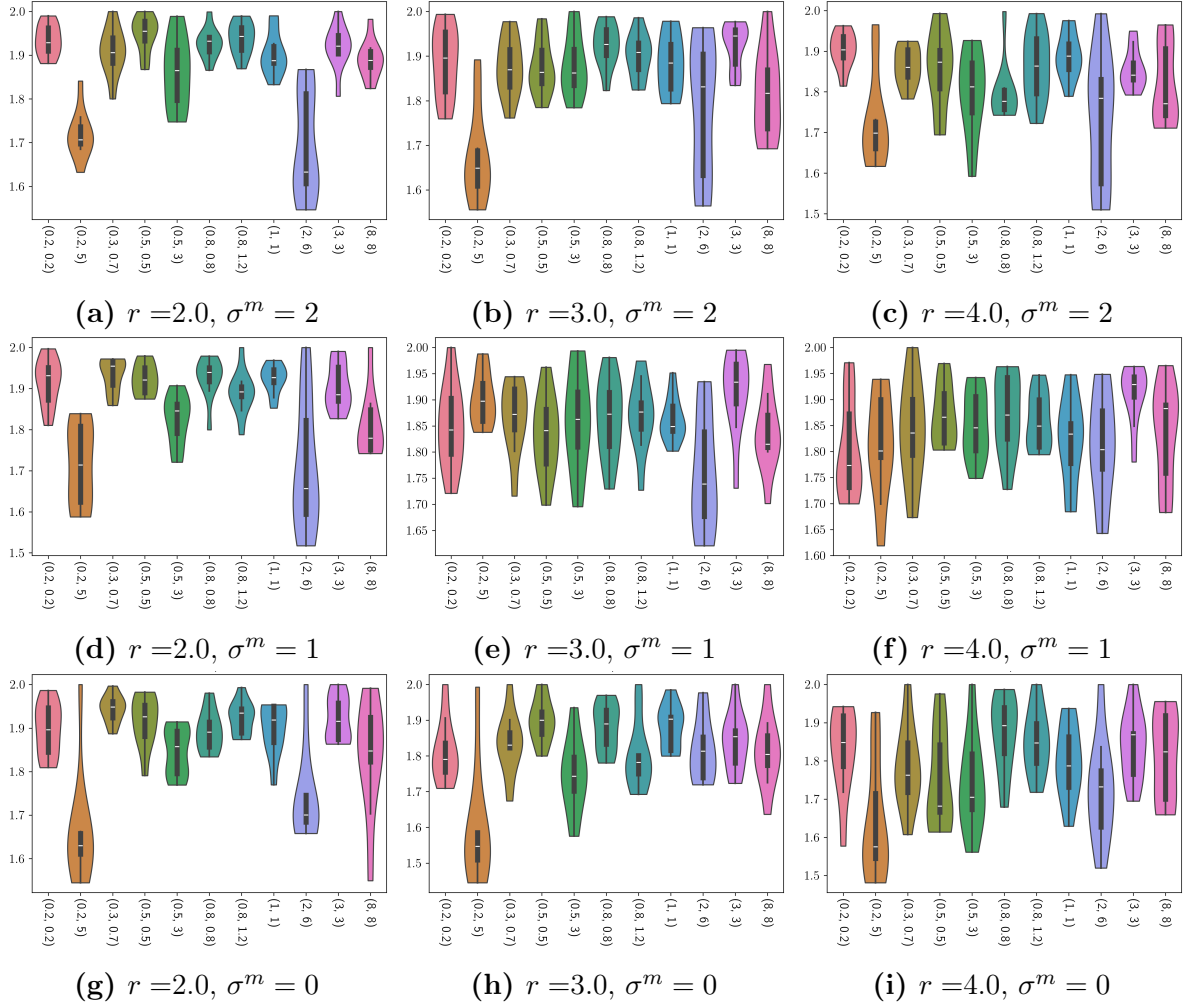


Figure 5.7. Cost for indoor office map. The rows from top to bottom have $\sigma^m = 2, 1, 0$ resp. The columns from left to right have $r = 2, 3, 4$ resp.

5.8.3 Results and Discussion

General trend in efficiency

We simulate 11 different α ranges to capture a wide range of behaviors. For this batch of simulations, we fix the number of robots to $n = 10$. For each α range, each sensing radius and noise levels we perform 10 simulations. We perform these experiments for both the indoor maps in Figure 5.3. This brings the total number of simulations to 1980. The results in Figure 5.5 show the distribution of the number of iterations taken to explore 99% of the DARPA SubT map and the distance travelled by the robots. The Y-axis in Figure 5.5(a) indicates the distribution of number of iterations used with a particular group to reduce at least 99% of initial entropy $\mathcal{H}(0)$. Similarly, the Y-axis in Figure 5.5(b) indicates the distribution of the total path length traveled by the robots. Wider (resp. narrower) violin denotes that more (resp. fewer) trials gave similar number of iterations.

First, we report the trade-off between time taken for entropy reduction and the total path length traveled by the robots for different α values in Figure 5.5. We choose the case $r = 2$ and $\sigma^m = 2$ to highlight this point. As seen in Figure 5.5(a), it is evident that increasing α values reduces the number of iterations required to reduce entropy. But, on the other hand Figure 5.5(b) shows that increasing the value of α also significantly increases the total path traveled by the robots (see (8,8) range plots for comparison). Figures 5.6 and 5.7 show violin plots for cost defined as the normalized sum of number of iterations required for entropy reduction and total path length on the DARPA SubT map and the indoor office map respectively.

From Figures 5.6 and 5.7, the general trend is that heterogeneous robots with some having $\alpha > 1$ tend to do better than other combinations. But these cases also give rise to higher variance in the results. It also important to note that this heterogeneity works best in smaller sensing radius and larger sensing noise case. When sensing radius is high and sensing noise is low, the different cases have comparable performance (compare (a) and (i)

Table 5.1. Mean cost compared with standard algorithms on DARPA subT map. The (a, b) range denotes the range of BE parameters, *i.e.* $\alpha_i \sim \text{PWC}[a, b]$, $\forall i \in \mathcal{A}$. Least cost for each row is given in bold and italics.

Radius	Noise	Nearest	Hungarian	(0.2,5)	(1,1)	(2,6)
2	2	1.83	1.77	1.70	1.88	<i>1.65</i>
2	1	1.80	1.73	1.69	1.90	<i>1.64</i>
2	0	1.69	1.67	<i>1.63</i>	1.91	1.71
3	2	1.75	1.73	<i>1.65</i>	1.89	1.81
3	1	1.83	<i>1.75</i>	1.89	1.85	<i>1.75</i>
3	0	1.79	1.68	<i>1.55</i>	1.86	1.81
4	2	1.71	<i>1.69</i>	<i>1.69</i>	1.89	1.79
4	1	1.64	<i>1.61</i>	1.80	1.85	1.80
4	0	1.64	1.59	<i>1.57</i>	1.78	1.71

plots of Figures 5.6 and 5.7). Thus in general, it is beneficial to have robots encoded with $\alpha > 1$ with robots possessing $\alpha < 1$ to balance exploration time and distance traveled.

Comparison with existing algorithms

We compared our algorithm with: i) a nearest-frontier allocation, and ii) a centralized Hungarian method for frontier allocation with Shannon’s entropy for frontier evaluation. For this case, we again fixed the number of agents to $n = 10$ and performed the experiments on all the maps shown in Figures 5.3 and 5.4. We chose $(0.2, 5)$, $(1, 1)$, and $(2, 6)$ as the ranges for α for these simulations. For each case, we performed 10 experiments with random initial conditions. The comparisons of our algorithm with standard baselines are reported in Tables 5.1, 5.2 and 5.3. The reported mean costs in these tables are the weighted sum of path traveled and the time taken to map the area. Notice that having a wide range of α performs better than using Shannon’s entropy.

Scaling with number of robots

Moreover, in order to evaluate the efficacy of our algorithm as the team size changes, we simulated our algorithm on the DARPA SubT map with different number of robots, specifically, $n \in [5, 10, 20, 40, 80]$. The mean cost values across 10 runs for each radius and noise parameters is shown in Table 5.4. As the number of agents increase in a

Table 5.2. Mean cost compared with standard algorithms on urban office environment. The (a, b) range denotes the range of BE parameters, *i.e.* $\alpha_i \sim \text{PWC}[a, b]$, $\forall i \in \mathcal{A}$. Least cost for each row is given in bold and italics.

Radius	Noise	Nearest	Hungarian	(0.2,5)	(1,1)	(2,6)
2	2	1.88	1.81	1.71	1.89	<i>1.63</i>
2	1	1.77	1.72	1.71	1.92	<i>1.65</i>
2	0	1.66	<i>1.61</i>	<i>1.61</i>	1.95	1.70
3	2	1.77	1.72	<i>1.65</i>	1.87	1.85
3	1	1.86	<i>1.75</i>	1.90	1.85	<i>1.75</i>
3	0	1.76	1.65	<i>1.55</i>	1.91	1.81
4	2	1.75	1.73	<i>1.70</i>	1.90	1.80
4	1	1.81	<i>1.80</i>	<i>1.80</i>	1.85	<i>1.80</i>
4	0	1.64	<i>1.59</i>	<i>1.59</i>	1.80	1.75

Table 5.3. Mean cost compared with standard algorithms on outdoor camp map. The (a, b) range denotes the range of BE parameters, *i.e.* $\alpha_i \sim \text{PWC}[a, b]$, $\forall i \in \mathcal{A}$. Least cost for each row is given in bold and italics.

Radius	Noise	Nearest	Hungarian	(0.2,5)	(1,1)	(2,6)
2	2	1.86	1.78	1.75	1.88	<i>1.60</i>
2	1	1.78	1.73	1.67	1.94	<i>1.62</i>
2	0	1.67	1.61	<i>1.60</i>	1.94	1.71
3	2	1.78	1.75	<i>1.70</i>	1.88	1.77
3	1	1.82	1.79	1.90	1.81	<i>1.70</i>
3	0	1.77	1.70	<i>1.51</i>	1.84	1.83
4	2	1.68	1.69	<i>1.66</i>	1.86	1.81
4	1	<i>1.73</i>	<i>1.73</i>	1.80	1.84	<i>1.73</i>
4	0	1.62	1.64	<i>1.59</i>	1.74	1.71

Table 5.4. Mean cost values for selected α ranges for DARPA SubT map with varying robot team size n . The (a, b) range denotes the range of BE parameters, *i.e.* $\alpha_i \sim \text{PWC}[a, b]$, $\forall i \in \mathcal{A}$. Least cost for each case is given in bold and italics.

Radius	Noise	$n = 5$			$n = 10$		
		(0.2,5)	(1,1)	(2,6)	(0.2,5)	(1,1)	(2,6)
2	2	1.75	1.92	<i>1.70</i>	1.70	1.88	<i>1.65</i>
3	1	1.93	1.87	<i>1.81</i>	1.89	1.85	<i>1.75</i>
4	0	<i>1.62</i>	1.81	1.79	<i>1.57</i>	1.78	1.71
Radius	Noise	$n = 20$			$n = 40$		
		(0.2,5)	(1,1)	(2,6)	(0.2,5)	(1,1)	(2,6)
2	2	1.60	1.85	<i>1.59</i>	<i>1.07</i>	1.31	1.09
3	1	1.80	1.81	<i>1.73</i>	1.27	1.28	<i>1.20</i>
4	0	1.49	1.68	<i>1.35</i>	<i>0.97</i>	1.16	1.30
Radius	Noise	$n = 80$					
		(0.2,5)	(1,1)	(2,6)			
2	2	0.84	0.79	<i>0.76</i>			
3	1	0.95	0.97	<i>0.88</i>			
4	0	<i>0.51</i>	0.65	0.79			

fixed environment, both the path length traveled by the robots and the time taken to complete the mapping reduces, reducing the total cost. Moreover, with $\alpha_i \sim \text{PWC}[0.2, 5]$, the exploration incurs a lower cost with lower mapping noise. Thus in presence of noisy mapping, a heterogeneous team with higher behavioral entropy parameters (here $\alpha_i \sim \text{PWC}[2, 6]$) is beneficial. This behavior is prominent with larger teams of robots where aggressive mapping schemes perform better.

Discussion on computation requirements

Our algorithm was simulated on a computer with 13th Gen Intel i7 chip and 32 GB of RAM. We compute the frontiers using the costmap’s numerical gradient. As such, noisy mapping retains more number of frontiers during exploration than more accurate (lower noise) mapping. Recall that a frontier’s utility is computed by calculating the BE for each grid cell inside a ball around the frontier. This poses a major bottleneck and contributes to the computation time. This issue remains true while comparing our

algorithm with a centralized Hungarian approach, as it involves computing Shannon’s entropy, similarly, for each grid cell in the ball around the frontier. However, when using the nearest frontier method, this computation time is omitted as this method only requires the distance to the nearest frontier. In terms of assigning frontiers among robots, d-PBRAG has a $O(m)$ communication cost, with m number of frontiers. Whereas centralized and distributed Hungarian methods have $O(m^3)$ and $O(m \log m)$ communication costs, respectively. Moreover, [95] showed that partially known rewards can produce oscillatory behavior in the distributed Hungarian algorithm. We recorded an average inter-iteration clock time (normalized by number of agents) of 93.26 ms for $\sigma^m = 0, r = 4$, and 274.39 ms for $\sigma^m = 2, r = 2$ cases (other σ^m, r combinations had times in between these extremes). In general, computation time increases when mapping noise or mapping radius increases.

5.9 Summary

We proposed a novel algorithm for multi-robot exploration based on a distributed frontier assignment scheme. We used heterogeneous BE to evaluate frontiers and used a distributed game-theoretic framework to assign these. We found that a team of robots with heterogeneous behaviors outperforms a team of homogeneous robots with respect to time and distance costs. This allowed us to heuristically prescribe BE parameters for robot teams to efficiently explore maps.

In the future, we would like to address constrained frontier assignment, efficiently allocate frontiers over longer time horizons, provide bounds on optimal BE parameter ranges for an environment, and perform experiments in challenging real-world settings.

This chapter is a reprint of material as it appears in “Behaviorally heterogeneous multi-agent exploration using distributed task allocation,” N. Mandal, A. Suresh, C. Nieto, and S. Martínez, *IEEE Robotics and Automation Letters*, pages 2983 – 2990, 2026. The dissertation author was the primary investigator and author of this paper.

Chapter 6

Robust Control Of Discrete-Time LTI Systems Using Stochastic Ensemble Systems

6.1 Introduction

Ensemble control, *i.e.*, investigating the ability of steering and manipulating an entire ensemble of (partially) unknown systems in a desired and optimal manner, has emerged from several science and engineering applications in recent years, *e.g.*, coordination of the movement of flocks in biology [99], manipulation of spin ensembles in nuclear magnetic resonance [100, 101], or desynchronization of pathological neurons in the brain in neuroscience [102].

6.2 Literature Review

Motivated by this, a huge body of seminal work has been done on the analysis of (deterministic) ensemble controllability [103, 104, 105] and synthesis of optimal ensemble controls [106, 107, 108] by developing new analytical and numerical methods, which have received increasing attention due to their application to robotics [105], energy systems [109, 110] and quantum control [111, 112]. However, the traditional ensemble control problem consists of driving a collection of initial states of a continuum of systems

to a set of final states with the same control input [113], which could be potentially conservative. The problem of reachability for finite-dimensional linear time-varying systems has been studied in [114, 115], but here the system varies in a continuum and not in a countable set. On the other hand, the reachability properties of bilinear ensemble systems are studied in [116]. In [106], the authors explore the problem of optimal control for stochastic linear systems. The stochastic nature comes from additive noise to the dynamics. In [117, 118] the authors study robustness of matrix properties to unknown system parameters. Reference [109] solves an ensemble control problem for devices with cyclic energy consumption patterns by utilizing techniques from the Markov Decision Process framework. Finally, the work [110] looks into a similar problem but with uncertain dynamics and uses techniques from stochastic and distributionally robust optimization in the process. However, in all the aforementioned work, either the control signal needs to be unique, or a continuum of ensemble (and partially unknown) systems is required to exist. This work aims to bridge this gap.

6.3 Contributions

Unlike ensemble control, we let the system parameters take values in a countable set. Moreover, we do not restrict the control to be unique but allow every sample run of the system to employ a different control function. This being said, we consider a so-called class of stochastic ensemble systems, which arise from the approximation of systems whose exact parameters are unknown. A known approach to obtain good parametric models consists of learning the distribution of for the model whose realizations correspond to a system approximation. As more data becomes available, the distributions become more accurate, resulting in a stochastic ensemble approximation. This enables us to i) study the reachability properties of a new class of stochastic ensemble systems, ii) provide a procedure for combining controls from such systems, iii) compute a control for the limiting

DLTI system, iv) characterize the variance of the state and control of a stochastic ensemble system produced by means of a Bernoulli distribution, and v) derive an improved result by means of least squares error minimization. As conclusions of these main contributions, we also can vi) compute the variance of the state and control of a stochastic ensemble system that is obtained in an streaming fashion. We illustrate our results in numerical examples.

Notations

We denote the set of real numbers, the set of non-negative real numbers, the set of integers, and the set of non-negative integers using $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, $\mathbb{Z}$, and $\mathbb{Z}_{\geq 0}$, respectively. We let $\mathbb{R}^n$ (similarly $\mathbb{Z}_{\geq 0}^n$) be the Cartesian product of $\mathbb{R}$ (similarly $\mathbb{Z}_{\geq 0}$) with itself. $\mathbb{R}^{n \times m}$ denotes the set of real matrices of order $n \times m$. The i^{th} component of a vector $\mathbf{v} \in \mathbb{R}^n$ is denoted by $\mathbf{v}_i$ and the ij^{th} entry of a matrix $\mathbf{M} \in \mathbb{R}^{n \times m}$ is denoted by $\mathbf{M}_{ij}$. For a set of square matrices $\{\mathbf{M}^{(s)}\}_{s \in \mathbb{Z}}$, the product operator is used to denote multiplication from right to left, *i.e.* $\prod_{i=1}^n \mathbf{M}^{(s)} := \mathbf{M}^{(n)} \cdots \mathbf{M}^{(1)}$. For a vector $\mathbf{v} \in \mathbb{R}^n$, $\mathbf{v} \geq 0$ denotes a component-wise inequality. We let $\text{abs} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be a function that produces a vector with component-wise absolute values from the input vector, *i.e.* $\text{abs}(\mathbf{v})_i = |\mathbf{v}_i| \in \mathbb{R}$, $\forall i \in \{1, \dots, n\}$. We denote the empty set using $\emptyset$.

6.4 Problem Formulation

Consider a discrete-time, unknown, linear time-invariant (LTI) system of the form

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k), \quad (6.1)$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state, $\mathbf{x}(0) = \mathbf{x}_0$ and $\mathbf{x}_f$ are the initial and desired final states, respectively, $\mathbf{u} \in \mathbb{R}^m$ is the control input, $\mathbf{A} \in \mathbb{R}^{n \times n}$ and $\mathbf{B} \in \mathbb{R}^{n \times m}$. The system matrices $\mathbf{A}$ and $\mathbf{B}$, as well as the initial state $\mathbf{x}_0$ are unknown, but it is possible to construct increasingly good approximations of these system parameters using some known approach, *e.g.*, any data-driven or Machine Learning-based method. This approximation is assumed

to be done through the following newly defined class of *stochastic ensemble systems*, for which we aim to characterize reachability properties.

Definition 6.4.1. (*Stochastic ensemble system*). Consider sequences of independent random variables $\mathcal{X}_0 := \{\mathbf{x}_0^{(s)} \in \mathbb{R}^n\}_{s \in \mathbb{Z}_{\geq 0}}$, $\mathcal{A} := \{\mathbf{A}^{(s)} \in \mathbb{R}^{n \times n}\}_{s \in \mathbb{Z}_{\geq 0}}$ and $\mathcal{B} := \{\mathbf{B}^{(s)} \in \mathbb{R}^{n \times m}\}_{s \in \mathbb{Z}_{\geq 0}}$ such that $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{X}_0$ are all independent of each other. For every $\sigma \in \mathbb{Z}_{\geq 0}$, a stochastic ensemble system is a linear, time varying (LTV) system of the form:

$$\widehat{\mathbf{x}}(\sigma, k+1) = \widehat{\mathbf{A}}(s_A(\sigma, k)) \widehat{\mathbf{x}}(\sigma, k) + \widehat{\mathbf{B}}(s_B(\sigma, k)) \widehat{\mathbf{u}}(\sigma, k) \quad (6.2a)$$

$$\widehat{\mathbf{x}}(\sigma, 0) = \mathbf{x}_0^{(s_0(\sigma))} \in \mathcal{X}_0, \quad (6.2b)$$

where, the indicator functions $s_A : \mathbb{Z}_{\geq 0} \times \mathbb{Z}_{\geq 0} \rightarrow \mathbb{Z}_{\geq 0}$, $s_B : \mathbb{Z}_{\geq 0} \times \mathbb{Z}_{\geq 0} \rightarrow \mathbb{Z}_{\geq 0}$ and $s_0 : \mathbb{Z}_{\geq 0} \rightarrow \mathbb{Z}_{\geq 0}$ are used to index different realizations of the random variables in $\mathcal{X}_0$, $\mathcal{A}$, and $\mathcal{B}$, respectively. Thus, $\widehat{\mathbf{A}}(s_A(\sigma, k))$ (similarly $\widehat{\mathbf{B}}(s_B(\sigma, k))$) represents a realization of $\mathbf{A}^{(s_A(\sigma, k))} \in \mathcal{A}$ (respectively $\mathbf{B}^{(s_B(\sigma, k))} \in \mathcal{B}$) for each σ and k . •

We use the terms random variable and realization interchangeably. The σ parameter is used to distinguish between different runs of the system. Also note that since the initial state and the system matrices in (6.2) are realizations of different random variables, each state $\widehat{\mathbf{x}}(\sigma, k)$ is also a realization of some random variable. Moreover, we need to formally define the notion of *stochastic ensemble approximation*.

Definition 6.4.2. (*Stochastic ensemble approximation*). Let $\mathbf{M} \in \mathbb{R}^{n \times m}$. Consider a sequence of independent random variables $\mathcal{M} := \{\mathbf{M}^{(s)}\}_{s \in \mathbb{Z}_{\geq 0}}$ such that $\exists B \in \mathbb{R}$ such that $\|\mathbf{M}^{(s)}\| \leq B, \forall s \in \mathbb{Z}_{\geq 0}$. Then, $\mathcal{M}$ is a stochastic ensemble approximation of $\mathbf{M}$ if and only if

$$\forall \varepsilon > 0, \Pr(\|\mathbf{M} - \sum_{s=1}^d w_s \mathbf{M}^{(s)}\|_p > \varepsilon) \rightarrow 0 \text{ as } d \rightarrow \infty,$$

with respect to some p -norm $\|\cdot\|_p$ and for some w_s 's of the form $[w_1, \dots, w_d]^\top \in \mathcal{S}^d := \{\mathbf{y} \in \mathbb{R}^d \mid \mathbf{1}^\top \mathbf{y} = 1, \mathbf{y} \geq 0\}$ for each $d \in \mathbb{Z}_{\geq 0}$. •

Furthermore, we assume the following.

Assumption 6.4.3. *There exists a stochastic ensemble system of the form (6.2) that consists of stochastic ensemble approximations of the original system given in (6.1).* •

Being concerned with meeting certain reachability requirements for (6.2), next we provide a formal definition for the notion of approximate reachability.

Definition 6.4.4. (*Approximate reachability of final state from an initial state*). Let $\mathcal{X}_0$, $\mathcal{A}$, and $\mathcal{B}$ be stochastic ensemble approximations of $\mathbf{x}_0$, $\mathbf{A}$, and $\mathbf{B}$ respectively. Consider the stochastic ensemble system in (6.2) and let $\mathbf{x}_f \in \mathbb{R}^n$. Suppose there exists a fixed time $K \in \mathbb{Z}_{\geq 0}$ and for any realization of the random variables $\widehat{\mathbf{x}}(\sigma, 0)$, $\widehat{\mathbf{A}}(s_A(\sigma, k))$, and $\widehat{\mathbf{B}}(s_B(\sigma, k))$ there exist controls $\widehat{\mathbf{u}}(\sigma, 0), \dots, \widehat{\mathbf{u}}(\sigma, K-1)$ that drive the realization $\widehat{\mathbf{x}}(\sigma, 0)$ through the trajectory $\widehat{\mathbf{x}}(\sigma, k)$, $\forall k \in \{1, \dots, K\}$ under the dynamics (6.2), *i.e.*

$$\begin{aligned} \widehat{\mathbf{x}}(\sigma, K) &= \prod_{k=0}^K \widehat{\mathbf{A}}(s_A(\sigma, k)) \widehat{\mathbf{x}}(\sigma, 0) \\ &\quad + \sum_{k=0}^{K-1} [\prod_{i=k+1}^K \widehat{\mathbf{A}}(s_A(\sigma, i))] \widehat{\mathbf{B}}(s_B(\sigma, k)) \widehat{\mathbf{u}}(\sigma, k). \end{aligned} \quad (6.3)$$

Then $\mathbf{x}_f$ is said to be approximately reachable from $\mathbf{x}_0$ if and only if $\{\widehat{\mathbf{x}}(\sigma, K)\}_{\sigma \in \mathbb{Z}_{\geq 0}}$ is a stochastic ensemble approximation of $\mathbf{x}_f$. •

Further, we need to formally introduce the notion of sample reachability as follows.

Definition 6.4.5. (*Sample reachability of final state from an initial state in unit time*).

Let $\mathcal{X}_0$, $\mathcal{X}_f$, and $\mathcal{A}$ be stochastic ensemble approximations of $\mathbf{x}_0$, $\mathbf{x}_f$ and $\mathbf{A}$ respectively, and $s_f : \mathbb{Z}_{\geq 0} \rightarrow \mathbb{Z}_{\geq 0}$ be an indicator function that denotes the realizations of the random variables in the set $\mathcal{X}_f := \{\mathbf{x}_f^{(s)} \in \mathbb{R}^n\}_{s \in \mathbb{Z}_{\geq 0}}$. Consider the stochastic ensemble system in (6.2) with $\widehat{\mathbf{B}}(k) = \mathbf{B}$. Suppose there exists a control $\{\widehat{\mathbf{u}}(\sigma, 0)\}_{\sigma \in \mathbb{Z}_{\geq 0}}$ for each $\sigma \in \mathbb{Z}_{\geq 0}$ that drives $\widehat{\mathbf{x}}(\sigma, 0) = \mathbf{x}_0^{(s_0(\sigma))}$ to $\widehat{\mathbf{x}}(\sigma, K) = \mathbf{x}_f^{(s_f(\sigma))}$ under the dynamics (6.2), in one ($K = 1$) time step, *i.e.*

$$\mathbf{x}_f^{(s_f(\sigma))} = \widehat{\mathbf{A}}(s_A(\sigma, 1)) \mathbf{x}_0^{(s_0(\sigma))} + \mathbf{B} \widehat{\mathbf{u}}(\sigma, 0). \quad (6.4)$$

Then $\mathbf{x}_f$ is sample reachable from $\mathbf{x}_0$ in unit time. •

Note that the control $\widehat{\mathbf{u}}(\sigma, 0)$ is chosen in order to drive $\mathbf{x}_0^{(s_0(\sigma))}$ to $\mathbf{x}_f^{(s_f(\sigma))}$ in the previous definition.

Our problem of interest can be cast as follows.

Problem 6.4.6. Given the setup and Assumption 6.4.3, characterize the relation between approximate reachability under the stochastic ensemble system (6.2) and the standard reachability under the DLTI system (6.1). Moreover, find controls that drive $\mathbf{x}_0$ to $\mathbf{x}_f$ under (6.1) using the stochastic ensemble system (6.2). •

We conclude this section by highlighting the difference between our notion of stochastic ensemble system with that of (deterministic) ensemble control.

Remark 6.4.7. (*Comparison with ensemble control*). The stochastic ensemble system in (6.2) has a similar structure to ensemble systems [103]. However, the σ that parameterizes the system takes discrete values. Moreover, $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{B}}$ do not vary continuously with respect to σ . We also allow the controls for each σ to be different, *i.e.*, we do not seek one single control that steers the ensemble system between points of interest. Finally, if a system

is ensemble controllable, then similar results to the ones in this chapter can be given, generalizing ensemble control systems and their objective. •

6.5 Control using Stochastic Ensemble Systems

In this section, we first show that reachability under the DLTI system (6.1) is sufficient for approximate reachability under the stochastic ensemble systems.

Recall that we use $\sigma \in \mathbb{Z}_{\geq 0}$ to index the runs of the system. Moreover, $s_0(\sigma)$ (respectively $s_A(\sigma, k)$, $s_B(\sigma, k)$) indexes the realization of the initial state (respectively of matrices $\mathbf{A}$ and $\mathbf{B}$ at the k^{th} time step) used in (6.2). In the sequel, we omit the σ for the sake of brevity wherever there is no confusion. Now we are ready to state the result.

Lemma 6.5.1. (Sufficient condition for approximate reachability using state reachability). *Let $\mathcal{X}_0$, $\mathcal{A}$, and $\mathcal{B}$ be stochastic ensemble approximations of $\mathbf{x}_0$, $\mathbf{A}$, and $\mathbf{B}$ respectively. Suppose that $\mathbf{x}_f$ is reachable from $\mathbf{x}_0$ in $K \in \mathbb{Z}_{\geq 0}$ time steps under the dynamics (6.1). Then, $\mathbf{x}_f$ is approximately reachable from $\mathbf{x}_0$ under the stochastic ensemble system in (6.2).*

Proof. Since $\mathcal{X}_0$, $\mathcal{A}$, and $\mathcal{B}$ are stochastic ensemble approximations of $\mathbf{x}_0$, $\mathbf{A}$, and $\mathbf{B}$ respectively $\exists B_{\mathcal{X}_0}$, $B_{\mathcal{A}}$, and $B_{\mathcal{B}}$ such that $\|\mathbf{x}_0^{(s)}\| \leq B_{\mathcal{X}_0}$, $\|A^{(s)}\| \leq B_{\mathcal{A}}$, and $\|B^{(s)}\| \leq B_{\mathcal{B}}$, $\forall s \in \mathbb{Z}_{\geq 0}$. Now, by the hypothesis, there exists controls $\mathbf{u}(0), \dots, \mathbf{u}(K-1)$ that drive $\mathbf{x}_0$ to $\mathbf{x}_f$ under the dynamics (6.1). Moreover since K is finite, all the controls can be bounded by a constant $B_{\mathbf{u}}$. We show that $\hat{\mathbf{u}}(\sigma, k) = \mathbf{u}(k)$, $\forall k \in \{1, \dots, K-1\}$ proves the claim. Let $\varepsilon > 0$. If we use the controls $\mathbf{u}(0), \dots, \mathbf{u}(K-1)$ in (6.3), we obtain

$$\begin{aligned} \mathbf{y}(1) - \sum_{s_0=1}^N w_{s_0,1} \hat{\mathbf{x}}(s(1)) &= \hat{\mathbf{A}}(s_A(1))(\mathbf{x}_0 - \sum_{s_0=1}^N w_{s_0,1} \mathbf{x}_0^{(s_0)}) \\ &\quad + \hat{\mathbf{B}}(s_B(1))(\mathbf{u}(0) - \sum_{s_0=1}^N w_{s_0,1} \mathbf{u}(0)), \end{aligned}$$

at $k = 1$ starting at $\mathbf{x}_0^{(s_0)}$ and for $[w_{s_0,1}] = [w_{1,1}, \dots, w_{N,1}] \in \mathcal{S}^N$ which makes $\mathcal{X}_0$ a stochastic ensemble approximation of $\mathbf{x}_0$ and $\mathbf{y}(1) = \hat{\mathbf{A}}(s_A(\sigma, 1))\mathbf{x}_0 + \hat{\mathbf{B}}(s_B(\sigma))\mathbf{u}(0)$. Note that here we are using the weights that make $\mathcal{X}_0$ a stochastic ensemble approximation

throughout and we use the independence properties of the random variables to separate the product and the sum. Now, since $\sum_{s_0=1}^N w_{s_0,1} \mathbf{u}(0) = \mathbf{u}(0)$, then

$$\|\mathbf{y}(1) - \sum_{s_0=1}^N w_{s_0,1} \widehat{\mathbf{x}}(s(1))\| \leq B_{\mathcal{A}} \|\mathbf{x}_0 - \sum_{s_0=1}^N w_{s_0,1} \mathbf{x}_0^{(s_0)}\|. \quad (6.5)$$

Next, by defining $\mathbf{z}(1) = \widehat{\mathbf{A}}(s_A(1))\mathbf{x}_0 + \mathbf{B}\mathbf{u}(0)$ and considering a $[w_{s_B(1),2}] = [w_{1,2}, \dots, w_{N,2}] \in \mathcal{S}^N$, $\mathcal{B}$ becomes a stochastic ensemble approximation of $\mathbf{B}$. Then, by similar arguments as before

$$\begin{aligned} \|\mathbf{z}(1) - \sum_{s_B(1)=1}^N w_{s_B(1),2} \mathbf{y}(1)\| &\leq \\ B_{\mathbf{u}} \|\mathbf{B} - \sum_{s_B(1)=1}^N w_{s_B(1),2} \mathbf{B}(s_B(1))\|. \end{aligned} \quad (6.6)$$

Finally, for $\mathbf{x}(1) = \mathbf{A}\mathbf{x}_0 + \mathbf{B}\mathbf{u}(0)$, we have

$$\begin{aligned} \|\mathbf{x}(1) - \sum_{s_A(1)=1}^N w_{s_A(1),3} \mathbf{z}(1)\| & \\ \leq \|\mathbf{A} - \sum_{s_A(1)=1}^N w_{s_A(1),3} \widehat{\mathbf{A}}(s_A(1))\| \|\mathbf{x}(0)\|, \end{aligned} \quad (6.7)$$

using $[w_{s_A(1),2}] = [w_{1,2}, \dots, w_{N,2}] \in \mathcal{S}^N$ which makes $\mathcal{A}$ a stochastic ensemble approximation of $\mathbf{A}$. Then, using the triangle inequality of the norms and (6.5)–(6.7), we obtain

$$\begin{aligned} \|\mathbf{x}(1) - \sum_{s_0=1}^N w_{s_0,1} \widehat{\mathbf{x}}(\sigma, 1)\| &\leq \\ \alpha [\|\mathbf{x}_0 - \sum_{s_0=1}^N w_{s_0,1} \mathbf{x}_0^{(s_0)}\| + \|\mathbf{B} - \sum_{s_B(1)=1}^N w_{s_B(1),2} \mathbf{B}(s_B(1))\| & \\ + \|\mathbf{A} - \sum_{s_A(1)=1}^N w_{s_A(1),3} \widehat{\mathbf{A}}(s_A(1))\|], \end{aligned}$$

where $\alpha = \max\{B_{\mathcal{A}}, B_{\mathbf{u}}, \|\mathbf{x}(0)\|\}$. Hence,

$$\begin{aligned} & \Pr(\|\mathbf{x}(1) - \sum_{s_0=1}^N w_{s_0,1} \widehat{\mathbf{x}}(s_0, 1)\| < \varepsilon) \geq \\ & \Pr(\|\mathbf{A} - \sum_{s_A(1)=1}^N w_{s_A(1),3} \widehat{\mathbf{A}}(s_A(1))\| < \frac{\varepsilon}{3\alpha}) + \\ & \Pr(\|\mathbf{B} - \sum_{s_B(1)=1}^N w_{s_B(1),2} \widehat{\mathbf{B}}(s_B(1))\| < \frac{\varepsilon}{3\alpha}) + \\ & \Pr(\|\mathbf{x}_0 - \sum_{s_0=1}^N w_{s_0,1} \mathbf{x}_0^{(s_0)}\| < \frac{\varepsilon}{3\alpha}) \rightarrow 1 \text{ as } N \rightarrow \infty. \end{aligned}$$

Thus, $\{\widehat{\mathbf{x}}(s(1))\}_{s(1) \in \mathbb{Z}_{\geq 0}}$ forms a stochastic ensemble approximation of $\mathbf{x}(1)$. Thus, using the fact that $\{\mathbf{x}_0^{(s_0)}\}_{s_0 \in \mathbb{Z}_{\geq 0}}$, $\{\widehat{\mathbf{A}}(s_A(1))\}_{s_A(1) \in \mathbb{Z}_{\geq 0}}$, and $\{\widehat{\mathbf{B}}(s_B(1))\}_{s_B(1) \in \mathbb{Z}_{\geq 0}}$ are stochastic ensemble approximations of $\mathbf{x}_0$, $\mathbf{A}$, and $\mathbf{B}$ respectively, we have shown that $\{\widehat{\mathbf{x}}(s(1))\}_{s(1) \in \mathbb{Z}_{\geq 0}}$ forms a stochastic ensemble approximation of $\mathbf{x}(1)$. Next using the fact that $\{\widehat{\mathbf{x}}(s(1))\}_{s(1) \in \mathbb{Z}_{\geq 0}}$, $\{\widehat{\mathbf{A}}(s_A(2))\}_{s_A(2) \in \mathbb{Z}_{\geq 0}}$, and $\{\widehat{\mathbf{B}}(s_B(2))\}_{s_B(2) \in \mathbb{Z}_{\geq 0}}$ are stochastic ensemble approximations of $\mathbf{x}(1)$, $\mathbf{A}$, and $\mathbf{B}$ respectively we can show that $\{\widehat{\mathbf{x}}(s(2))\}_{s(2) \in \mathbb{Z}_{\geq 0}}$ forms a stochastic ensemble approximation of $\mathbf{x}(2)$. By induction on $\mathbf{x}(1), \dots, \mathbf{x}(K)$, the proof can be completed. ■

Note that the previous lemma states that the desired state $\mathbf{x}_f$ is reachable from $\mathbf{x}_0$ only if $\mathbf{x}_f$ is approximately reachable from $\mathbf{x}_0$. Also, note that we deal with a much weaker version of reachability here. All we require is that $\mathbf{x}_f$ be in the reachable subspace from $\mathbf{x}_0$. We do not require the whole $\mathbb{R}^n$ to be reachable.

Next, by using the notion of sample reachability (Definition 6.4.5), we synthesize a control input sequence that drives the system in (6.1) from $\mathbf{x}_0$ to $\mathbf{x}_f$.

Lemma 6.5.2. (Approximation of control of LTI systems using stochastic ensemble systems). *Consider the system in (6.1) with initial state $\mathbf{x}_0$ and desired final state $\mathbf{x}_f$. Let $\mathcal{X}_0$, $\mathcal{X}_f$ and $\mathcal{A}$ be stochastic ensemble approximations of $\mathbf{x}_0$, $\mathbf{x}_f$ and $\mathbf{A}$ respectively. Next consider the stochastic ensemble system in (6.2) with $\widehat{\mathbf{B}}(k) = \mathbf{B}$, $\forall k \in \{1, \dots, K-1\}$.*

Suppose that $\mathbf{x}_f$ is sample reachable from $\mathbf{x}_0$ in unit time. Let $\{\widehat{\mathbf{u}}(\sigma, 0)\}_{\sigma \in \mathbb{Z}_{\geq 0}}$ be as in Definition 6.4.5. Then for each $\varepsilon > 0$,

$$\lim_{N \rightarrow \infty} \Pr(\|\mathbf{B}\mathbf{v} - \mathbf{B} \sum_{\sigma=1}^N w_{\sigma} \widehat{\mathbf{u}}(\sigma, 0)\| > \varepsilon) \rightarrow 0 \quad (6.8)$$

with w_{σ} 's of the form $[w_1, \dots, w_N]^{\top} \in \mathcal{S}^N$ for each $N \in \mathbb{Z}_{\geq 0}$ and with $\mathbf{B}\mathbf{v}$ satisfying

$$\mathbf{x}_f = \mathbf{A}\mathbf{x}_0 + \mathbf{B}\mathbf{v}. \quad (6.9)$$

Proof. We do this in a very similar way to the proof of Lemma 6.5.1. So we skip most of the details due to lack of space. Consider an $\varepsilon > 0$. Then, consider the state evolution equations

$$\mathbf{x}_f^{(s_f)} = \widehat{\mathbf{A}}(s_A(1))\mathbf{x}_0 + \mathbf{B}\mathbf{u}'(0), \quad (6.10a)$$

$$\mathbf{x}_f = \widehat{\mathbf{A}}(s_A(1))\mathbf{x}_0 + \mathbf{B}\mathbf{u}''(0). \quad (6.10b)$$

Note that (6.10a) has been formulated by choosing controls such that they drive the initial condition $\mathbf{x}_0$ (and not the realizations) to the final state $\mathbf{x}_f^{(s_f)}$. Similarly (6.10b) has been formulated by choosing controls such that they drive the initial condition $\mathbf{x}_0$ to the final state $\mathbf{x}_f$ (and not the realizations). Then, using very similar arguments as in the proof of Lemma 6.5.1 it is possible to show that the controls $\mathbf{u}'$, and $\mathbf{u}''$ exist and can be attained by taking the weighted sum of the controls in (6.4).

Moreover, using the triangle inequality of the norms and probability theory, it can be shown that

$$\begin{aligned}
& \Pr(\|\mathbf{B}\mathbf{v} - \mathbf{B} \sum_{s_0=1}^N w_{s_0,1} \hat{\mathbf{u}}(s_0)\| < \varepsilon) \geq \\
& \Pr(\|\mathbf{A} - \sum_{s_A(1)=1}^N w_{s_A(1),3} \hat{\mathbf{A}}(s_A(1))\| < \frac{\varepsilon}{3\beta}) + \\
& \Pr(\|\mathbf{x}_f - \sum_{s_f=1}^N w_{s_f,2} \mathbf{x}_f^{(s_f)}\| < \frac{\varepsilon}{3\beta}) + \\
& \Pr(\|\mathbf{x}_0 - \sum_{s_0=1}^N w_{s_0,1} \mathbf{x}_0^{(s_0)}\| < \frac{\varepsilon}{3\beta}) \rightarrow 1 \text{ as } N \rightarrow \infty,
\end{aligned}$$

where $\beta = \max\{B_{\mathcal{A}}, 1, \|\mathbf{x}_0\|\}$. ■

The previous result states that if $\mathbf{x}_f$ is sample reachable from $\mathbf{x}_0$ in unit time, $\mathbf{x}_f$ is also reachable under (6.1). It is worthwhile to note that we are not approximating $\mathbf{B}$ using a stochastic ensemble approximation. If we did, then essentially we would introduce convolution-like sums which makes the problem more complicated. Further, if $\mathbf{x}_f$ is reachable from $\mathbf{x}_0$ only in multiple time steps, then the matter of non-unique control sequences poses another problem.

Note that the claim in Lemma 6.5.2 works for any sequence of random variables that form stochastic ensemble approximations as defined in Definition 6.4.2. In the next section we study stochastic ensemble systems that are generated from sampling the DLTI system parameters in (6.1).

6.6 On Stochastic Ensemble Systems Generated through Sampling

Here, we deal with a particular kind of stochastic ensemble system whose realizations are produced from samples of (6.1). This corresponds to the following scenario. Consider the individual components of the states to be nodes and the $\mathbf{A}$ matrix describing the

interconnection between them. Suppose each node is aware of which neighbors are antagonistic and which neighbors are cooperative, but is unaware of the absolute magnitude of influence of its neighbors. The realizations of $\hat{\mathbf{A}}$ are hence obtained based on the relative order of influence of other nodes on a particular node.

In such a scenario, we assume that the stochastic ensemble system is produced in the following way. Suppose the vectors and matrices are transformed into probability mass functions over an underlying sample space. This mass function produces samples from the sample space which correspond to the realizations of the stochastic ensemble system. In particular, we first define a function $\gamma : \mathbb{R}^n \rightarrow \mathbb{R}$, as $\gamma(\mathbf{w}) := \mathbf{1}^\top \text{abs}(\mathbf{w})$, and with a slight abuse of notation, we consider $\gamma : \mathbb{R}^{n \times m} \rightarrow \mathbb{R}$ as $\gamma(\mathbf{M}) := \max_{j \in \{1, \dots, m\}} \mathbf{1}^\top \text{abs}(\mathbf{M}_{*j})$. This helps us in describing a *vector sampling scheme* procedure through the following lemma, whose proof omitted since it is trivial.

Lemma 6.6.1. (Vector sampling scheme). *Consider a vector $\mathbf{w} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$. Then, let $\bar{\mathbf{w}} := (1/\gamma(\mathbf{w}))\text{abs}(\mathbf{w})$. Consider $\bar{\mathbf{w}}$ as a mass function on the sample space $\Omega = \{1, \dots, n\}$ and let $\{s_1, \dots, s_N\}$ be N samples of Ω . To each sample s_i , associate a vector $\mathbf{s}^{(i)}$ as*

$$[\mathbf{s}^{(i)}]_j := \begin{cases} \gamma(\mathbf{w}), & \text{if } j = s_i \text{ and } [\mathbf{w}]_j > 0; \\ -\gamma(\mathbf{w}), & \text{if } j = s_i \text{ and } [\mathbf{w}]_j < 0; \\ 0, & \text{otherwise.} \end{cases}$$

Then, the mean of the random variable corresponding to the i^{th} component of $\sum_{i=1}^N \mathbf{s}^{(i)}$ is $\mathbf{w}_i$. Finally, $\lim_{N \rightarrow \infty} \Pr(\|\mathbf{w} - \frac{1}{N} \sum_{i=1}^N \mathbf{s}^{(i)}\| > \varepsilon) \rightarrow 0$, for each $\varepsilon > 0$.

■

Note that for matrices, the same procedure can be performed via a similar transformation by considering each row (or column) separately as vectors.

The stochastic ensemble system generated in this process has nice properties related

to the system (6.1). First, it retains the sparsity structure of the DLTI system (6.1). Moreover, since each component is a Bernoulli random variable, using known properties we can provide convergence rates for each realization. Next, we analyze the variance of the average control computed considering stochastic ensemble approximations produced as in Lemma 6.6.1. The next result follows from Hoeffding's inequality.

Lemma 6.6.2. (Variance of average control). *Let $\mathcal{X}_0$, $\mathcal{X}_f$ and $\mathcal{A}$ be stochastic ensemble approximations of $\mathbf{x}_0$, $\mathbf{x}_f$ and $\mathbf{A}$ respectively generated using Lemma 6.6.1. Suppose $\{\hat{\mathbf{u}}(\sigma, 0)\}_{\sigma \in \mathbb{Z}_{\geq 0}}$ satisfy the hypothesis of Lemma 6.5.2 and let $\mathbf{v}$ be as in (6.9). Then for each $\varepsilon > 0$,*

$$\begin{aligned} \Pr(\|\mathbf{B}(\mathbf{v} - \frac{1}{N} \sum_{\sigma=1}^N \mathbf{u}^{(s)})\| > \varepsilon) &\leq 2C[\exp(-\frac{2N\varepsilon^2}{9\beta^2\gamma(\mathbf{x}_f)^2}) \\ &+ \exp(-\frac{2N\varepsilon^2}{9\beta^2\gamma(\mathbf{x}_0)^2}) + \exp(-\frac{2N\varepsilon^2}{9\beta^2\gamma(\mathbf{A})^2})], \end{aligned}$$

where C is a constant dependent only on n .

Proof. First we characterize the convergence rates of the realizations of $\mathcal{X}_0$, $\mathcal{X}_f$, and $\mathcal{A}$ using the procedure in Lemma 6.6.1. Note that

$$\begin{aligned} \|\mathbf{x}_0 - \frac{1}{N} \sum_{s_0=1}^N \mathbf{x}_0^{(s_0)}\| &\leq \eta(n) \max_i |[\mathbf{x}_0]_i - \frac{1}{N} \sum_{s_0=1}^N [\mathbf{x}_0^{(s_0)}]_i|, \\ \|\mathbf{x}_f - \frac{1}{N} \sum_{s_f=1}^N \mathbf{x}_f^{(s_f)}\| &\leq \eta(n) \max_i |[\mathbf{x}_f]_i - \frac{1}{N} \sum_{s_f=1}^N [\mathbf{x}_f^{(s_f)}]_i|, \end{aligned} \tag{6.11}$$

where $\eta(n)$ is a constant dependent on n such that $\|\mathbf{w}\| \leq \eta(n)\|\mathbf{w}\|_\infty$, $\forall \mathbf{w} \in \mathbb{R}^n$. Moreover,

$$\begin{aligned} \|\mathbf{A} - \frac{1}{N} \sum_{s_A(1)=1}^N \hat{\mathbf{A}}(s_A(1))\| &\leq \\ \mu(n) \max_{j \in \{1, \dots, n\}} \sum_{i \in \{1, \dots, n\}} |[\mathbf{A}]_{ij} - \frac{1}{N} \sum_{s_A(1)=1}^N [\hat{\mathbf{A}}(s_A(1))]_{ij}|, \end{aligned} \tag{6.12}$$

where $\mu(n)$ is a constant dependent on n such that $\|\mathbf{M}\| \leq \mu(n)\|\mathbf{M}\|_\infty$, $\forall \mathbf{M} \in \mathbb{R}^{n \times n}$.

Next, by using Hoeffding's inequality [119] and exploiting the Bernoulli nature of the random variables, the right hand side of the inequalities in (6.11) and (6.12) can be

bounded in probability for all $i \in \{1, \dots, n\}$ and for each $\varepsilon > 0$ as,

$$\begin{aligned}\Pr(|[\mathbf{x}_0]_i - \frac{1}{N} \sum_{s_0=1}^N [\mathbf{x}_0^{(s_0)}]_i| > \varepsilon) &\leq 2\exp(-\frac{2N\varepsilon^2}{\gamma(\mathbf{x}_0)^2}), \\ \Pr(|[\mathbf{x}_f]_i - \frac{1}{N} \sum_{s_f=1}^N [\mathbf{x}_f^{(s_f)}]_i| > \varepsilon) &\leq 2\exp(-\frac{2N\varepsilon^2}{\gamma(\mathbf{x}_f)^2}), \\ \Pr(|[\mathbf{A}]_{ij} - \frac{1}{N} \sum_{s_A(1)=1}^N [\widehat{\mathbf{A}}(s_A(1))]_{ij}| > \varepsilon) &\leq 2\exp(-\frac{2N\varepsilon^2}{\gamma(\mathbf{A})^2})\end{aligned}$$

Applying triangle inequality returns the results. ■

We can use the proof technique here to also characterize the rate of convergence in Lemma 6.5.1 considering stochastic ensemble approximations produced as in Lemma 6.6.1.

Lemma 6.6.3. (Variance of state trajectory). *Let $\mathcal{X}_0$, $\mathcal{A}$ and $\mathcal{A}$ be stochastic ensemble approximations of $\mathbf{x}_0$, $\mathbf{A}$ and $\mathbf{B}$ respectively, generated using Lemma 6.6.1. Then $\mathbf{x}(K)$ in Lemma 6.5.1 satisfies, for each $\varepsilon > 0$,*

$$\Pr(\|\mathbf{x}(K) - \frac{1}{N} \sum_{\sigma=1}^N \widehat{\mathbf{x}}(\sigma, K)\| > \varepsilon) \leq 2C(e^{c_1} + e^{c_2} + e^{c_3}),$$

where $c_1 = -\frac{2N\varepsilon^2}{9\|\mathbf{u}\|^2\gamma(\mathbf{B})^2}$, $c_2 = -\frac{2N\varepsilon^2}{9\|\mathbf{A}\|^2\gamma(x_0)^2}$, $c_3 = -\frac{2N\varepsilon^2}{9\|\mathbf{x}_0\|^2\gamma(\mathbf{A})^2}$ and C only depends on n .

Proof. The proof follows the same arguments as in the proofs of Lemmas 6.5.1 and 6.6.2. ■

6.6.1 Sample Averaging using Least Squares Error Minimization

In the previous section, we assigned uniform weights to each realization in order to produce the stochastic ensemble approximation. For each sample run of the process, we can make this averaging method better by introducing a least square error minimization with the samples produced from the vector sampling scheme in Lemma 6.6.1. Since the weights associated with the optimization problem are restricted to be in the simplex $\mathcal{S}^n$

(as per Definition 6.4.2), we approach this problem in two different ways. We describe each of them next and compare their performances later.

Accumulated Least Squares Error (ALSE) Minimization

In this approach, we take into account all the realizations of the stochastic ensemble approximations $\mathcal{X}_0, \mathcal{X}_f$, and $\mathcal{A}$ all at once. Then we compute the associated averaging weights from least squares error minimization and finally use these weights for the computed control. Next, we provide a closed form solution to the least squares error minimization problem by exploiting the entries of the vectors in Lemma 6.6.1.

The samples produced in Lemma 6.6.1 have a very particular structure to them. In fact, they have a non-zero entry in one component and have *zeros* everywhere else (*i.e.* their support is a singleton set). Hence, the whole problem boils down to adjusting the weights individually for the components in order to minimize the norm distance of the weighted sum of the samples and the vector. We provide this in the next result.

Lemma 6.6.4. (Least square error problem solution for singleton support samples). *Consider a vector $\mathbf{w} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$. Let $\bar{\mathbf{w}} := (1/\gamma(\mathbf{w}))\text{abs}(\mathbf{w})$ and suppose $\{\mathbf{s}^{(1)}, \dots, \mathbf{s}^{(N)}\}$ is a set of N sample vectors produced using the sampling scheme in Lemma 6.6.1. Let $\mathcal{S}$ be the matrix whose i^{th} column is $\mathbf{s}^{(i)}$. Then a solution to*

$$\min_{\mathbf{y} \in \mathcal{S}^N} \|\mathcal{S}\mathbf{y} - \mathbf{w}\|_2^2, \quad (6.13)$$

is given by $\mathbf{y}^$ with $[\mathbf{y}^*]_j = \alpha_j/n_j$, where $\alpha_j = |[\mathbf{w}]_k|$ with k such that $[\mathbf{s}^{(j)}]_k \neq 0$ and $n_j = |\{i \in \{1, \dots, N\} \mid [\mathbf{s}^{(i)}]_k \neq 0\}|$.*

Proof. Note that the problem is convex in $\mathbf{y}$ and that the set $\{1, \dots, n\}$ can be split into two disjoint sets $\mathcal{D}_1 := \{p \in \{1, \dots, n\} \mid \exists q \text{ s.t. } [\mathbf{s}^{(q)}]_p \neq 0\}$ and $\mathcal{D}_2 := \{1, \dots, n\} \setminus \mathcal{D}_1$. Now, since the support of $\mathbf{s}^{(i)}$ is a singleton, the problem can be rewritten as

$$\min_{\mathbf{y} \in \mathcal{S}^N} \sum_{i \in \mathcal{D}_1} \left(\sum_{j \in s(i)} [\mathbf{s}^{(j)}]_i \mathbf{y}_j - \mathbf{w}_i \right)^2 + \sum_{i \in \mathcal{D}_2} \mathbf{w}_i^2 =: J(\mathbf{y}),$$

where $s(i) := \{p \in \{1, \dots, N\} \mid [\mathbf{s}^{(p)}]_i \neq 0\}$. It is easy to see that for the solution $\mathbf{y}^*$ in the claim, the value of the cost function is $J(\mathbf{y}^*) = \sum_{i \in \mathcal{D}_2} \mathbf{w}_i^2$. As this is the minimum value of $J(\cdot)$, the proof is complete. $\blacksquare$

It is worth noting that if the set $\mathcal{D}_2$ in the proof of Lemma 6.6.4 becomes empty, then $J(\mathbf{y}^*) = 0$. This means that if the samples are rich enough so that all the non-zero entries of the original vector appear at least once in the set $\{\mathbf{s}^{(1)}, \dots, \mathbf{s}^{(N)}\}$, then the weights assigned using the least squares problem recreates the original vector perfectly. This does not translate to the case for matrices as then the associated weights to the matrix samples are coupled using multiple non-zero entries across the matrix, as illustrated in Section 6.7.

Streaming Least Squares Error (SLSE) Minimization

In this case, the controls are attained from sequentially changing the realizations of the vectors and matrices. For the uniform weights case, the solution can be made better by adding more samples. For the ALSE minimization case, the weights have to be determined beforehand and then the controls computed, since we do not assign the same weights to each sample. Moreover, as the number of samples increases, the weights assigned to the previous samples need to be recomputed in order to satisfy (6.13).

However for the SLSE minimization case, the weights to the controls can be updated sequentially in order to construct a sub-optimal solution to (6.13). Essentially, suppose $\mathbf{E}_1$ and $\mathbf{E}_2$ are estimates of $\mathbf{E}^*$, with $\mathbf{E}_1, \mathbf{E}_2$ and $\mathbf{E}^*$ being comparable entities from $\{\mathcal{X}_0, \mathcal{X}_f, \mathcal{A}\}$, *i.e.* they are arbitrary elements in the same sequence. Then we solve for

$$\min_{(w_1, w_2) \in \mathcal{S}^2} \|w_1 \mathbf{E}_1 + w_2 \mathbf{E}_2 - \mathbf{E}^*\|_2^2. \quad (6.14)$$

The computed controls are updated using the same weights. We explain this in Algorithm 4.

Algorithm 4. Sequential control computation with least square error estimation

Require: $\mathcal{X}_0, \mathcal{X}_f, \mathcal{A}, \mathbf{x}_0, \mathbf{x}_f, \mathbf{A}, \mathbf{B}, \text{errorBound}$

Ensure: control $\mathbf{v}(k)$

```

1: error  $\leftarrow \infty$ ;  $\hat{\mathbf{x}}_0 \leftarrow \hat{\mathbf{x}}(\sigma, 0) \in \mathcal{X}_0$ ;
2:  $\hat{\mathbf{x}}_f \leftarrow \mathbf{x}_f(s_f(\sigma)) \in \mathcal{X}_f$ ;  $\hat{\mathbf{A}} \leftarrow \hat{\mathbf{A}}(s_A(\sigma, 1)) \in \mathcal{A}$ ;
3:  $\mathbf{v} \leftarrow \hat{\mathbf{v}}$  such that  $\hat{\mathbf{x}}_f = \hat{\mathbf{A}}\hat{\mathbf{x}}_0 + \mathbf{B}\hat{\mathbf{v}}$ 
4: while error > errorBound do
5:   Randomly choose  $i \in \{1, 2, 3\}$ 
6:   if  $i = 1$  then
7:     update  $\hat{\mathbf{x}}_0$ 
8:      $(w_1^*, w_2^*) \leftarrow$  solution of (6.14) with  $\mathbf{E}_1 = \text{old } \hat{\mathbf{x}}_0, \mathbf{E}_2 = \text{new } \hat{\mathbf{x}}_0$  and  $\mathbf{E}^* = \mathbf{x}_0$ .
9:   else if  $i = 2$  then
10:    update  $\hat{\mathbf{x}}_f$ 
11:     $(w_1^*, w_2^*) \leftarrow$  solution of (6.14) with  $\mathbf{E}_1 = \text{old } \hat{\mathbf{x}}_f, \mathbf{E}_2 = \text{new } \hat{\mathbf{x}}_f$  and  $\mathbf{E}^* = \mathbf{x}_f$ .
12:   else if  $i = 3$  then
13:    update  $\hat{\mathbf{A}}$ 
14:     $(w_1^*, w_2^*) \leftarrow$  solution of (6.14) with  $\mathbf{E}_1 = \text{old } \hat{\mathbf{A}}, \mathbf{E}_2 = \text{new } \hat{\mathbf{A}}$  and  $\mathbf{E}^* = \mathbf{A}$ .
15:   end if
16:    $\mathbf{v}' \leftarrow \hat{\mathbf{v}}$  such that  $\hat{\mathbf{x}}_f = \hat{\mathbf{A}}\hat{\mathbf{x}}_0 + \mathbf{B}\hat{\mathbf{v}}$ 
17:   error  $\leftarrow \|(w_1^* - 1)\mathbf{v} + w_2^*\mathbf{v}'\|$ 
18:    $\mathbf{v} \leftarrow w_1^*\mathbf{v} + w_2^*\mathbf{v}'$ 
19: end while
```

We conclude this section by comparing the averaging methods provided earlier.

6.6.2 Comparison between the Averaging Methods

Since the weights obtained in the least squares error minimization are obtained from an optimization problem, we can compare their convergence rate with the uniform averaging scheme.

Lemma 6.6.5. (Comparison between averaging methods). *Suppose $\{\mathbf{E}_i\}_{i=1}^N$ are estimates of $\mathbf{E}^*$, with $\{\mathbf{E}_i\}_{i=1}^N$ and $\mathbf{E}^*$ being comparable entities from $\{\mathcal{X}_0, \mathcal{X}_f, \mathcal{A}\}$. Let $\mathbf{F}_{\text{uniform}} := (1/N) \sum_{i=1}^N \mathbf{E}_i$. Let $\mathbf{F}_{\text{ALSE}} := \sum_{i=1}^N w_i^* \mathbf{E}_i$ with $[w_1^*, \dots, w_N^*]$ being a solution of*

$$\min_{[w_1, \dots, w_N] \in \mathcal{S}^N} \left\| \mathbf{E}^* - \sum_{i=1}^N w_i \mathbf{E}_i \right\|_2^2. \quad (6.15)$$

Finally, let $\mathbf{F}_{\text{SLSE}} := \mathbf{F}(N)$ where $\mathbf{F}(N)$ is a solution to the difference equation $\mathbf{F}(n) = v_1^* \mathbf{F}(n-1) + v_2^* \mathbf{F}(n-2)$ starting from $\mathbf{F}(1) = \mathbf{E}_1, \mathbf{F}(2) = \mathbf{E}_2$ and where v_i^* comes from Algorithm 4. Then,

$$\|\mathbf{E}^* - \mathbf{F}_{\text{ALSE}}\| \leq \|\mathbf{E}^* - \mathbf{F}_{\text{SLSE}}\| \leq \|\mathbf{E}^* - \mathbf{F}_{\text{uniform}}\|. \quad (6.16)$$

Proof. Note that $[v_1^*, \dots, v_N^*]$ is a feasible solution to (6.15). This results in the first inequality. The second inequality holds since $(1/N)\mathbf{1}$ is a feasible solution to the optimization problem in Algorithm 4. ■

6.7 Simulations and Analysis

In this section, we provide simulation examples to validate our results in two scenarios. In the first case, we verify Lemma 6.6.5 in a power systems example. In the second case, we study the error in the computed state for a system with large dimensions. We use CVX MATLAB toolbox to solve the optimization problems.

6.7.1 Error in Computed Control

Here, we take an example system in the form of (6.1) from [120] using MATLAB's power toolbox. The system dimensions are $n = 10$ and $m = 5$ (with $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$)

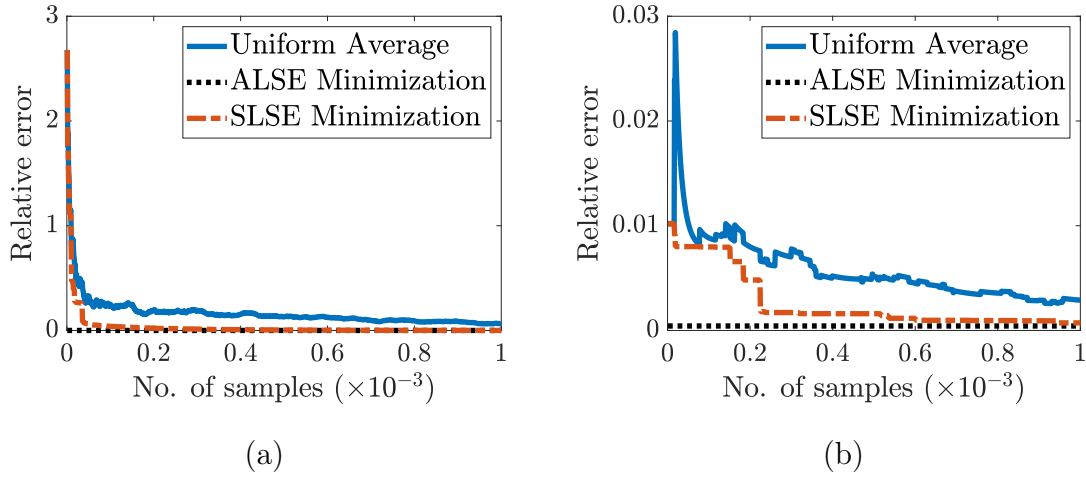


Figure 6.1. Relative error in estimation versus number of samples for example in Section 6.7.1. (a) Relative error in estimated initial condition $\mathbf{x}_0$. (b) Relative error in estimated system matrix $\mathbf{A}$.

and we choose the initial condition randomly.

First, we verify the ordering in estimation errors for the different averaging methods in Figure 6.1. This is in accordance with (6.16). Moreover, note that in Figure 6.1(a), it is possible to obtain *zero* relative error for the initial condition estimation using ALSE Minimization and SLSE Minimization as per the discussion following Lemma 6.6.4. Next, we deal with the errors in the computed control and the computed state. The goal is to drive the state close to the origin in one time step. The actual control that does this was computed by taking $\mathbf{u}_{\text{actual}} = \mathbf{B}^\dagger(-\mathbf{A}\mathbf{x}_0)$, where $\mathbf{B}^\dagger$ denotes the Moore-Penrose pseudoinverse. Then the three different controls $\mathbf{u}_{\text{computed}}$ were computed using the methods listed in this chapter (*i.e.* uniform average, ALSE minimization and SLSE minimization). The comparison of the relative error in computed control with respect to the number of samples is shown in Figure 6.2(a). Note that the $\mathbf{u}_{\text{computed}}$ for the ALSE minimization case was computed after considering all the samples. We also computed the relative error in the computed state with respect to the number of samples and present the results in Figure 6.2(b). It is worth mentioning that the error in the computed control and the

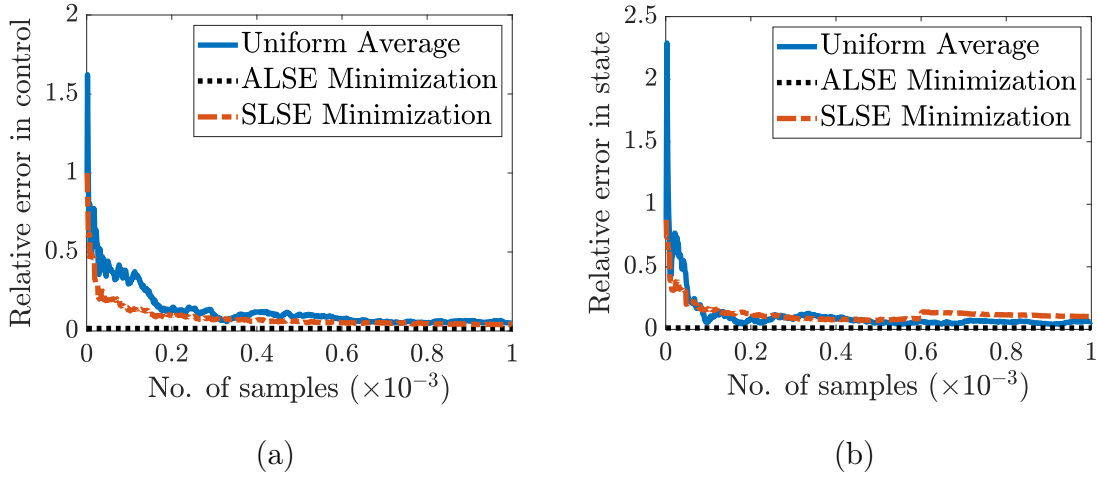


Figure 6.2. Relative error versus number of samples for example in Section 6.7.1. (a) Relative error in control $\frac{\|\mathbf{u}_{\text{computed}} - \mathbf{u}_{\text{actual}}\|}{\|\mathbf{u}_{\text{actual}}\|}$. (b) Relative error in state.

computed states do not always follow the same ordering as the estimation errors of the initial condition and the system matrix.

6.7.2 Error in Computed State

Here, we simulate our algorithms in a trajectory tracking problem for a system with large dimensions. Notice that the sample averaging method works for one time step controls only. Thus, it can be used to track a predefined trajectory of states. For this example, we take the system dimensions as $n = 100$, and $m = 80$ and generate $\mathbf{A} \in \mathbb{R}^{n \times m}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$, $\mathbf{x}(0) \in \mathbb{R}^n$, $\mathbf{x}(1) \in \mathbb{R}^n$, and $\mathbf{x}(2) \in \mathbb{R}^n$ randomly. The matrices $\mathbf{A}$, and $\mathbf{B}$ are used as the system matrices and $\mathbf{x}(0)$ is used as the initial condition. The states $\mathbf{x}(1)$, and $\mathbf{x}(2)$ are used as the reference trajectory.

Similar to the previous case, for each $t \in \{0, 1\}$, the control for each realization $\mathbf{A}^{(s)}$, $\mathbf{x}^{(s)}(t)$, and $\mathbf{x}^{(s)}(t+1)$ is computed as $\mathbf{u}_{\text{computed}} = \mathbf{B}^\dagger(\mathbf{x}^{(s)}(t+1) - \mathbf{A}^{(s)}\mathbf{x}^{(s)}(t))$. The relative error in the computed states with respect to the number of samples is given in Figure 6.3. Notice that (similar to what we observed in Section 6.7.1) the ordering given in Lemma 6.6.5 that holds for the estimation errors for the initial condition and the system

matrix, may not hold for the computed states as shown in Figure 6.3. Also, the errors in Figure 6.3 are much larger than the errors in Figure 6.2. This is because, the dimensions in this case are much larger than the previous case. Using the idea of convergence of the computed control in Lemma 6.5.2, we can give similar results for the convergence of computed states, omitted due to the lack of space.

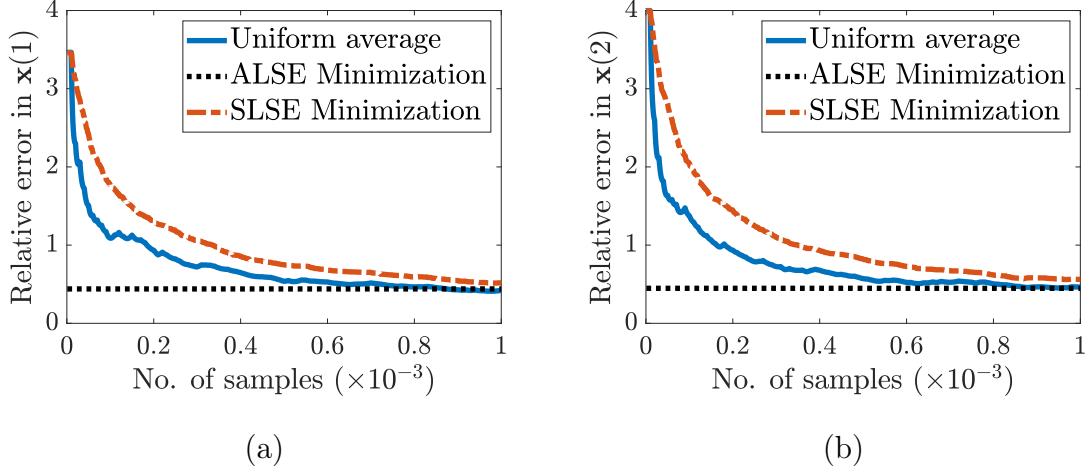


Figure 6.3. Relative error in state $\frac{\|\mathbf{x}_{\text{computed}} - \mathbf{x}_{\text{actual}}\|}{\|\mathbf{x}_{\text{actual}}\|}$ versus number of samples for example in Section 6.7.2. (a) Relative error in $\mathbf{x}(1)$. (b) Relative error in $\mathbf{x}(2)$.

6.8 Summary

In this chapter, we studied the reachability of a new class of stochastic ensemble systems, where are not required to have a continuum of ensembles. We provided a necessary condition for reachability under multiple time steps for the limiting DLTI system using approximate reachability for the stochastic ensemble system. We also provided a sufficient condition that averages the control obtained from the sample reachability condition for the ensemble system to find a control for the DLTI system. We provided convergence rates when the stochastic ensemble systems are produced using sampling. In the future, we will extend the sufficient condition to multi-time and continuous-time cases, and will study stochastic ensemble approximations of nonlinear systems.

This chapter a reprint of material as it appears in “Control of discrete-time LTI systems using stochastic ensemble systems,” N. Mandal, M. Khajenejad, and S. Martínez, in *SIAM Conf. on Control and Its Applications (CT)*, Philadelphia, PA, USA, July 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 7

Conclusion and Future Work

This dissertation studied distributionally robust games (DRG's) and established the applications of seeking their Nash equilibria (DRoNE's). Primary area of applications chosen was the problem of task allocation among self-interested agents.

Chapter 3 provided analysis related to the DRG's and then produced centralized and distributed algorithms that seek their DRoNE's. We presented conditions under which these DRoNE's exist, and related them to the NE of the associated stochastic game. Finally, we provided conditions on the parameters of the ISBRAG dynamics (and its distributed version d-ISBRAG) that ensure convergence to arbitrarily small regions around the DRoNE's. In the distributed case, each agent approximates the supergradients of their (possibly) non-smooth utility by exchanging information between their neighbors using a dynamic consensus protocol.

There are several worthwhile directions for future research, which includes extending the class of games compatible with this framework by introducing more general inertial terms. Another natural direction to explore is to characterize similar results for standard classes of games like potential games, aggregate games, and games with decision dependent strategies. This leaves room for the exploration of distributionally robust generalized Nash equilibria. Moreover, right now, the randomness does not depend on the strategies chosen by the agents. This motivated us to explore multi-stage Stackelberg games with

decision dependent uncertainty. This is related to an ongoing research project where we are interested in analyzing AI-assisted task allocation schemes with human agents in the loop. This gives us the opportunity to deploy robust algorithms that promotes truth telling in the face of humans trying to “game the system”. Finally, another natural research direction is to handle simultaneous communication and estimation updates among agents. The framework of passivity has gained significant attention in this regard, and combining passivity with non-smooth analysis could lead to useful algorithms for a broader class of applications.

Chapter 4 presented a game theoretic formulation of an optimal task allocation problem for a group of agents. By allowing agents to assign weights between *zero* and *one* for each task, we relaxed the combinatorial nature of the problem. This led to a partition and weight game, whose NE formed a superset of the optimal task partition. Then, we provided a distributed best-response projected gradient ascent by which convergence to the NE of the weight game was guaranteed. This is special case of the framework studied in Chapter 3.

Again, there are several directions of future research in this regard. One natural extension is to introduce constraints on number of tasks for each agent. This poses a significant challenge in the fact that it reintroduces the combinatorial nature of the problem. One way to address this is to produce a game formulation that encodes near optimal solutions to the aforementioned combinatorial problem. To that effect, it is worthwhile to look at the literature of submodular optimization, which has several results on sub-optimality guarantees through greedy methods. Another interesting direction of research is to address the situation where tasks are continually generated and the agents have to address them on line. This motivated us to introduce a precedence order (through a directed graph) on the tasks as part of an ongoing research project. The precedence graph can also be used to encode partial knowledge as tasks become visible (resp. executable) only when at least one (resp. all) of its neighbors are completed. We hope to use a

combination of on line optimization, game theory and submodular optimization to address this problem. Finally, another worthwhile direction of research is to generalize the setup to continuous space of tasks to account for coverage problems.

In Chapter 5, we proposed a novel algorithm for multi-robot exploration based on a distributed frontier assignment scheme. We used heterogeneous BE to evaluate frontiers and used a distributed game-theoretic framework to assign these. We found that a team of robots with heterogeneous behaviors outperforms a team of homogeneous robots with respect to time and distance costs. This allowed us to heuristically prescribe BE parameters for robot teams to efficiently explore maps.

A future direction of research is to provide an analytical frame to assess the efficacy of heterogeneous behaviors in teams of agents. This could be done by comparing prices of anarchy for different values of the behavioral parameter and establishing when it is useful.

In Chapter 6, we studied the reachability of a class of stochastic ensemble systems. We provided a necessary condition for reachability under multiple time steps for the limiting DLTI system using approximate reachability for the stochastic ensemble system. We also provided a sufficient condition that averages the control obtained from the sample reachability condition for the ensemble system to find a control for the DLTI system.

An immediate future direction of research is to extend the sufficient condition to multi-time and continuous-time cases, and further to study stochastic ensemble approximations of nonlinear systems. This can be achieved by using piece-wise affine bounds on the non-linearity and using the proposed method. Further, the analysis can be extended to broader classes of interactions between agents and not just Bernoulli ones.

Bibliography

- [1] F. Santambrogio, *Optimal Transport for Applied Mathematicians*. Springer, 2015.
- [2] P. Mohajerin Esfahani and D. Kuhn, “Data-driven distributionally robust optimization using the Wasserstein metric: performance guarantees and tractable reformulations,” *Mathematical Programming*, vol. 171, no. 1-2, pp. 115–166, 2018.
- [3] D. Boskos, J. Cortés, and S. Martínez, “Dynamic evolution of distributional ambiguity sets and precision tradeoffs in data assimilation,” in *European Control Conference*, Naples, Italy, June 2019, pp. 2252–2257.
- [4] E. Sontag, “Remarks on input to state stability of perturbed gradient flows motivated by model-free feedback control learning,” *Systems and Control Letters*, vol. 161, p. 105138, 2022.
- [5] C. M. Kellett and P. M. Dower, “A generalization of input-to-state stability,” in *IEEE Conf. on Decision and Control*, 2012, pp. 2970–2975.
- [6] N. Noroozi, R. Geiselhart, L. Grüne, B. S. Rüffer, and F. R. Wirth, “Nonconservative discrete-time iss small-gain conditions for closed sets,” *IEEE Transactions on Automatic Control*, vol. 63, no. 5, pp. 1231–1242, 2017.
- [7] E. D. Sontag and Y. Wang, “Notions of input to output stability,” *Systems and Control Letters*, vol. 38, no. 4, p. 235–248, 1999.
- [8] R. Diestel, “Graph theory,” *Graduate Texts in Mathematics*, pp. 173–207, 2017.
- [9] T. Basar, “Lecture notes on non-cooperative game theory,” *Game Theory Module of the Graduate Program in Network Mathematics*, pp. 3–6, 2010.
- [10] Y. Narahari, *Game theory and mechanism design*. World Scientific, 2014, vol. 4.
- [11] J. R. Marden, G. Arslan, and J. S. Shamma, “Cooperative control and potential games,” *IEEE Transactions on Systems, Man, & Cybernetics. Part B: Cybernetics*, vol. 39, no. 6, p. 1393–1407, 2009.
- [12] M. Zhu and S. Martínez, “Distributed coverage games for energy-aware mobile sensor networks,” *SIAM Journal on Control and Optimization*, vol. 51, no. 1, pp. 1–27, 2013.

- [13] Y. Pan and L. Pavel, “Games with coupled propagated constraints in optical networks with multi-link topologies,” *Automatica*, vol. 45, no. 4, pp. 871–880, 2009.
- [14] W. Saad, Z. Han, H. V. Poor, and T. Başar, “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.
- [15] 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.
- [16] P. Wankhede, N. Mandal, S. Martínez, and P. Tallapragada, “Opinion dynamics for utility maximizing agents: exploring the impact of resource penalty,” *IEEE Transactions on Control of Network Systems*, vol. 12, no. 1, pp. 5–17, 2025.
- [17] Q. Zhu and T. Başar, “Revisiting game-theoretic control in socio-technical networks: Emerging design frameworks and contemporary applications,” *IEEE Control Systems Letters*, 2025.
- [18] T. Roughgarden and I. Talgam-Cohen, “Approximately optimal mechanism design,” *Annual Review of Economics*, vol. 11, no. 1, pp. 355–381, 2019.
- [19] S. Hutchinson, B. Turan, and M. Alizadeh, “Safe pricing mechanisms for distributed resource allocation with bandit feedback,” *IEEE Transactions on Control of Network Systems*, vol. 11, no. 4, pp. 2010–2021, 2024.
- [20] 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.
- [21] P. Frihauf, M. Krstic, and T. Basar, “Nash equilibrium seeking for games with non-quadratic payoffs,” in *IEEE Conf. on Decision and Control*, Atlanta, USA, December 2010, pp. 881–886.
- [22] 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, 2017.
- [23] M. Ye, Q. Han, L. Ding, and S. Xu, “Distributed Nash equilibrium seeking in games with partial decision information: A survey,” *Proceedings of the IEEE*, vol. 111, no. 2, pp. 140–157, 2023.
- [24] J. Koshal, A. Nedić, and U. V. Shanbhag, “A gossip algorithm for aggregative games on graphs,” in *IEEE Conf. on Decision and Control*, 2012, pp. 4840–4845.
- [25] F. Salehisadaghiani and L. Pavel, “Distributed Nash equilibrium seeking: A gossip-based algorithm,” *Automatica*, vol. 72, pp. 209–216, 2016.

- [26] A. Nedić, A. Ozdaglar, and P. A. Parrilo, “Constrained consensus and optimization in multi-agent networks,” *IEEE Transactions on Automatic Control*, vol. 55, no. 4, p. 922–938, 2010.
- [27] S. Kia, B. V. Scoy, J. Cortés, R. Freeman, K. Lynch, and S. Martínez, “Tutorial on dynamic average consensus. The problem, its applications, the algorithms,” *IEEE Control Systems Magazine*, vol. 39, no. 3, pp. 40 – 72, 2018.
- [28] 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, 2018.
- [29] C. D. Persis and S. Grammatico, “Distributed averaging integral Nash equilibrium seeking on networks,” *Automatica*, vol. 110, p. 108548, 2019.
- [30] A. Kannan and U. V. Shanbhag, “Distributed iterative regularization algorithms for monotone Nash games,” in *IEEE Conf. on Decision and Control*, 2010, pp. 1963–1968.
- [31] R. T. Rockafellar, “Variational analysis of nash equilibrium,” *Vietnam Journal of Mathematics*, vol. 46, no. 1, pp. 73–85, 2018.
- [32] Y. Chen, C. Kizilkale, and M. Arcak, “Solving monotone variational inequalities with best response dynamics,” in *IEEE Conf. on Decision and Control*, 2024, pp. 1751–1756.
- [33] T. Cunis and I. Kolmanovsky, “Input-to-state stability of a bilevel proximal gradient descent algorithm,” *IFAC Papers Online*, vol. 56, no. 2, pp. 7474–7479, 2023.
- [34] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on Stochastic Programming: Modeling and Theory*. Philadelphia, PA: SIAM, 2014, vol. 16.
- [35] A. Ben-Tal, L. E. Ghaoui, and A. Nemirovski, *Robust optimization*. Princeton University Press, 2009.
- [36] S. Kim, R. Pasupathy, and S. G. Henderson, “A guide to sample average approximation,” *Handbook of simulation optimization*, pp. 207–243, 2014.
- [37] H. E. Scarf, K. Arrow, and S. Karlin, “A min-max solution of an inventory problem,” Rand Corporation Santa Monica, Tech. Rep., 1957.
- [38] D. Bertsimas and M. Sim, “The price of robustness,” *Operations Research*, vol. 52, no. 1, pp. 35–53, 2004.
- [39] C. Qu, H. Jia, and P. You, “Decision-dependent distributionally robust optimization with application to dynamic pricing,” in *IEEE Conf. on Decision and Control*, 2025, pp. 693–698.

- [40] D. Li and S. Martínez, “Online optimization and data assimilation with performance guarantees,” *IEEE Transactions on Automatic Control*, vol. 66, no. 5, pp. 2115–2129, 2020.
- [41] D. Boskos, J. Cortés, and S. Martínez, “High-confidence data-driven ambiguity sets for time-varying linear systems,” *IEEE Transactions on Automatic Control*, vol. 69, no. 2, pp. 797–812, 2023.
- [42] S. Qu, D. Meng, Y. Zhou, and Y. Dai, “Distributionally robust games with an application to supply chain,” *Journal of Intelligent & Fuzzy Systems*, vol. 33, no. 5, pp. 2749–2762, 2017.
- [43] Y. Liu, H. Xu, S. S. Yang, and J. Zhang, “Distributionally robust equilibrium for continuous games: Nash and Stackelberg models,” *European Journal of Operational Research*, vol. 265, no. 2, pp. 631–643, 2018.
- [44] J. Gao and H. Tembine, “Distributionally robust games: Wasserstein metric,” in *Int. Joint Conf. on Neural Networks*, 2018, pp. 1–8.
- [45] G. Peng, T. Zhang, and Q. Zhu, “A data-driven distributionally robust game using Wasserstein distance,” in *Int. Conf. on Decision and Game Theory for Security*. Springer, 2020, pp. 405–421.
- [46] F. Fabiani and B. Franci, “On distributionally robust generalized Nash games defined over the Wasserstein ball,” *Journal of Optimization Theory & Applications*, vol. 199, no. 1, pp. 298–309, 2023.
- [47] Z. Wang, G. Pantazis, S. Grammatico, M. M. Zavlanos, and K. H. Johansson, “Wasserstein distributionally robust nash equilibrium seeking with heterogeneous data: A lagrangian approach,” *arXiv preprint arXiv:2511.14048*, 2025.
- [48] N. Lanzetti, S. Fricker, S. Bolognani, F. Dörfler, and D. Paccagnan, “Strategically robust game theory via optimal transport,” *arXiv preprint arXiv:2507.15325*, 2025.
- [49] L. Liu, S. Liu, H. Xu, and D. E. Quevedo, “Incomplete-information dynamic Stackelberg equilibrium seeking by a distributed distributionally robust feedback approach,” *IEEE Transactions on Cybernetics*, 2025.
- [50] N. Mandal, A. Suresh, C. Nieto, and S. Martínez, “Behaviorally heterogeneous multi-agent exploration using distributed task allocation,” *IEEE Robotics and Automation Letters*, pp. 2983 – 2990, 2026.
- [51] H. L. Royden and P. Fitzpatrick, *Real analysis*. Macmillan New York, 1988, vol. 32.
- [52] C. Clason, *Introduction to functional analysis*. Springer Nature, 2020.
- [53] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

- [54] Z. Jiang, E. Sontag, and Y. Wang, “Input-to-state stability for discrete-time nonlinear systems,” *IFAC Proceedings and Volumes*, vol. 32, no. 2, pp. 2403–2408, 1999.
- [55] J. M. Danskin, *The theory of max-min and its application to weapons allocation problems*. Springer Science & Business Media, 2012, vol. 5.
- [56] A. Cherukuri and J. Cortés, “Distributed algorithms for convex network optimization under non-sparse equality constraints,” in *Allerton Conf. on Communications, Control and Computing*, Monticello, IL, Sep. 2016, pp. 452–459.
- [57] H. L. Choi, L. Brunet, and J. P. How, “Consensus-based decentralized auctions for robust task allocation,” *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 912–926, 2009.
- [58] F. Bullo, E. Frazzoli, M. Pavone, K. Savla, and S. L. . Smith, “Dynamic vehicle routing for robotic systems,” *Proceedings of the IEEE*, vol. 99, no. 9, pp. 1482–1504, 2011.
- [59] A. Sadeghi and S. L. Smith, “Heterogeneous task allocation and sequencing via decentralized large neighborhood search,” *Unmanned Systems*, vol. 5, no. 02, pp. 79–95, 2017.
- [60] H. W. Kuhn, “The Hungarian method for the assignment problem,” *Naval Research Logistics*, vol. 2, no. 1–2, p. 83–97, May 1955.
- [61] S. Chopra, G. Notarstefano, M. Rice, and M. Egerstedt, “A distributed version of the Hungarian method for multirobot assignment,” *IEEE Transactions on Robotics*, vol. 33, no. 4, pp. 932–947, 2017.
- [62] J. Cerquides, A. Farinelli, P. Meseguer, and S. D. Sarvapali, “A tutorial on optimization for multi-agent systems,” *The Computer Journal*, vol. 57, no. 6, pp. 799–824, 2014.
- [63] A. Prasad, S. Sundaram, and H. L. Choi, “Min-max tours for task allocation to heterogeneous agents,” in *IEEE Conf. on Decision and Control*, 2018, pp. 1706–1711.
- [64] N. Rezazadeh and S. S. Kia, “Distributed strategy selection: A submodular set function maximization approach,” *Automatica*, vol. 153, p. 111000, 2023.
- [65] J. Vondrák, “Optimal approximation for the submodular welfare problem in the value oracle model,” in *ACM Symposium on Theory of Computing*, 2008, pp. 67–74.
- [66] G. Calinescu, C. Chekuri, M. Pal, and J. Vondrák, “Maximizing a monotone submodular function subject to a matroid constraint,” *SIAM Journal on Computing*, vol. 40, no. 6, pp. 1740–1766, 2011.
- [67] S. Lloyd, “Least squares quantization in PCM,” *IEEE Transactions on Information Theory*, vol. 28, no. 2, p. 129–137, 1982.

- [68] M. Santos, Y. Diaz-Mercado, and M. Egerstedt, “Coverage control for multirobot teams with heterogeneous sensing capabilities,” *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 919–925, 2018.
- [69] M. Santos and M. Egerstedt, “Coverage control for multi-robot teams with heterogeneous sensing capabilities using limited communications,” in *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2018, pp. 5313–5319.
- [70] J. Cortés, S. Martínez, T. Karatas, and F. Bullo, “Coverage control for mobile sensing networks,” *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.
- [71] E. Frazzoli and F. Bullo, “Decentralized algorithms for vehicle routing in a stochastic time-varying environment,” in *IEEE Conf. on Decision and Control*, Paradise Island, Bahamas, Dec. 2004, pp. 3357–3363.
- [72] H. Aziz, A. Pal, A. Pourmiri, F. Ramezani, and B. Sims, “Task allocation using a team of robots,” *Current Robotics Reports*, vol. 3, no. 4, pp. 227–238, 2022.
- [73] S. Leitner, “Emergent task allocation and incentives: An agent-based model,” *Central European Journal of Operations Research*, pp. 1–29, 2024.
- [74] M. O. Afacan, “A task-allocation problem,” *Journal of Mathematical Economics*, vol. 82, pp. 285–290, 2019.
- [75] T. Demeulemeester, D. Goossens, B. Hermans, and R. Leus, “A pessimist’s approach to one-sided matching,” *European Journal of Operational Research*, vol. 305, no. 3, pp. 1087–1099, 2023.
- [76] R. Abebe, R. Cole, V. Gkatzelis, and J. D. Hartline, *A Truthful Cardinal Mechanism for One-Sided Matching*, 2020, pp. 2096–2113.
- [77] R. Konda, R. Chandan, D. Grimsman, and J. R. Marden, “Balancing asymptotic and transient efficiency guarantees in set covering games,” in *American Control Conference*, 2022, pp. 4416–4421.
- [78] A. C. Chapman, R. A. Micillo, R. Kota, and N. R. Jennings, “Decentralized dynamic task allocation using overlapping potential games,” *The Computer Journal*, vol. 53, no. 9, pp. 1462–1477, 2010.
- [79] S. Kim and M. Egerstedt, “Heterogeneous coverage control with mobility-based operating regions,” in *American Control Conference*, 2022.
- [80] R. W. Rosenthal, “A class of games possessing pure strategy Nash equilibria,” *International Journal of Game Theory*, vol. 2, no. 1, p. 65–67, 1973.
- [81] D. Monderer and L. S. Shapley, “Potential games,” *Games and Economic Behavior*, vol. 14, p. 124–143, 1996.

- [82] H. Durrant-Whyte and T. Bailey, “Simultaneous localization and mapping: Part i,” *IEEE Robotics and Automation Magazine*, vol. 13, no. 2, p. 99–110, 2006.
- [83] T. Bailey and H. Durrant-Whyte, “Simultaneous localization and mapping: Part ii,” *IEEE Robotics and Automation Magazine*, vol. 13, no. 3, p. 108–117, 2006.
- [84] Y. Tian, Y. Chang, F. H. Arias, C. Nieto-Granda, J. P. How, and L. Carlone, “Kimera-multi: Robust, distributed, dense metric-semantic slam for multi-robot systems,” *IEEE Transactions on Robotics*, vol. 38, no. 4, 2022.
- [85] C. Nieto, J. G. R. III, and H. I. Christensen, “Coordination strategies for multi-robot exploration and mapping,” *International Journal of Robotics Research*, vol. 33, no. 4, pp. 519–533, 2014.
- [86] G. W. et al., “Information theoretic MPC for model-based reinforcement learning,” in *IEEE Int. Conf. on Robotics and Automation*. IEEE, 2017, pp. 1714–1721.
- [87] S. Koga, A. Asgharivaskasi, and N. Atanasov, “Active exploration and mapping via iterative covariance regulation over continuous SE(3) trajectories,” in *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*. IEEE, 2021, pp. 2735–2741.
- [88] D. V. Knippenberg, L. H. Nishii, and D. J. G. Dwertmann, “Synergy from diversity: Managing team diversity to enhance performance,” *Behavioral Science & Policy*, vol. 6, no. 1, pp. 75–92, 2020.
- [89] A. Suresh, C. Nieto-Granda, and S. Martínez, “Robotic exploration using a generalized behavioral entropy,” *IEEE Robotics and Automation Letters*, vol. 9, no. 9, pp. 8011–8018, 2024.
- [90] B. Yamauchi, “Frontier-based exploration using multiple robots,” in *Proceedings of the second international conference on Autonomous agents*, 1998, pp. 47–53.
- [91] Z. S. et al., “Frontier detection and reachability analysis for efficient 2d graph-slam based active exploration,” in *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*.
- [92] C. E. Shannon, “A mathematical theory of communication,” *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.
- [93] D. Prelec, “The probability weighing function,” *Econometrica*, vol. 66, no. 3, pp. 497–527, 1998.
- [94] W. Suttle, A. Suresh, and C. Nieto-Granda, “Behavioral entropy-guided dataset generation for offline reinforcement learning,” in Arxiv.
- [95] N. Mandal, M. Khajenejad, and S. Martínez, “Distributed task allocation for self-interested agents with partially unknown rewards,” *IEEE Transactions on Automatic Control*, vol. 70, no. 9, pp. 6284–6291, 2025.

- [96] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, ser. Intelligent Robotics and Autonomous Agents. The MIT Press, 2005.
- [97] J. M. Amigó, S. G. Balogh, and S. Hernández, “A brief review of generalized entropies,” *Entropy*, vol. 20, no. 11, p. 813, 2018.
- [98] J. G. R. et al, “The DARPA SubT urban circuit mapping dataset and evaluation metric,” in *Int. Symposium on Experimental Robotics*. Springer, 2020, pp. 391–401.
- [99] R. W. Brockett, “On the control of a flock by a leader,” *Proceedings of the Steklov Institute of Mathematics*, vol. 268, no. 1, pp. 49–57, 2010.
- [100] J.-S. Li and N. Khaneja, “Control of inhomogeneous quantum ensembles,” *Physical review A*, vol. 73, no. 3, p. 030302, 2006.
- [101] S. J. Glaser, T. Schulte-Herbruggen, M. Sieveking, O. Schedletsky, N. C. Nielsen, O. W. Sørensen, and C. Griesinger, “Unitary control in quantum ensembles: Maximizing signal intensity in coherent spectroscopy,” *Science*, vol. 280, no. 5362, pp. 421–424, 1998.
- [102] Y. Zhai, I. Z. Kiss, and J. L. Hudson, “Control of complex dynamics with time-delayed feedback in populations of chemical oscillators: Desynchronization and clustering,” *Industrial & engineering chemistry research*, vol. 47, no. 10, pp. 3502–3514, 2008.
- [103] J. S. Li and N. Khaneja, “Ensemble control of linear systems,” in *IEEE Conf. on Decision and Control*, 2007, pp. 3768–3773.
- [104] K. Beauchard, J.-M. Coron, and P. Rouchon, “Controllability issues for continuous-spectrum systems and ensemble controllability of bloch equations,” *Communications in Mathematical Physics*, vol. 296, no. 2, pp. 525–557, 2010.
- [105] A. Becker and T. Bretl, “Approximate steering of a unicycle under bounded model perturbation using ensemble control,” *IEEE Transactions on Robotics and Automation*, vol. 28, no. 3, pp. 580–591, 2012.
- [106] J. Qi, A. Zlotnik, and J. S. Li, “Optimal ensemble control of stochastic time-varying linear systems,” *Systems & Control Letters*, vol. 62, no. 11, pp. 1057–1064, 2013.
- [107] A. Zlotnik and J. S. Li, “Synthesis of optimal ensemble controls for linear systems using the singular value decomposition,” in *American Control Conference*, 2012, pp. 5849–5854.
- [108] S. Wang and J.-S. Li, “Fixed-endpoint minimum-energy control of bilinear ensemble systems,” in *IEEE Conf. on Decision and Control*, 2015, pp. 1078–1083.
- [109] M. Chertkov, V. Chernyak, and D. Deka, “Ensemble control of cycling energy loads: Markov decision approach,” *Energy Markets and Responsive Grids: Modeling, Control, and Optimization*, pp. 363–382, 2018.

- [110] A. Hassan, R. Mieth, D. Deka, and Y. Dvorkin, “Stochastic and distributionally robust load ensemble control,” *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4678–4688, 2020.
- [111] J. S. Li and N. Khaneja, “Ensemble control of bloch equations,” *IEEE Transactions on Automatic Control*, vol. 54, no. 3, pp. 528–536, 2009.
- [112] R. Brockett and N. Khaneja, “On the stochastic control of quantum ensembles,” *System theory: modeling, analysis and control*, pp. 75 – 96, 2000.
- [113] J. S. Li and J. Qi, “Ensemble control of time-invariant linear systems with linear parameter variation,” *IEEE Transactions on Automatic Control*, vol. 61, no. 10, pp. 2808–2820, 2015.
- [114] J. S. Li, “Ensemble control of finite-dimensional time-varying linear systems,” *IEEE Transactions on Automatic Control*, vol. 56, no. 2, pp. 345–357, 2010.
- [115] L. Tie and J. S. Li, “On controllability of discrete-time linear ensemble systems with linear parameter variation,” in *American Control Conference*, 2016, pp. 6357–6362.
- [116] W. Zhang and J. S. Li, “Ensemble control on lie groups,” *SIAM Journal on Control and Optimization*, 2021.
- [117] M. Vidyasagar, “Statistical learning theory and randomized algorithms for control,” *IEEE Control Systems Magazine*, vol. 18, no. 6, pp. 69–85, 1998.
- [118] R. Tempo, G. Calafiore, and F. Dabbene, *Randomized algorithms for analysis and control of uncertain systems: with applications*. Springer, 2013, vol. 7.
- [119] W. Hoeffding, “Probability inequalities for sums of bounded random variables,” *Journal of the American Statistical Association*, vol. 58, no. 301, pp. 13–30, 1963.
- [120] F. Pasqualetti, F. Dorfler, and F. Bullo, “Attack detection and identification in cyber-physical systems,” *IEEE Transactions on Automatic Control*, vol. 58, no. 11, p. 2715–2729, 2013.