Strategic Interactions and Incentive Mechanisms on Multi-scale Networks

by

Kun Jin

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical and Computer Engineering)
in the University of Michigan
2023

Doctoral Committee:

Professor Mingyan Liu, Chair
Professor Tilman Börgers
Associate Professor Vijay Subramanian
Associate Professor Yevgeniy Vorobeychik
Professor Michael Wellman

Kun Jin

kunj@umich.edu

ORCID iD: 0000-0002-5293-2745

To my parents.

Acknowledgments

I have been fortunate to complete this dissertation with the help and support of amazing mentors, colleagues, collaborators, and friends. I want to start by thanking my advisor, Mingyan Liu, for giving me a chance to start this journey and keeping the position for me during my long visa wait, for her insights and encouragements, for sharing with me her knowledge of a wide range of research areas, and for giving me tremendous support. Without her, I wouldn't be able to grow as a researcher. I appreciate all the help and suggestions she provided for both my research and things beyond. It has been a great honor for me to have her as my advisor.

I am extremely grateful to Professor Yevgeniy Vorobeychik for providing great help in the fundamental work of this dissertation. I would also like to thank the researchers in the MURI project, especially Professor Yevgeniy Vorobeychik, Professor Michael Wellman, Professor Jeffrey Brantingham, and Professor David Kempe for the inspiring and thoughtful suggestions during the MURI meetings.

I would like to thank all professors I took courses from at the University of Michigan, especially Professor Vijay Subramanian, Professor Tilman Börgers, and Professor Demosthenis Teneketzis. I benefitted significantly from their excellent courses and learned important fundamental knowledge for my research.

During my time at the University of Michigan, I have had the privilege to work with many wonderful researchers: Professor Yevgeniy Vorobeychik, Professor Jefferey Brantingham, Professor David Kempe, Professor Bo Li, Dr. Charles Kamhoua, Armin Sarabi, Chaowei Xiao, Chenlan Wang, Demba Komma, Mahdi Khalili, Mehrdad Moharrami, Parinaz Naghizadeh, Tongxin Yin, Xiusi Chen, Xueru Zhang, Yang Liu, Ziyuan Huang, and Zhongzhu Chen. Thank you for your insights, and I have learned a lot from you.

Finally, I want to thank my parents for their endless love and support. Thank you for having my back and always believing in me, you gave me the courage to pursue doctoral study. I also want to express my gratitude to the rest of my family, thank you for your love and support, and thank you for taking care of my grandparents who brought me up together with my parents.

This work is supported by ARO under contract W911NF1810208 and by the NSF program on Fairness in AI in collaboration with Amazon under Award No. IIS-2040800.

TABLE OF CONTENTS

List of Figures

FIGURE

List of Tables

TABLE

List of Appendices

Abstract

The strategic interactions among a large number of interdependent agents are commonly modeled as network games. The research in network games has seen significant advances over the last decade and the network game framework allows us to model and solve real-world problems such as the provisioning of public goods, decision-making in cyber-physical systems, and the understanding of shock propagation in financial markets. In this thesis, we are interested in games on networks that enjoy various structural properties that arise naturally in many applications, such as groups, communities, and multi-relational interdependence, and seek to explore such structural properties in the analysis of these games. These properties often result in a multi-scale structure, in which agents can be grouped into larger communities/units, which can then be further grouped, and so on. These communities can be physical or logical, depending on what the graphical connectivity represents. We aim to develop analytical and algorithmic tools for studying this type of network game, with a particular interest in equilibrium analysis and the design of intervention and incentive mechanisms.

On the equilibrium analysis and computation front, the novelty of our work lies in the utilization of structural properties. In particular, we utilize the similarity among community members and propose structured conditions that significantly reduce the verification complexity of equilibrium properties such as existence, uniqueness, and stability. Similarly, for computation, we develop several algorithmic approaches that greatly reduce the computational complexity by leveraging the sparsity in a multi-scale structure, and we derive sufficient conditions for the convergence of these algorithms.

On the mechanism design and intervention front, we develop a novel multi-scale intervention model where agents form local groups and each group has a local planner. The planners are non-cooperative and their decisions are interdependent through the connections of the agents. We characterize the Stackelberg equilibrium of the system and study how the equilibrium efficiency is influenced by the network structure and the budget allocation of the planners. We also study the mechanism design and intervention using strategic classification and regression framework, where agents' actions include not only (honest) effort but also (dishonest) cheating, both may help the agent achieve the same decision outcome but only honest effort improves the planner's objective. We establish Stackelberg game models where the planner moves first by publishing and committing to an incentive mechanism that includes a decision rule (algorithm) as well as a subsidy mechanism,

followed by the agents' simultaneously best response. We model the agents' interdependence in such a game and show how the subsidy influences the agents' decision making and the resulting Stackelberg equilibria.

Chapter 1

Introduction

The strategic decision making process of agents who are connected through a network is often modeled as a *network game* [52, 45]. In network games, the utility of an agent depends on its own actions as well as the actions of other agents in its local neighborhood defined by interaction graphs and adjacency matrices. The network games framework can be used to capture different forms of interdependencies between agents' decisions, e.g., allowing agents' actions to be a strategic substitute or complement to each of its neighbors' efforts, the provision of public goods [3, 17, 53, 97], and security [41, 56, 95].

A typical formalization of a network game is as follows. The game contains N agents $a_1, \ldots, a_N$, connected by a graph $\mathcal{G}$. We typically denote the utility function of agent a_i as $u_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}; \mathcal{G})$, where $\boldsymbol{x}_i$ is the strategy/action by agent a_i, and $\boldsymbol{x}_{-i}$ is the strategies/actions by all agents other than a_i, and the topology of $\mathcal{G}$ determines the agents' interdependence in the game. A typical example of utility functions studied in network games is the linear-quadratic utility function [30],

$$u_i(x_i, \boldsymbol{x}_{-i}; G) := \underbrace{b_i x_i}_{\text{marginal benefit}} + \underbrace{\sum_{j \in \mathcal{N}_i} g_{ij} x_i x_j}_{\text{network influence}} - \underbrace{\frac{1}{2} x_i^2}_{\text{action cost}}$$

where the graph's influence is captured by the corresponding adjacency matrix $G = (g_{i,j})_{1 \le i,j \le N}$ (assume $g_{ii} = 0, \forall i$) and $\mathcal{N}_i = \{g_{ij} \neq 0\}$ is the index set of a_i's neighbors in the graph. The marginal benefit and action cost are the *individual components* while the *network influence* is jointly determined by other agents' utilities and the network. Agents will try to maximize their own utilities in network games, and an agent's action can influence other agents' decision-making even if they are not directly connected with each other in the network.

However, in many real world problems, there are often communities/groups in a network, where the connections within each group and between different groups can be significantly different in terms of connection strengths and frequencies, e.g., the Karate Club network in Figure 1.1. Therefore, a limitation of conventional network games is that all edges in the network are treated in the same way. This motivates us to extend the conventional framework to better fit the networks with group structures or hierarchies of group structures. In this thesis, we name such networks *multi-scale* networks. We first formalize how to represent the games on a multi-scale network, and then study how to leverage the structure and the representation to design more efficient methods to compute and analyze the equilibrium.

Equally important are the control and intervention problems in multi-scale network games. We will study how to utilize the multi-scale representation to study decentralized intervention, and how the group-level connectivity and welfare differences influence the system equilibria. We also generalize the mechanism design problem to allow multiple planners in the network, where the planners can either be cooperative or non-cooperative.

This thesis also studies mechanism design and interventions on network games with incomplete information, where a planner only has access to a particular type of partial information about the agents, in the form of some intermediate features but not the actual actions of the agents, while the agents can obtain the same feature using different actions, some honest but costly and others dishonest but cheaper. Honest actions contribute to the planner's goals, while dishonest actions can lead to feature improvement but are not beneficial to the planner's goals. It is thus important for the planner to induce honest effort in such a setting. We model honest and dishonest actions under a strategic classification and regression framework, where the planner designs decision rules in the form of classifiers or regression functions and the agents best respond and receive decision outcomes, which appear as the benefit terms in their utility function. Therefore, the decision rules can be viewed as an intervention. This framework also provides a useful formalism for modeling honest and dishonest actions, treated as improvement and gaming actions, respectively, where, while both can improve the decision outcome for the agents, only the improvement action contributes to the planner's objective. We also explore the use of a subsidy mechanism when the decision rule itself cannot sufficiently incentivize honest actions.

In the next few sections we elaborate on the key concepts examined in this thesis.

1.1 Network Structure

Network structures arise naturally when agents have different levels of proximity, either in terms of distance in the physical world, or in terms of logical measurements like values or interests in the virtual world. Agents in the same location or having similar interests tend to form communities or groups on the network, and members within the same group may have similar behavior and share similar properties.

The Karate Club network [32] is a famous network with group structures, where the connections represent out-of-club interactions of the individuals; we can think of each group in Figure 1.1 as a community of friends. Communities can also contain agents that have similar roles in the

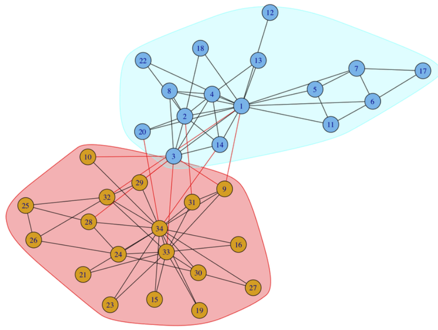


Figure 1.1: Communities in the Karate Club network.

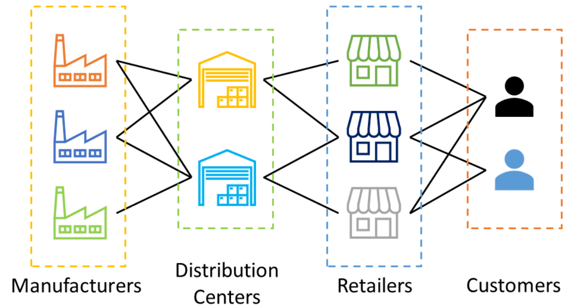


Figure 1.2: The Supply Chain Network with Groups.

network. For example, in the supply chain network in Figure 1.2, all manufacturers can be treated as a manufacturer group.

Such structures present both opportunities and challenges in analysis. On one hand, the emergence of such a community structure implies some type of *sparsity* in the network – as shown in Fig. 1.1 the connectivity between the two groups is far less than that within the same group – that we may be able to exploit. On the other hand, the same type of community structures also implies *heterogeneity* – groups must be distinct in some way to be different groups; this is seen in the different roles each group plays in Fig. 1.2. This heterogeneity often presents significant challenges in analysis; for instance, groups may have multiple non-cooperative local planners, making control and intervention fundamentally different from the conventional, single social planner model.

1.2 Taking Advantage of Network Structure in Analysis and Computation

The community structure on a network captures the similarity among groups of agents. We will utilize this fact to derive structured conditions used to verify equilibrium properties and algorithms to compute the equilibria. Such verification and computation are shown to be much more efficient than conventional methods that treat the network as flat.

One way of utilizing the structural information is *by abstracting* the different communities: treat each group as a super node, and create a super network accordingly. Each super node summarizes the behavior of a group of agents and the connections between the super nodes represent a summary of the group level interactions. By utilizing the network structure, we come up with a multi-scale representation of the network by formally introducing nodes on different levels. With large enough group sizes, the super network has a much smaller size than the original network while still being able to capture important information from the original network under certain conditions. We design novel NE computation algorithms based on the multi-scale representation of the game that greatly improves the computational speed while guaranteeing convergence to the same Nash equilibrium.

1.3 Utilizing Network Structure in Control and Intervention

The different communities in the network also represent the different interests of agents, and thus we extend the conventional single planner mechanism design framework to a multiple non-cooperative planners framework.

Group structures are a common phenomenon across networks of all types, be it social, technological, political, or economic, the multi-planner modeling consideration allows us to investigate a number of interesting features that often arise in realistic strategic and decentralized decision making in network games. These planners may or may not have common interests and their interventions and mechanisms will typically be interdependent. We establish a 2-level Stackelberg game model and show that the intervention design can be viewed as a higher level network game between the planners. We are particularly interested in the efficiency of decentralized decision making and how the efficiency is influenced by the budget allocation among the planners. We discover that the efficiency of the Stackelberg equilibrium can be decomposed into level specific terms, and help us build a multi-scale view of the efficient and inefficient aspects of the multi-planner game.

1.4 Strategic Classification and Regression as Intervention

A growing literature in strategic classification and regression shows that decision rules, i.e., classifiers or regression functions, can be viewed as incentive mechanisms in a game where a planner cannot observe the agents' actions but only a proxy, i.e., features, which allows the agents to use different actions (honest/improvement vs. dishonest/cheating actions) to attain the same feature improvement and thus favorable outcome but only honest/improvement actions benefit the planner. A common approach in this type of strategic learning problem is to use the Stackelberg game models where the planner moves first by publishing and committing to a decision rule (algorithm) and then the agents simultaneously and independently best respond to the deployed algorithm. We identify situations where the decision rule itself can never incentivize improvement actions, and introduce novel incentive mechanisms for strategic classification and regression problems to address this issue. Our approach is to include an additional subsidy mechanism published in the first stage which gives discounts on certain actions; the agents simultaneously best respond to the decision rule accounting for the subsidy. We formalize the subsidy as a discount mechanism, and show how such subsidies induce a uniformly better outcome for both the decision maker and the agents. Furthermore, when the subsidy is implemented by a third party who is also a first-stage mover, the third party can not only uniformly improve every party's utility, but also alleviate fairness issues among different subgroups in the system.

1.5 Outline of the Thesis

This thesis aims to develop a theoretical framework for modeling and analyzing strategic interactions and incentive mechanisms in network games with multi-scale structures. We focus on two broad areas, *equilibrium analysis* and *mechanism design*. We present novel modeling techniques and algorithms to build non-trivial extensions to the existing literature. A review of the literature is given in Chapter 1.6.

Chapters 2 and 3 present methods for modeling networks with multi-scale structures and the corresponding equilibrium analysis and computation.

In Chapter 2, we formally define the games on multi-scale networks and establish a representation of the multi-scale network games. We contrast the multi-scale perspective of the game with the conventional flat perspective. We then develop efficient algorithmic approaches that leverage this multi-scale structure and perspective and derive sufficient conditions for the convergence of our algorithms to a Nash equilibrium. Chapter 2 is based on our paper [49] "Multi-Scale Games: Representing and Solving Games on Networks with Group Structure" that appeared in the Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI) in 2021.

In Chapter 3, we extend the above study to *multi-relational* networks, where the connectivity among agents are action-dimension dependent, i.e., the network is now multiple networks superimposed together. This results in an extended adjacency matrix where each entry maps to an *agent-action component*. Structural information (such as clustering) maps to certain partitions and blocks over this matrix and is utilized to derive a set of *structured* conditions that guarantee the existence and uniqueness of a Nash equilibrium. We show that these structured conditions are computationally much easier and less costly to verify than those derived from prior methods that ignore the structural information inherent in the network with very little loss in accuracy (tightness of the sufficient conditions). Chapter 3 is a much more extended version of our paper [47] "Games on Networks with Community Structure: Existence, Uniqueness and Stability of Equilibria" that appeared in the Proceedings of the IEEE American Control Conference (ACC) in 2020.

Chapters 4 and 5 study the mechanism design and intervention in network games, games with multiple sub-populations, and multi-scale networks.

Chapter 4 presents a multi-planner intervention mechanism in network games with communities. We establish a Stackelberg game model where the planners move first to design intervention and agents move next in the intervened game. The game between the planners can be studied as a higher level network game among themselves and we derive the Stackelberg equilibrium for this system and study its efficiency properties. We also show that the efficiency loss can be decomposed into level specific efficiency loss components, i.e., a planner-level loss term and an agent-level loss term, where the planner-level loss term is jointly determined by the structure of the network and the budget allocation. Whether the budget is transferable also significantly influences the system outcome, and we show that even selfish planners may have incentives to share the budget with neighboring groups when positive externalities exist in the network, which resembles real world behaviors like vaccine donations. These analyses also provide useful insights into the global budget allocation problem of a planner at one level higher than the local group planners. Chapter 4 is based on our paper [48] "Multi-planner Intervention in Network Games with Community Structures" that appeared in the Proceedings of the 60th IEEE Conference on Decision and Control (CDC) in 2021.

Chapter 5 studies how we can use strategic classification and regression algorithms along with subsidies to intervene in a game with multiple sub-populations in the system. We use the Stackelberg game model to study the system and focus on two specific mechanism design scenarios. The first is a two-party system where the algorithm designer also designs the supportive subsidy mechanism in the first stage, and the agents will simultaneously best respond to both the decision rule and the subsidy in the second stage. The second is a three-party system where the algorithm designer designs the decision rule, and a third-party designs the subsidy in the first stage, and the agents still best respond to both in the second stage. The third-party has social well-being metrics as its objectives, which is different from the algorithm designer's selfish objective, and thus designs the subsidy differently. We show how a supportive subsidy mechanism, combined with the machine learning algorithm can serve as a meta incentive mechanism that can improve the algorithm robustness, the efficiency, the fairness, and the agents' utilities simultaneously. We also study how the two-party and three-party systems result in different Stackelberg equilibria, and how the chosen social well-being metric influences the Stackelberg equilibria. Chapter 5 is based on our paper [51] "Incentive Mechanisms in Strategic Classification and Regression Problems" that appeared in the Proceedings of the ACM Economics and Computation (EC) in 2022.

1.6 Literature Review

We next review the literature most relevant to this thesis, where these prior works are related to at least one of the following topics: equilibrium analysis and computation in network games, multi-scale structures in networks, mechanism and intervention design in network games, strategic classification and regression, and learning in network games. In the remainder of this section, we use *conventional* models to refer to the prior work and specifically point out the novelty and differences of our work.

Equilibrium analysis and computation. Conventional (unstructured) network games and their equilibrium outcomes have been studied in a variety of application areas, including the private provision of public goods [3, 17, 53], security decision making in interconnected cyber-physical

Related Works	Network Games	Multi-scale Structure	Mechanism Design & Intervention	Strategic Classification & Regression	Learning in Games
[80, 70, 71, 72, 74]	✓		✓		
[62, 77, 4]	✓	✓			
[57, 7, 6]	✓				✓
[16, 35, 67, 34, 54, 65, 42]				✓	
[81, 27, 14, 59, 96, 43, 44, 66, 83]				✓	✓
[60, 75, 82]	✓			✓	✓
Chapter 2 & 3	✓	✓			
Chapter 4	✓	✓	✓		
Chapter 5			✓	✓	
Chapter 6	✓	✓	✓	✓	

Table 1.1: A Summary of Our Works and Related Works.

systems [41, 56], and shock propagation in financial markets [1]. A common line of research in this literature is to study the effect of the network structure on the existence, uniqueness, and stability of the game equilibria (see [45] for a survey). In particular, unstructured network games with linear best-response functions have been studied in [10, 68, 74]. Bramoulle *et al.* [10] uncovered the importance of the lowest eigenvalue of the adjacency matrix of the network in determining the uniqueness and stability of the Nash equilibrium. Miura-Ko *et al.* [68] show that if the adjacency matrix of the network is strictly diagonally dominant, then the Nash equilibrium is unique. Naghizadeh and Liu [74] identify necessary and sufficient conditions for the existence and uniqueness of Nash equilibria in network games with linear best responses by establishing a connection to linear complementarity problems.

Unstructured network games with nonlinear best-response functions have been studied in [3, 2, 100, 63, 79, 73]. Allouch [3] introduces a sufficient condition for the uniqueness of Nash equilibrium called *network normality* which imposes lower and upper bounds on the derivative of Engel curves. Acemoglu *et al.* [2] consider a network game with idiosyncratic shocks and show that if the best response mapping is either a contraction (Lipschitz with constant strictly smaller than one), or a bounded non-expansive mapping, then the game has a unique Nash equilibrium. Zhou *et al.* [100] establish a connection between nonlinear complementary problems (NCP) and

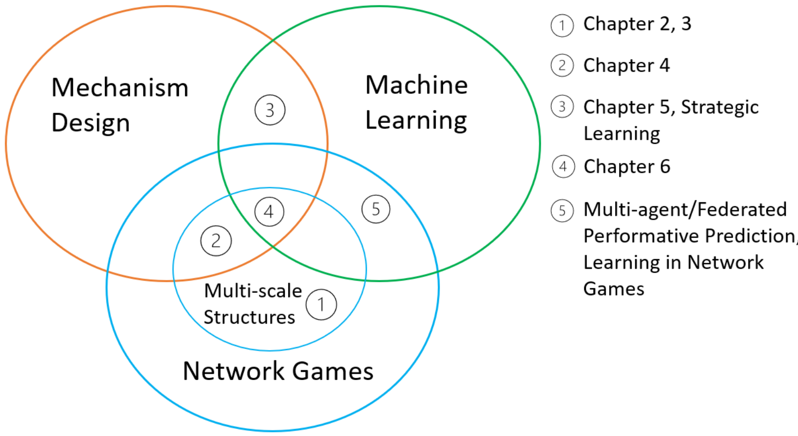


Figure 1.3: A Venn Diagram for the Sections and Related Works.

network games, and use existing results from the NCP literature to find sufficient conditions for the uniqueness of Nash equilibria. The works of [63, 79, 73] use the framework of variational inequalities to study network games with nonlinear best responses. Naghizadeh and Liu [73] show that a sufficient condition for the uniqueness and stability of the Nash equilibrium can be determined by the lowest eigenvalue of matrices constructed based on the slope of the agents' interaction functions and the intensity of their interactions. Parise and Ozdaglar [79] identify an operator in the variational inequality problem which involves the derivative of the agent's cost with respect to its action, and show that various properties of this operator and its Jacobian will determine conditions for the existence, uniqueness, and stability of a Nash equilibrium.

Multi-scale structures. While network games are a powerful modeling framework, most previous works fail to capture a common feature of human organization: groups and communities. Indeed, the investigation of communities, or close-knit groups, in social networks is a major research thread in network science. Moreover, such groups often have a hierarchical structure [22, 32]. For example, strategic interactions among organizations in a marketplace often boil down to interactions among their constituent business units, which are, in turn, comprised of individual decision makers. In the end, it is those lowest-level agents who ultimately accrue the consequences of these interactions (for example, corporate profits would ultimately benefit individual shareholders). Moreover, while there are clear interdependencies among organizations, individual utilities are determined by a combination of individual actions of some agents, together with *aggregate* decisions by the groups (e.g., business units, organizations). For example, an employee's bonus is determined in part by their performance in relation to their co-workers, and in part by how well their employer (organization) performs against its competitors in the marketplace. Previous works on network games that involve group or community structure focus on finding such structures; e.g., community detection in networks using game theoretic methods have been studied in [62, 77, 4]. In contrast, our works in Chapters 2 and 3 focus on utilizing the multi-scale structures in the equilibrium analysis and computation.

Mechanisms and interventions. Mechanisms [80, 70, 71, 72, 74] and interventions [30] are implemented by the planner in a network game to shape the strategic interaction of the agents and to incentivize them to take certain actions that optimize the planner's objective like the social welfare. *Interventions* in a network game typically refers to changes in certain game parameters made by a planner with a budget constraint, who wishes to induce a more socially desirable outcome (in terms of social welfare) under the revised game. A prime example is the study presented in [30], where the intervention takes the form of changing the agents' standalone marginal benefit terms (in a linear quadratic utility model) and changes are costly; this is done by a central/global planner, who wishes to find the set of interventions that lead to the highest equilibrium social welfare subject to a cost constraint. These interventions are different from more conventional mechanisms such as auctions, but they can be very effective in specific problems settings. The overall Stackelberg game structure is also what we will primarily utilize in this thesis.

Strategic classification and regression. The strategic classification and regression problem is modeled as a sequential (two-stage) Stackelberg game where the planner is the first-mover, who designs, publishes, and commits to a decision rule (e.g., a classifier); the agents simultaneously best-respond to the decision rule by the planner, by manipulating their input features to obtain a desirable decision outcome so as to maximize their utilities. Previous works in this field [16, 35, 67, 34, 54, 65, 42] study the problem where one or more agents strategically, and independent from other agents, choose actions to manipulate their features in response to the published machine learning decision rule while the planner seeks to design the optimal decision rule in anticipation of the agents' manipulation.

Performative predictions. Performative predictions [81] study the problem where the data distributions shifts with the deployed decision models. Typical scenarios of performative prediction include *strategic classification and regression* [35, 26, 67, 42, 12, 20, 65, 91, 34, 54, 101]. Performative prediction has been primarily studied in a centralized setting, with pertinent literature studying the algorithm convergence [64, 27, 14, 59, 96] and algorithm development [43, 44, 66, 83].

More recently, [60] formalized multi-agent/player performative predictions, where agents can communicate over a network and try to learn a common decision rule. The agents have heterogeneous distribution shifts (responses) to the model, and the authors study the convergence of decentralized algorithms to the performative stable (PS) solution. Decentralized performative predictions capture the heterogeneity in agents'/clients' responses to the decision model and avoid centralized data collection for training. [75] propose a decentralized multi-player performative prediction framework where the players react to competing institutions' actions. [82] proposes a replicator dynamics model with label shift.

Learning in games. Learning in games [6, 7, 57] focus on mechanism design problems where the planner or the agents have incomplete knowledge of the parameters in the game. These works study how to learn both the network structure and payoffs of games from data.

Part I

Utilizing the Multi-Scale Network Structure for Equilibrium Computation and Analysis

Chapter 2

Multi-Scale Games: Representing and Solving Games on Networks with Group Structure

2.1 Introduction

Network games provide a natural machinery to compactly represent strategic interactions among agents whose payoffs exhibit sparsity in their dependence on the actions of others. Besides encoding interaction sparsity, however, real networks often exhibit a multi-scale structure, in which agents can be grouped into communities, those communities further grouped, and so on, and where interactions among such groups may also exhibit sparsity. We present a general model of multi-scale network games that encodes such a multi-level structure. We then develop several algorithmic approaches that leverage this multi-scale structure, and derive sufficient conditions for the convergence of these to a Nash equilibrium. Our numerical experiments demonstrate that the proposed approaches enable orders of magnitude improvements in scalability when computing Nash equilibria in such games. For example, we can solve previously intractable instances involving up to 1 million agents in under 15 minutes.

Strategic interactions among interconnected agents are commonly modeled using the network, or graphical, game formalism [52, 45]. In such games, the utility of an agent depends on his own actions as well as those of its network neighbors. Many variations of games on networks have been considered, with applications including the provision of public goods [3, 17, 53, 97], security [41, 56, 95], and financial markets [1].

While network games are a powerful modeling framework, they fail to capture a common feature of human organization: groups and communities. Indeed, the investigation of communities, or close-knit groups, in social networks is a major research thread in network science. Moreover, such groups often have a hierarchical structure [22, 32]. For example, strategic interactions among organizations in a marketplace often boil down to interactions among their constituent business units, which are, in turn, comprised of individual decision makers. In the end, it is those lowest-level agents who ultimately accrue the consequences of these interactions (for example, corporate profits would ultimately benefit individual shareholders). Moreover, while there are clear interdependencies

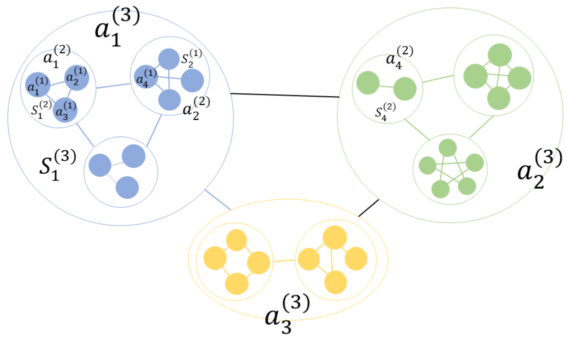


Figure 2.1: An Illustration of a Multi-scale (3-Level) Network.

among organizations, individual utilities are determined by a combination of individual actions of some agents, together with *aggregate* decisions by the groups (e.g., business units, organizations). For example, an employee's bonus is determined in part by their performance in relation to their co-workers, and in part by how well their employer (organization) performs against its competitors in the marketplace.

We propose a novel *multi-scale game model* that generalizes network games to capture such hierarchical organizations of individuals into groups. Figure 2.1 offers a stylized example in which three groups (e.g., organizations) are comprised of 2-3 subgroups each (e.g., business units), which are in turn comprised of 2-5 individual agents. Specifically, our model includes an explicit hierarchical network structure that organizes agents into groups across a series of levels. Further, each group is associated with an action that deterministically aggregates the decisions of its constituent agents. The game is grounded at the lowest level, where the agents are associated with scalar actions and utility functions that have a modular structure in the strategies taken at each level of the game. For example, in Figure 2.1, the utility function of an individual member a_j of level-3 group $a_3^{(3)}$ is a function of the strategies of (i) a_j's immediate neighbors (represented by links between pairs of filled-in circles), (ii) a_j's level-2 group and its network neighbor (the small hollow circles), and (iii) a_j's level-3 group, $a_3^{(3)}$ (large hollow circle) and its network neighbors, $a_1^{(3)}$ and $a_2^{(3)}$.

Our next contribution is a series of iterative algorithms for computing pure strategy Nash equilibria that explicitly leverage the proposed multi-scale game representation. The first of these simply takes advantage of the compact game representation in computing equilibria. The second algorithm we propose offers further innovation through an iterative procedure that alternates between game levels, treating groups themselves as pseudo-agents in the process. We present sufficient conditions for the convergence of this algorithm to a pure strategy Nash equilibrium through a connection to Structured Variational Inequalities [39], although the result is limited to games with two levels. To address the latter limitation, we design a third iterative algorithm that now converges even in games with an arbitrary number of levels.

Our final contribution is an experimental evaluation of the proposed algorithms compared to best response dynamics. In particular, we demonstrate orders of magnitude improvements in scalability, enabling us to solve games that cannot be solved using a conventional network game representation.

Related Work: Network games have been an active area of research; see e.g., surveys by [45] and [11]. We now review the most relevant papers. Conditions for the existence, uniqueness, and stability of Nash equilibria in network games under general best responses are studied in [79, 73, 88, 10]. Variational inequalities (VI) are used in these works to analyze the fixed point and contraction properties of the best response mappings. It is identified in [79, 73, 88] that when the Jacobian matrix of the best response mapping is a P-matrix or is positive definite, a feasible unique Nash equilibrium exists and can be obtained by best-response dynamics [88, 79]. In this chapter, we extended the analysis of equilibrium and best responses for a conventional network game to that in a multi-scale network game, where the utility functions are decomposed into separable utility components to which best responses are applied separately. This is similar to the generalization from a conventional VI problem to an SVI problem [39, 38, 40, 8] problem.

Previous works on network games that involve group or community structure focus on finding such structures; e.g., community detection in networks using game theoretic methods have been studied in [62, 77, 4]. By contrast, our work focuses on analyzing a network game with a given group/community structure and using the structure as an analytical tool for the analysis of equilibrium and best responses.

2.2 Preliminaries

A general *normal-form game* is defined by a set of agents (players) $I = \{1, \ldots, N\}$, with each agent a_i having an action/strategy space K_i and a utility function $u_i(x_i, \boldsymbol{x}_{-i})$ that i aims to maximize; $x_i \in K_i$ and x_{-i} denotes the actions by all agents other than i. We term the collection of strategies of all agents $\boldsymbol{x}$ a *strategy profile*. We assume $K_i \subset \mathbb{R}$ is a compact set.

We focus on computing a *Nash equilibrium (NE)* of a normal-form game, which is a strategy profile with each agent maximizing their utility given the strategies of others. Formally, $\boldsymbol{x}^*$ is a *Nash equilibrium* if for each agent i,

$$x_i^* \in \underset{x_i \in K_i}{\text{argmax}} \; u_i(x_i, \boldsymbol{x}_{-i}^*). \tag{2.1}$$

A *network game* encodes structure in the utility functions such that they only depend on the actions of network neighbors. Formally, a network game is defined over a weighted graph (I, E), with each node an agent and E is the set of edges; the agent's utility $u_i(x_i, \boldsymbol{x}_{-i})$ reduces to

$u_i(x_i, \boldsymbol{x}_{I_i})$, where I_i is the set of network neighbors of i, although we will frequently use the former for simplicity.

An agent's best response is its best strategy given the actions taken by all the other agents. Formally, the best response is a set defined by

$$BR_i(\boldsymbol{x}_{-i}, u_i) = \underset{x_i}{\text{argmax}}\ u_i(x_i, \boldsymbol{x}_{-i}). \tag{2.2}$$

Whenever we deal with games that have a unique best response, we will use the singleton best response set to also refer to the player's best response strategy (the unique member of this set).

Clearly, a NE of a game is a fixed point of this best response correspondence. Consequently, one way to compute a NE of a game is through *best response dynamics (BRD)*, which is a process whereby agents iteratively and asynchronously (that is, one agent at a time) take the others' actions as fixed values and play a best response to them.

We are going to use this BRD algorithm as a major building block below. One important tool that is useful for analyzing BRD convergence is *Variational Inequalities (VI)*. To establish the connection between NE and VI we assume the utility functions $u_i, \forall i = 1, \dots, N$, are continuously twice differentiable. Let $K = \prod_{i=1}^{N} K_i$ and define $F : \mathbb{R}^N \to \mathbb{R}^N$ as follows:

$$F(\boldsymbol{x}) := \Big(- \nabla_{x_i} u_i(\boldsymbol{x}) \Big)_{i=1}^{N} . \tag{2.3}$$

Then $\boldsymbol{x}^*$ is said to be a solution to VI(K, F) if and only if

$$(\boldsymbol{x} - \boldsymbol{x}^*)^T F(\boldsymbol{x}^*) \geq 0,\ \forall \boldsymbol{x} \in K\ . \tag{2.4}$$

In other words, the solution set to VI(K, F) is equivalent to the set of NE of the game. Now, we can define the condition that will guarantee the convergence of BRD.

Definition 1. ***The*** P_Υ ***condition****: The* Υ *matrix generated from* $F : \mathbb{R}^N \to \mathbb{R}^N$ *is given as follows*

$$\Upsilon(F) = \begin{bmatrix} \alpha_1(F) & -\beta_{1,2}(F) & \cdots & -\beta_{1,N}(F) \\ -\beta_{2,1}(F) & \alpha_2(F) & \cdots & -\beta_{2,N}(F) \\ \vdots & \vdots & \ddots & \vdots \\ -\beta_{N,1}(F) & -\beta_{N,2}(F) & \cdots & \alpha_N(F) \end{bmatrix},$$

$\alpha_i(F) = \inf_{\boldsymbol{x} \in K} ||\nabla_i F_i||_2$, $\beta_{i,j}(F) = \sup_{\boldsymbol{x} \in K} ||\nabla_j F_i||_2$, $i \neq j$. *If* $\Upsilon(F)$ *is a **P-matrix**, that is, if all of its principal components have a positive determinant, then we say* F *satisfies the* P_Υ *condition.*

Theorem 1. *[88] If the agents' utility functions are individually concave, and* F *satisfies the* P_Υ

condition, then F is strongly monotone on K, and VI(K, F) has a unique solution. Moreover, BRD converges to the unique NE from an arbitrary initial state.

2.3 A Multi-Scale Game Model

Consider a conventional network (graphical) game with the set I of N agents situated on a network $G = (I, E)$, each with a utility function $u_i(x_i, \boldsymbol{x}_{I_i})$, with I_i the set of i's neighbors, I the full set of agents/nodes and E the set of edges connecting them.[1] Suppose that this network G exhibits the following structure and feature of the strategic dependence among agents: agents can be partitioned into a collection of groups $\{S_k\}$, where k is a group index, and an agent a_i in the kth group (i.e., $a_i \in S_k$) has a utility function that depends (i) on the strategies of its network neighbors in S_k, and (ii) *only on the aggregate strategies* of groups other than k (see, e.g., Fig. 2.1). Further, these groups may go on to form larger groups, whose aggregate strategies impact each other's agents, giving rise to a *multi-scale* structure of the network. This kind of structure is very natural in a myriad of situations. For example, members of criminal organizations take stock of individual behavior by members of their own organization, but their interactions with other organizations (criminal or otherwise) are perceived in group terms (e.g., how much another group has harmed theirs). A similar multi-level interaction structure exists in national or ethnic conflicts, organizational competition in a market place, and politics. Indeed, a persistent finding in network science is that networks exhibit a multi-scale interaction structure (i.e., communities, and hierarchies of communities) [32, 22].

We present a general model to capture such multi-scale structures. Formally, an L-level structure is given by a hierarchical graph structure $\{G^{(l)}\}$ for each level l, $1 \leq l \leq L$, where $G^{(l)} = (\{S_k^{(l)}\}_k, E^{(l)})$ represents the level-l structure. The first component, $\{S_k^{(l)}\}_k$ prescribes a partition, where agents in level $l-1$ form disjoint groups given by this partition; each group is viewed as an agent in level l, denoted as $a_k^{(l)}$. Notationally, while both $a_k^{(l)}$ and $S_k^{(l)}$ bear the superscript (l), the former refers to a level-l agent, while the latter is the group (of level-$(l-1)$ agents) that the former represents. The set of level-l agents is denoted by $I^{(l)}$ and their total number $N^{(l)}$. The second component, $E^{(l)}$, is a set of edges that connect level-l agents, encoding the dependence relationship among the groups they represent. This structure is anchored in level 1 (the lowest level), where sets $S_k^{(1)}$ are singletons, corresponding to agents a_k in the game, who constitute the set I.

To illustrate, the multi-scale structure shown in Fig. 2.1 is given by $G^{(1)} = G = (\{S_k^{(1)}\}_k = I, E^{(1)} = E)$, as well as how level-1 agents are grouped into level-2 agents, how level-2 agents are further grouped into level-3 agents, and the edges connecting these groups at each level.

It should be obvious that the above multi-scale representation of a graphical game is a gener-

[1] The edges are generally weighted, resulting in a weighted adjacency matrix on which the utility depends.

alization of a conventional graphical game, as any such game essentially corresponds to a $L = 1$ multi-scale representation. On the other hand, not all conventional graphical games have a meaningful $L > 1$ multi-scale representation (with non-singleton groups of level-1 agents); this is because our assumption that an agent's utility only depends on the *aggregate* decisions by groups other than the one they belong to implies certain properties of the dependence structure. For the remainder of this chapter, we will proceed with a given multi-scale structure defined above, while in Appendix A.7 we outline a set of conditions on a graphical game G that allows us to represent it in a (non-trivial) multi-scale fashion.

Since the resulting multi-scale network is strictly hierarchical, we can define a *direct supervisor* of agent $a_i^{(l)}$ in level-l to be the agent $a_k^{(l+1)}$ corresponding to the level-$(l+1)$ group k that the former belongs to. Similarly, two agents who belong in the same level-l group k are (level-l) *group mates*. Finally, note that any level-1 agent a_i belongs to exactly one group in each level l. We index a level-l group to which a_i belongs by k_{il}.

In order to capture the agent dependence on aggregate actions, we define an *aggregation function* $\sigma_k^{(l)}$ for each level-l group k that maps individual actions of group members to $\mathbb{R}$ (a *group strategy*). Specifically, consider a level-l group $S_k^{(l)}$ with level-$(l-1)$ agents in this group playing a strategy profile $\boldsymbol{x}_{S_k^{(l)}}$. The (scalar) group strategy, which is also the strategy for the corresponding level-$(l+1)$ agent, is determined by the aggregation function,

$$x_k^{(l)} = \sigma_k^{(l)}(\boldsymbol{x}_{S_k^{(l)}}). \tag{2.5}$$

A natural example of this is linear (e.g., agents respond to total levels of violence by other criminal organizations): $\sigma_k^{(l)}(\boldsymbol{x}_{S_k^{(l)}}) = \sum_{i \in S_k^{(l)}} x_i^{(l)}$.

The L-level structure above is captured strategically by introducing structure into the utility functions of agents. Let $I_{k_{il}}$ denote the set of neighbors of level-l group k to which level-1 agent a_i belongs; i.e., this is the set of level-l groups that interact with agent a_i's group. This level-1 agent's utility function can be decomposed as follows:

$$u_i(x_i, \boldsymbol{x}_{-i}) = \sum_{l=1}^{L} u_{k_{il}}^{(l)}\left(x_{k_{il}}^{(l)}, \boldsymbol{x}_{I_{k_{il}}}^{(l)}\right). \tag{2.6}$$

In this definition, the level-l strategies $x_k^{(l)}$ are implicitly functions of the level-1 strategies of agents that comprise the group, per a recursive application of Eqn. (2.5). Consequently, the utility is an additive function of the hierarchy of group-level components for increasingly (with l) abstract groups of agents. Note that conventional network games are a special case with only a single level ($L = 1$).

To illustrate, if we consider just two levels (a collection of individuals and groups to which they directly belong), the utility function of each agent a_i is a sum of two components:

$$u_i(x_i, \boldsymbol{x}_{-i}) = u^{(1)}_{k_{i1}}\left(x^{(1)}_{k_{i1}}, \boldsymbol{x}^{(1)}_{I_{k_{i1}}}\right) + u^{(2)}_{k_{i2}}\left(x^{(2)}_{k_{i2}}, \boldsymbol{x}^{(2)}_{I_{k_{i2}}}\right).$$

In the first component, $x^{(1)}_{k_{i1}} = x_i$, since level-1 groups correspond to individual agents, whereas $\boldsymbol{x}^{(1)}_{I_{k_{i1}}}$ is the strategy profile of i's neighbors *belonging to the same group as* i, given by $E^{(1)}$. The second utility component now depends only on the aggregate strategy $x^{(2)}_{k_{i2}}$ of the group to which i belongs, as well as the aggregate strategies of the groups with which i's group interacts, given by $E^{(2)}$.

2.4 Algorithms and Analysis

Consider the BRD algorithm (formalized in Algorithm 1) in which we iteratively select an agent who plays a best response to the strategy of the rest from the previous iteration.

Algorithm 1: BRD Algorithm

```
Initialize the game, t = 0, x_i(0) = (x_0)_i, i = 1, ..., N;
while not converged do
    for i = 1:N do
        x_i(t+1) = BR_i(x_{-i}(t), u_i)
    t <- t + 1
```

The conventional BRD algorithm operates on the "flattened" utility function which evaluates utilities explicitly as functions of the strategies played by all agents $a_i \in I$. Our goal henceforth is to develop algorithms that take advantage of the special multi-scale structure and enable significantly better scalability than standard BRD, while preserving the convergence properties of BRD.

2.4.1 Taking Advantage of Multi-Scale Utility Representation

The simplest way to take advantage of the multi-scale representation is to directly leverage the structure of the utility function in computing best responses. Specifically, the multi-scale utility function is more compact than one that explicitly accounts for the strategies of all neighbors of i (which includes *all* of the players in groups other than the one i belongs to). This typically results in a direct computational benefit to computing a best response. For example, in a game with a linear best response, this can result in an exponential reduction in the number of linear operations.

The resulting algorithm, *Multi-Scale Best-Response Dynamics (MS-BRD)*, which takes advantage of our utility representation is formalized as Algorithm 2. The main difference from BRD is that it explicitly uses the multi-scale utility representation: in each iteration, it updates the aggregated

Algorithm 2: Multi-Scale BRD (MS-BRD)

```
Initialize the game, t = 0, x_i^(1)(0) = (x_0)_i, i = 1, ..., N
for l = 2:L do
    for k = 1:N^(l) do
        x_k^(l)(0) = σ_k^(l)(x_{S_k^(l)}(0));
while not converged do
    for i = 1:N (Level-1) do
        x_i^(1)(t+1) = BR_i(x_{-i}^(1)(t), u_i)
    for l = 2:L do
        for k = 1:N^(l) do
            x_k^(l)(t+1) = σ_k^(l)(x_{S_k^(l)}(t+1));
    t ← t+1;
```

strategies at all levels for the groups to which the most recent best-responding agent belongs. Since MS-BRD simply performs operations identical to BRD but efficiently, its convergence is guaranteed under the same conditions (see Theorem 1). Next, we present iterative algorithms for computing NE that take further advantage of the multi-scale structure, and study their convergence.

2.4.2 Taking Advantage of Multi-Scale Strategic Dependence Structure

In order to take full advantage of the multi-scale game structure, we now aim to develop algorithms that treat groups explicitly as agents, with the idea that iterative interactions among these can significantly speed up convergence. Of course, in our model groups are not actual agents in the game: utility functions are only defined for agents in level 1. However, note that we already have well-defined group strategies – these are just the aggregations of agent strategies at the level immediately below, per the aggregation function (2.5). Moreover, we have natural utilities for groups as well: we can use the corresponding group-level component of the utility of any agent in the group (note that these are identical for all group members in Eqn. (2.6)). However, using these as group utilities will in fact not work: since ultimately the game is only among the agents in level 1, equilibria of all of the games at more abstract levels *must be consistent with equilibrium strategies in level 1*. On the other hand, we need to enforce consistency only between neighboring levels, since that fully captures the across-level interdependence induced by the aggregation function. Therefore, we define the following *pseudo-utility functions* for agents at levels other than 1, with agent k in level l corresponding to a subset of agents from level $l-1$:

$$\hat{u}_k^{(l)} = u_k^{(l)}\left(x_k^{(l)}, \boldsymbol{x}_{I_k}^{(l)}\right) - L_k^{(l,l-1)}\left(x_k^{(l)}, \sigma_k^{(l)}(\boldsymbol{x}_{S_k^{(l)}})\right) - L_k^{(l,l+1)}\left(\sigma_k^{(l+1)}(\boldsymbol{x}_{S_k^{(l+1)}}), x_k^{(l+1)}\right). \quad (2.7)$$

The first term is the level-l component of the utility of any level-1 agent in group k. The second and third terms model the inter-level inconsistency loss that penalizes a level-l agent $a_k^{(l)}$, where $L_k^{(l,l+1)}$ and $L_i^{(l,l-1)}$ penalize its inconsistency with the level-$(l+1)$ and level-$(l-1)$ entities respectively. In general, $L_k^{(l,l+1)}$ is a different function from $L_k^{(l+1,l)}$; we elaborate on this further below.

The central idea behind the second algorithm we propose is simple: in addition to iterating best response steps at level 1, we now interleave them with best response steps taken by agents at higher levels, which we can since strategies and utilities of these pseudo-agents are well defined. This algorithm is similar to the augmented Lagrangian method in optimization theory, where penalty terms are added to relax an equality constraint and turn the problem into one with separable operators. We can decompose this type of problem into smaller subproblems and solve the subproblems sequentially using the alternating direction method (ADM) [98, 8]. The games at adjacent levels are coupled through the equality constraints on their action profiles given by Eqn (2.5), and the penalty functions are updated before starting a new iteration. The full algorithm, which we call *Separated Hierarchical BRD (SH-BRD)*, is provided in Algorithm (3).

The penalty updating rule in iteration t of Algorithm (3) is:

1. For $l = 2, \ldots, L, i = 1, \ldots, N^{(l)}$

$$L_i^{(l,l-1)}\left(x_i^{(l)}, \sigma_i^{(l)}(\boldsymbol{x}_{S_i^{(l)}}(t+1))\right) = h_i^{(l)}\left[x_i^{(l)} - \sigma_i^{(l)}(\boldsymbol{x}_{S_i^{(l)}}(t+1)) + \lambda_i^{(l)}(t)\right]^2. \tag{2.8}$$

2. For $l = 1, \ldots, L-1; i = 1, \ldots, N^{(l)}$, where $a_i^{(l)} \in S_k^{(l+1)}$

$$L_k^{(l,l+1)}\left(\sigma_k^{(l+1)}(\boldsymbol{x}_{S_k^{(l+1)}}), x_k^{(l+1)}(t)\right) = h_k^{(l+1)}\left[\sigma_k^{(l+1)}(\boldsymbol{x}_{S_k^{(l+1)}}) - x_k^{(l+1)}(t) - \lambda_k^{(l+1)}(t)\right]^2. \tag{2.9}$$

3. For $l = 2, \ldots, L, i = 1, \ldots, N^{(l)}$

$$\lambda_i^{(l)}(t+1) = \lambda_i^{(l)}(t) - h_i^{(l)}\left[\sigma_i^{(l)}(\boldsymbol{x}_{S_i^{(l)}}(t+1)) - x_i^{(l)}(t+1)\right]. \tag{2.10}$$

When updating, all other variables are treated as fixed, and $\boldsymbol{\lambda}^{(l)}(0)$, $h_i^{(l)} > 0$ are chosen arbitrarily.

Unlike MS-BRD, the convergence of the SH-BRD algorithm is non-trivial. To prove it, we exploit a connection between this algorithm and Structured Variational Inequalities (SVI) with separable operators [38, 40, 8]. To formally state the convergence result, we need to make several explicit assumptions.

Algorithm 3: Separated Hierarchical BRD (SH-BRD)

Initialize the game, $t = 0, x_i^{(1)}(0) = (\boldsymbol{x}_0)_i, i = 1, \ldots, N^{(0)}$
for *l = 2:L* **do**
  for *k = 1:*$N^{(l)}$ **do**
    $\boldsymbol{x}_k^{(l)}(0) = \sigma_k^{(l)}(\boldsymbol{x}_{S_k^{(l)}}(0));$
while *not converged* **do**
  for *l = 1:L* **do**
    for *i = 1:*$N^{(l)}$ *(l to* $l-1$ *Penalty Update, if* $l > 1$*)* **do**
      Update $L_i^{(l,l-1)}$
    for *i = 1:*$N^{(l)}$ *(l to* $l+1$ *Penalty Update, if* $l < L$*)* **do**
      Update $L_k^{(l,l+1)}$, where $a_i^{(l)} \in S_k^{(l+1)}$
    for *i = 1:*$N^{(l)}$ *(Best Response)* **do**
      $x_i^{(l)}(t+1) = BR_i\left(\sigma_i^{(l)}(\boldsymbol{x}_{S_i^{(l)}}(t+1)), \boldsymbol{x}_{I_i}^{(l)}(t), x_k^{(l+1)}(t), \hat{u}_i^{(l)}\right)$
  $t \leftarrow t+1;$

Assumption 1. *The functions* $u_k^{(l)}, \forall l = 1, \ldots, L, \forall k = 1, \ldots, N^{(l-1)}$ *are twice continuously differentiable in* $x_k^{(l)}, \boldsymbol{x}_{I_k}^{(l)}$.

Assumption 2. $-\nabla_{x_i^{(l)}} u_i^{(l)}$ *are monotone* $\forall l = 1, \ldots, L, \forall i = 1, \ldots, N^{(l-1)}$. *The solution set of* $\nabla_{x_i^{(l)}} u_i^{(l)} = 0, \forall l = 1, \ldots, L, \forall i = 1, \ldots, N^{(l-1)}$ *is nonempty, with solutions in the interior of the action spaces.*

Let $F^{(l)}$ be defined as in Equation (2.3) for each level-l pseudo-utility.

Assumption 3. $F^{(l)}$ *satisfy the* P_Υ *condition.*

Note that these assumptions directly generalize the conditions required for the convergence of BRD to our multi-scale pseudo-utilities. The following theorem formally states that SH-BRD converges to a NE for *2-level games*.

Theorem 2. *Suppose* $L = 2$. *If Assumptions 1 and 3 hold,* SH-BRD *converges to a NE, which is unique.*

The full proof of this theorem, which makes use of the connection between SH-BRD and SVI, is provided in the appendix. The central issue, however, is that there are no established convergence guarantees for ADM-based algorithms for SVI with 3 or more separable operators. Alternative algorithms for SVI can extend to the case of 3 operators using parallel operator updates with regularization terms, but no approaches exist that can handle more than 3 operators [38]. We thus propose an algorithm for iteratively solving multi-scale games that uses the general idea from SH-BRD, but packs all levels into two *meta-levels*. The two meta-levels each have to be comprised of consecutive levels. For example, if we have 5 levels, we can have $\{1, 2, 3\}$ and $\{4, 5\}$ combinations, but not $\{1, 2, 4\}$ and $\{3, 5\}$. Upon grouping levels together to obtain a meta-game with only two meta-levels, we can apply what amounts to a 2-level version of the SH-BRD. This yields an algorithm, which we call *Hybrid Hierarchical BRD (HH-BRD)*, that now provably converges to a NE for an arbitrary number of levels L given assumptions 1-3.

As presenting the general version of HH-BRD involves cumbersome notation, we illustrate the idea by presenting it for a 4-level game (Algorithm 4). The fully general version is deferred to the Supplement. In this example, the objectives of the meta-levels are defined as

$$\begin{aligned}
\hat{u}_i^{(sl_1)} &= u_i^{(1)} + u_{k_{i2}}^{(2)} - L_{k_{i3}}^{(sl_1, sl_2)}\left(\sigma_{k_{i3}}^{(3)}(\boldsymbol{x}_{S_{k_{i3}}^{(3)}}), x_{k_{i3}}^{(3)}\right), \\
\hat{u}_{k_{i3}}^{(sl_2)} &= u_{k_{i3}}^{(3)} + u_{k_{i4}}^{(4)} - L_{k_{i3}}^{(sl_2, sl_1)}\left(x_{k_{i3}}^{(3)}, \sigma_{k_{i3}}^{(3)}(\boldsymbol{x}_{S_{k_{i3}}^{(3)}})\right).
\end{aligned}$$

Algorithm 4: Hybrid Hierarchical BRD

Initialize the game, $t = 0, x_i^{(1)}(0) = (\boldsymbol{x}_0)_i, i = 1, \ldots, N^{(0)}$
for *l = 2:4* **do**
  for *k = 1:*$N^{(l)}$ **do**
    $\boldsymbol{x}_k^{(l)}(0) = \sigma_k^{(l)}(\boldsymbol{x}_{S_k^{(l)}}(0))$;
while *not converged* **do**
  for *k = 1:*$N^{(3)}$ *(Meta-Level-1 Penalty Update)* **do**
    Update $L_k^{(sl_1, sl_2)}$
  for $i = 1 : N^{(1)}$ *(Level-1)* **do**
    $x_i^{(1)}(t+1) = BR_i\left(\boldsymbol{x}_{I_i}^{(1)}(t), \boldsymbol{x}_{I_{k_{i2}}}^{(2)}(t), x_{k_{i3}}^{(3)}(t), \hat{u}_i^{(sl_1)}\right)$
  for *j = 1:*$N^{(2)}$ *(Level-2)* **do**
    $\boldsymbol{x}_j^{(2)}(t+1) = \sigma_j^{(2)}(\boldsymbol{x}_{S_j^{(2)}}(t+1))$
  for *k = 1:*$N^{(3)}$ *(Meta-Level-2 Penalty Update)* **do**
    Update $L_k^{(sl_2, sl_1)}$
  for $k = 1 : N^{(3)}$ *(Level-3)* **do**

$$x_k^{(3)}(t+1) = BR_i\left(\sigma_k^{(3)}(\boldsymbol{x}_{S_k^{(3)}}(t+1)), \boldsymbol{x}_{I_k}^{(3)}(t), \boldsymbol{x}_{-p}^{(4)}(t), \hat{u}_k^{(sl_2)}\right), (a_k^{(3)} \in S_p^{(4)})$$

  for *p = 1:*$N^{(4)}$ *(Level-4)* **do**
    $\boldsymbol{x}_p^{(4)}(t+1) = \sigma_p^{(4)}(\boldsymbol{x}_{S_p^{(4)}}(t+1))$
  $t \leftarrow t + 1$;

Theorem 3. *Suppose Assumptions 1-3 hold Then* HH-BRD *finds the unique NE.*

Proof Sketch. We first "flatten" the game within each meta-level to obtain an effective 2-level game. We then use Theorem 2 to show this 2-level game converges to the unique NE of the game under SH-BRD. Finally, we prove that SH-BRD and HH-BRD have the same trajectory given the same initialization, thus establishing the convergence for HH-BRD. Please see Appendix A.4 for full proof. □

HH-BRD combines the advantages of both MS-BRD and SH-BRD: not only does it exploit the sparsity embedded in the network topology, but it also avoids the convergence problem of SH-BRD when the number of levels is higher than three. Indeed, there is a known challenge in the related work on structured variational inequalities that convergence is difficult when we involve three or more operators [38], which we leverage for our convergence results, with operators mapping to levels in our multi-scale game representation. One may be concerned that HH-BRD pseudocode appears to involve greater complexity (and more steps) than SH-BRD. However, this does not imply greater algorithmic complexity, but is rather due to our elaboration of the steps within each super level. Indeed, as our experiments below demonstrate, the superior theoretical convergence of HH-BRD also translates into a concrete computational advantage of this algorithm.

2.5 Numerical Results and Analysis

In this section, we numerically compare the three algorithms introduced in Section 2.4, as well as the conventional BRD. We only consider settings that satisfy Assumptions 1-3; consequently, we focus on the comparison of computational costs. We use two measures of computational cost: floating-point operations (FLOPs) in the case of games with a linear best response (a typical measure for such settings), and CPU time for the rest. All experiments were performed on a machine with A 6-core 2.60/4.50 GHz CPU with hyperthreaded cores, 12MB Cache, and 16GB RAM.

Games with a Linear Best Response (GLBRs) GLBRs [10, 19, 68] feature utility functions such that an agent's best response is a linear function of its neighbors' actions. This includes quadratic utilities of the form

$$u_i(x_i, x_{I_i}) = a_i + b_i x_i + \left(\sum_{j\in I_i} g_{ij} x_j\right) x_i - c_i x_i^2, \tag{2.11}$$

since an agent's best response is:

$$BR_i(x_{I_i}, u_i) = \frac{\sum_{j\in I_i} g_{ij} x_j}{2c_i} - b_i.$$

We consider a 2-level GLBR and compare three algorithms: BRD (baseline), MS-BRD, and HS-BRD (note that in 2-level games, HH-BRD is identical to HS-BRD, and we thus don't include it here). We construct random 2-level games with utility functions based on Equation (2.11). Specifically, we generalize this utility so that Equation (2.11) represents only the level-1 portion, $u_i^{(1)}$, and let the level-2 utilities be

$$u_k^{(2)}(x_k, \boldsymbol{x}_{I_k}) = x_k^{(2)} \sum_{p \neq k} v_{kp} x_p^{(2)}$$

for each group k. At every level, the existence of a link between two agents follows the Bernoulli distribution where $P_{exist} = 0.1$. If a link exists, we then generate a parameter for it. The parameters of the utility functions are sampled uniformly in $[0, 1]$ without requiring symmetry. Please refer to Appendix A.5 and A.5.1 for further details. Results comparing BRD, MS-BRD, and SH-BRD are shown in Table 2.1. We observe dramatic improvement in the scalability of using MS-BRD compared to conventional BRD. This improvement stems from the representational advantage provided by multi-scale games compared to conventional graphical games (since without the multi-scale representation, we have to use the standard version of BRD for equilibrium computation). We see further improvement going from MS-BRD to SH-BRD which makes algorithmic use of the multi-scale representation.

Size	**BRD**	**MS-BRD**	**SH-BRD**
30^2	$(2.51\pm0.18)\times10^6$	$(1.03\pm0.07)\times10^5$	$(\mathbf{9.81\pm0.81})\times10^4$
50^2	$(2.53\pm0.18)\times10^7$	$(5.33\pm0.04)\times10^5$	$(\mathbf{4.35\pm0.07})\times10^5$
100^2	$(4.46\pm0.32)\times10^8$	$(4.36\pm0.31)\times10^6$	$(\mathbf{3.56\pm0.29})\times10^6$
200^2	$(6.73\pm0.58)\times10^9$	$(3.48\pm0.29)\times10^7$	$(\mathbf{2.79\pm0.21})\times10^7$
500^2	$(2.84\pm0.21)\times10^{11}$	$(5.69\pm0.41)\times10^8$	$(\mathbf{4.04\pm0.29})\times10^8$

Table 2.1: Convergence and Complexity (FLOPs) Comparison with Linear Best Response under Multiple Initializations.

Games with a Non-Linear Best Response Next, we study the performance of the proposed algorithms in 2- and 3-level games, with the same number of groups in each level (we systematically vary the number of groups). Since SH-BRD and HH-BRD are identical in 2-level games, the latter is only used in 3-level games. All results are averaged over 30 generated sample games. The non-linear best response fits a much broader class of utility functions than the linear best response. The best responses generally don't have closed-form representations. In this case, we can't use linear equations to find the best response and instead have to apply gradient-based methods. In our instances, the utility with non-linear best responses is generated by adding an exponential cost term to the utility function used in GLBRs. Please refer to Appendix A.5 and A.5.2 for further details.

Size	BRD	MS-BRD	SH-BRD
30^2	1.50±0.05	1.02±0.02	**0.54±0.01**
50^2	26.70±0.36	3.70±0.14	**1.81±0.04**
100^2	1512±9	23.81±0.69	**12.10±0.13**
200^2	> 18000	287.2±5.4	**133.6±2.5**
500^2	nan	5485±13	**2524±10**

Table 2.2: CPU Times on a Single Machine on 2-Level Games with General Best Response Functions; All Times are in Seconds.

Table 2.2 shows the CPU time comparison between all algorithms. The scalability improvements from our proposed algorithms are substantial, with orders of magnitude speedup in some cases (e.g., from $\sim$ 25 minutes for the BRD baseline, down to $\sim$ 12 seconds for SH-BRD for games with 10K agents). Furthermore, BRD fails to solve instances with 250K agents, which can be solved by SH-BRD in $\sim$ 42 min. Again, we separate here the representational advantage of multi-scale games, illustrated by MS-BRD, and the algorithmic advantage that comes from SH-BRD. Note that SH-BRD, which takes full advantage of the multi-scale structure, also exhibits significant improvement over MS-BRD, yielding a factor of 2-3 reduction in runtime.

Size	BRD	MS-BRD	SH-BRD
30^2	1.21±0.04	0.63± 0.01	**0.037±0.003**
50^2	23.88±0.16	1.99±0.04	**0.079±0.004**
100^2	1461±14	15.49±0.24	**0.304±0.006**
200^2	> 18000	192.0±1.2	**1.87±0.05**
500^2	nan	4258±56 s	**28.79±0.37**

Table 2.3: CPU Times on a Single Machine for 2-Level, Linear/nonlinear Best-response Games; All Times are in Seconds.

Our next set of experiments involves games in which level-1 utility has a linear best response, but level-2 utility has a non-linear best response. The results are shown in Table 2.3. We see an even bigger advantage of SH-BRD over the others: it is now typically orders of magnitude faster than even MS-BRD, which is itself an order of magnitude faster than BRD. For example, in games with 250K agents, in which BRD fails to return a solution, MS-BRD takes more than 1 hour to find a solution, whereas SH-BRD finds a solution under 30 seconds.

Size	BRD	MS-BRD	SH-BRD	HH-BRD
10^3	1.23±0.03	0.59±0.01	0.76±0.03	**0.43±0.02**
20^3	696.0±8.7	3.78±0.09	6.05±0.08	**3.35±0.09**
30^3	$>$ 18000	15.70±0.11	25.13±0.14	**13.39±0.11**
50^3	nan	68.59±0.75	138.8±1.1	**57.98±0.69**
100^3	nan	1126±6	2343±21	**877.1±11.5**

Table 2.4: CPU Times in Seconds on a Single Machine on 3-Level, General Best Response Games; All Times are in Seconds.

Finally, Table 2.4 presents the results of HH-BRD in games with > 2 levels compared to SH-BRD, which does not provably converge in such games. In this case, HH-BRD outperforms the other alternatives, with up to 22% improvement over MS-BRD; indeed, we find that SH-BRD is considerably worse even than MS-BRD.

2.6 Chapter Conclusions

We proposed a novel representation of network games that have a multi-scale network structure. These generalize network games, but with special structures that agent utilities are additive across the levels of hierarchy, with utility at each level depending only on the aggregate strategies of other groups. We present several iterative algorithms that make use of the multi-scale game structure, and show that they converge to a pure strategy Nash equilibrium under similar conditions as for best response dynamics in network games. Our experiments demonstrate that the proposed algorithms can yield orders of magnitude scalability improvement over conventional best response dynamics. Our multi-scale algorithms can reveal the extent to which one's group affiliation impacts one's strategic decision making, and how strategic interactions among groups impact strategic interactions among individuals.

Chapter 3

Structured Network Games: Leveraging Relational Information in Equilibrium Analysis

3.1 Introduction

As discussed in Chapter 2, the network game framework can be used to capture different forms of interdependencies between agents' decisions, such as allowing the action of an agent to be a strategic substitute or complement to that of its neighbors. While these problems have been extensively studied in prior works, see e.g., [3, 17, 53, 10, 68, 74, 2, 100, 63, 79, 73], the focus has been largely on network games with a single, generic underlying network governing the relationship among agents. These will be referred to as *unstructured* network games in this chapter; accordingly, their corresponding adjacency matrices will be referred to as *regular* adjacency matrices. The main focus of this line of work is in identifying sufficient conditions under which the Nash equilibrium (NE) of such a network game is unique and stable, see e.g., [79, 73]. Such conditions are important from a number of perspectives. First, many equilibrium computation algorithms require assumptions that guarantee the uniqueness of the NE. To ensure the convergence or effectiveness of such algorithms, verification of such assumptions is important. Moreover, knowledge about properties like the uniqueness and stability of an NE can help a (mechanism) designer in devising more suitable mechanisms and interventions for the system.

In contrast, this chapter focuses on network games with *structures* that arise naturally, which closely follows the idea in Chapter 2 and further extends to multi-relational networks. Specifically, we consider two (non-mutually exclusive) families of structures in this study: (1) In the first, the underlying network enjoys certain special graphical properties, a prime example being agents forming *local communities*, thereby creating sub-graphs that are more strongly dependent/connected within themselves; (2) In the second case, agents enjoy *multi-relational* interactions, whereby they are connected over multiple (parallel) networks, each of which governs the dependency relationship of a different action dimension in a high-dimensional action space. Our goal is to understand how the existence of these types of structures affects the resulting Nash equilibria analysis, and how to exploit such structures, when they exist, to provide characterizations of conditions for the existence, uniqueness, and stability of NEs in such network games. Similar to [3, 2, 100, 79, 73], we consider utility functions with nonlinear best-response functions and use the equivalent variational inequality (VI) representation to study the properties of the NE.

It turns out that such network structures can be used to our advantage in analyzing the NEs of these games, as well as in substantially reducing the computational requirement for equilibrium computation and for verifying the aforementioned conditions. This is done by adopting an *extended* or *generalized* form of adjacency matrix defined over the space of both agents and actions, and identifying *partitions* in this matrix as a result of the types of structures in the network and/or in the agents' dependent relationships described earlier.

Specifically, the computational effort in verifying the uniqueness and stability conditions derived using existing methods, also referred to as *unstructured conditions* in this chapter, entails the study of the properties of a game Jacobian (this is the Jacobian matrix of the operator in the VI problem yielding the NE). This matrix is in general asymmetric, making the condition verification a co-NP-complete problem; this means the computational complexity grows faster than polynomial time in the size of the game (total number of agents for an unstructured network game, or total number of agent-action pairs for a structured network game). In the special case of a symmetric game Jacobian, the verification complexity of such conditions still grows in polynomial time in the size of the game.

In contrast, taking advantage of the partition structures in a structured network game, we derive a new set of conditions, also referred to as *structured conditions*, that depend only on the partitions in the game Jacobian and the size of the partitions. Both the structured and the unstructured conditions rely on matrices derived from utility functions and adjacency matrices. When the unstructured derived matrix is asymmetric (resp. symmetric), the structured condition reduces the verification complexity multiple degrees in exponential time (resp. polynomial time), where the degrees depend on the size of the network and the partition structures. Empirically, as we show in Section 3.7, the verification of the new condition takes only 2% of the CPU time needed to verify the unstructured conditions from existing literature and avoids memory overflow in large games.

Reducing the verification complexity is of great conceptual and practical interest: it allows us to obtain a high-level understanding of a game much faster and avoid computational resource bottlenecks, especially for large games, and enables early decisions. However, such computational efficiency gain does come at a cost, in that the structured conditions are stronger, i.e., they are sufficient conditions that imply the unstructured conditions. In other words, some games may satisfy the unstructured conditions (which guarantee the uniqueness and stability of the NE) but fail to satisfy the structured conditions. Our extensive numerical experiments show that this sufficiency gap is small in general, and the sample games that lead to such sufficiency gaps have concentrated features. We characterize such features and provide real-world interpretations.

In addition to the uniqueness and stability of the NE, we also introduce a new notion of centrality for structured network games, which can be viewed as a generalization of degree centrality. This new centrality measures the influence or importance of a partition in the network game. We show that this notion of centrality can help identify additional conditions for the uniqueness and stability of the NE.

Our main contributions are summarized as follows.

- We provide sufficient conditions for the existence, uniqueness, and stability of the Nash equilibrium in network games, by taking advantage of the partition structures which may arise due to, e.g., the existence of communities or multi-relational interactions.

- We show that these conditions are sufficient conditions that imply those obtained in previous works, but that they are computationally much easier to verify than their counterparts obtained using conventional methods without utilizing the partition structures.

- The partition structure further allows us to define a new centrality measure, which can be used to verify the uniqueness and stability of the NE in these games.

- We conduct numerical experiments that compare our new conditions with conditions in previous works in terms of their verification complexity and strengths, which shed light on when using these new conditions is advantageous.

The remainder of the chapter is organized as follows. We provide motivating examples for the study of structured networks in Section 3.2. In Section 3.3, we introduce our model of structured network games. We present our results on the existence and uniqueness of Nash equilibria in Section 3.4, followed by results on the stability of Nash equilibria in Section 3.5. We propose a generalized notion of degree centrality for this class of games in Section 3.6. We present and discuss our numerical experiment results in Section 3.7, and conclude in Section 3.8.

Related work: A common line of research in network games studied the effect of the network on the existence, uniqueness, and stability of the game equilibria (see [45] for a survey). In particular, unstructured network games with linear best-response functions have been studied in [10, 68, 74]. Bramoulle *et al.* [10] uncovered the importance of the lowest eigenvalue of the adjacency matrix of the network in determining the uniqueness and stability of the Nash equilibrium. Miura-Ko *et al.* [68] showed that if the adjacency matrix of the network is strictly diagonally dominant, then the Nash equilibrium is unique. Naghizadeh and Liu [74] identified necessary and sufficient conditions for the existence and uniqueness of Nash equilibria in network games with linear best-replies by establishing a connection to linear complementarity problems.

Unstructured network games with nonlinear best-response functions have been studied in [3, 2, 100, 63, 79, 73]. Allouch [3] introduces a sufficient condition for the uniqueness of Nash equilibrium called *network normality* which imposes lower and upper bounds on the derivative of Engel curves. Acemoglu *et al.* [2] consider a network game with idiosyncratic shocks and show that if the best response mapping is either a contraction with a Lipschitz constant smaller than one or a bounded non-expansive mapping, then the game has a unique Nash equilibrium. Zhou *et al.* [100] establish a connection between nonlinear complementary problems (NCP) and network games, and use existing results from the NCP literature to find sufficient conditions for the uniqueness of Nash equilibria. The works of [63, 79, 73] use the framework of variational inequalities to study network games with nonlinear best-replies. Naghizadeh and Liu [73] show that a sufficient condition for the uniqueness and stability of the Nash equilibrium can be determined by the lowest eigenvalue of matrices constructed based on the slope of the agents' interaction functions and the intensity of their interactions. Parise and Ozdaglar [79] identify an operator in the variational inequality problem which involves the derivative of the agent's cost with respect to its action, and show that various properties of this operator and its Jacobian will determine conditions for the existence, uniqueness, and stability of a Nash equilibrium.

3.2 Motivating Examples

We elaborate on the idea of structured networks through a running example consisting of a number of vendors/store owners (strategic agents) selling similar merchandise.

Communities due to stronger connectivity Community formation stems from situations where the strengths of agents' strategic interactions are (statistically) different within certain groups compared to those between these groups. One type of community structure, commonly studied and discovered using spectral analysis [77, 76], is characterized by groups with much stronger connectivity (higher density of connections or existence of edges, as well as higher edge weights on those edges) within themselves, and much weaker connectivity (lower density of edges and smaller edge weights) between them.

Consider the case of three store owners, with agents a_1 and a_2 in close proximity of each other and a_3 located far away. Further, consider the store owners' single action of selecting business hours. Since a_1 and a_2 offer similar goods, their individual decisions on business hours (from complete overlap to mutually exclusive) will have direct consequences on the other's business volume and goods sold, resulting in a stronger dependence relationship between the two. Their dependence on (or influence of) a_3 may be far weaker. This is illustrated in the left of Figure 3.1, where the stronger relationship between a_1 and a_2 is indicated by a thicker edge; here a_1 and a_2 form a group or community.

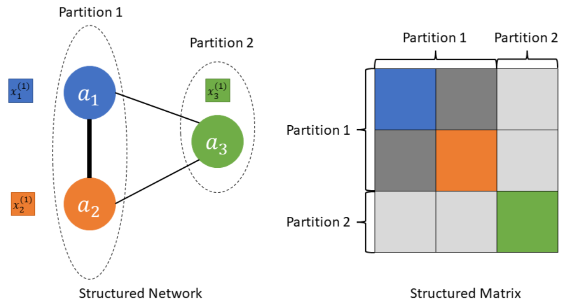


Figure 3.1: Network with Communities/groups: 3 Agents and 1 Action Dimension; a_1 and a_2 Form the First Group, a_3 is a Singleton Group.

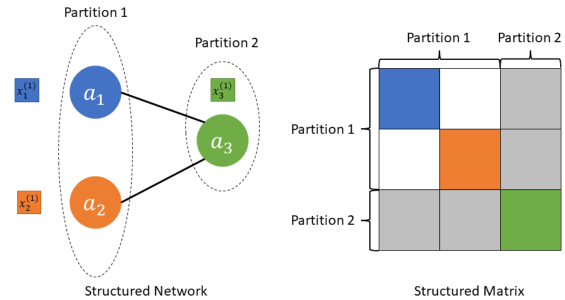


Figure 3.2: Bipartite Graph, where a_1 and a_2 are on One Side, and Exhibit the Same Type of Dependence on a_3 on the Other Side.

The recognition of groups gives rise to a "block" view in the structured adjacency matrix shown on the right-hand side of Figure 3.1. Given there is only a single action, here every row and column is associated with an agent, resulting in a 3×3 interaction matrix. The colors of the diagonal entries match the identity of the agents depicted in the network as well as their indices. The diagonal elements in the regular interaction matrix represent an agent's self influence, whereas the off-diagonal elements represent the agents' mutual influence. Similarly, the diagonal blocks in the structured matrix represent a partition's self influence, whereas the off-diagonal blocks represent the partitions' mutual influence. These quantities will be precisely defined in the next section.

Communities due to the similarity in function Communities can also result from a logical relationship. As an example, consider again the three store owners, where a_1 and a_2 carry completely orthogonal merchandise (e.g., a bakery vs. a hardware store) but both rely on a_3 to provide store security and vehicle rental as needed, and consider the single action of staffing levels. In this case, the dependency only exists between a_1, a_3, and between a_2, a_3, but not between a_1, a_2, resulting in a bipartite network shown in Figure 3.2. Yet in this case it is still appropriate to view a_1 and a_2 as belonging to the same group, because they each exhibit very similar dependence on another group.

Partition on action dimensions Our next example is more complex and introduces a high-dimensional action space. The general idea is that when actions are high-dimensional, the interaction/dependency relationships among agents can be different for different action dimensions, effectively resulting in multiple parallel networks superimposed on each other; this will also be referred to as a *multi-relational* network game.

Consider again the three store owners, each of which now has both physical and online sales. It is reasonable to expect that the decisions a seller makes about what goods to display in the window or offer for sampling may have more impact on other similar stores in its physical proximity (one set of agents), whereas its decisions on webpage design, layout, picture quality, and payment options may only impact its online competitors (another set of agents). In such multi-relational games, the action dimensions naturally create action-based structures on the network.

Suppose we capture the above with two action dimensions (in-store and online decisions), and further suppose in our case a_1 and a_2 are much more in direct competition in terms of physical-store sales due to their proximity, but that a_2 and a_3 are much more in direct competition in terms of their online sales due to high similarity in the goods they carry. This means that agents form different groups in different actions, as illustrated in Figure 3.3, where agents are represented by different colors and action dimensions are represented by different shades. In the first (solid color) action, a_1 and a_2 form a group whereas in the second (faded color) action, a_2 and a_3 form a group.

To capture the multiple action dimensions, we will define an *extended* adjacency/interaction matrix where each row and each column represent an agent-action dimension pair; the regular adjacency matrix is then a special case of this extended definition when the action dimension is 1. Under this definition, we have a 6×6 interaction matrix for this example. This is shown in Figure 3.4. Given such a matrix, partitions can emerge either by agents (similar sets of agents form the same group regardless of the action dimension) or by actions (interactions along different dimensions tend to be orthogonal), as is the case in this example and shown in Figure 3.4.

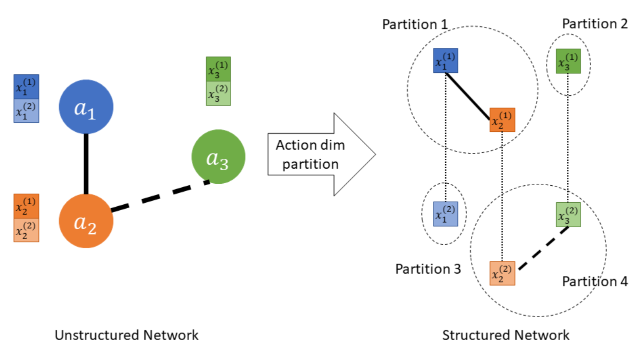


Figure 3.3: Action Dimension Partitioned Graph, with 3 Agents and 2 Actions. a_1 and a_2 are Closely Connected on Action Dimension 1, while a_2 and a_3 are Closely Connected on Action Dimension 2.

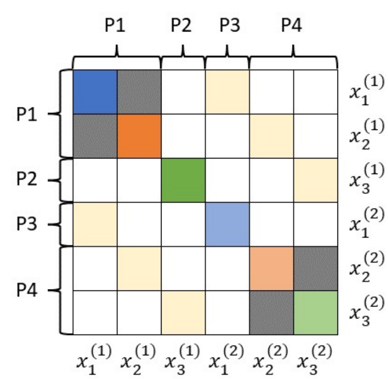


Figure 3.4: Action Dimension Partitioned Matrix for the Network in Figure 3.3.

Arbitrary partitions in the extended adjacency matrix We note that the idea of a partitioned extended adjacency matrix is not restricted to the above example scenarios. In principle, this can be done across both agents and action dimensions in arbitrary ways.

It is important to note that in our analysis, we will be working with the Jacobian (∇F) of a best-response operator (F), defined precisely in subsequent sections, rather than the extended adjacency matrix (G) itself. This is because the former captures both first-order and second-order information about the utility functions, which is not captured in the latter. In particular, the extended adjacency matrix does not reveal cross-action dependencies induced by the utility functions. We will, however, also show that there is a direct correspondence between a partition on the extended adjacency matrix and that on the Jacobian. For this reason, the use of the adjacency matrix in this section is for illustration purposes only, as it is much more intuitive and straightforward to visualize the described partition structures in an adjacency matrix.

Please refer to Appendix B.1 for concrete numbers used in Figure 1-4.

3.3 Model and Preliminaries

3.3.1 The Structured Network Game Model

We consider a structured network game among N agents $\mathcal{N} = \{a_1, \ldots, a_N\}$, each with K action dimensions. The multi-relational network is represented by a multi-relational graph with extended adjacency matrix G. The adjacency matrix on the k-th action dimension is denoted as $G^{(k)}$, and is a submatrix of G.

The edge weight $G_{ij}^{(k)} \in \mathbb{R}$ is a real number representing the strength of influence agent a_j has on agent a_i (or a_i's dependence on a_j) in the k-th action dimension.

We use $x_i^{(k)} \in \mathbb{R}$ to denote the k-th action of agent a_i, $\boldsymbol{x}_i = (x_i^{(k)})_{k=1}^K \in \mathbb{R}^K$ to denote the action vector of a_i, and $\boldsymbol{x}_i^{(-k)} = [x_i^{(1)}, \cdots, x_i^{(k-1)}, x_i^{(k+1)}, \cdots, x_i^{(K)}]^T$ to denote the action profile of a_i, excluding the k-th dimension. In addition, let $\boldsymbol{x}^{(k)} = (x_i^{(k)})_{i=1}^N \in \mathbb{R}^N$ be the action vector of all agents on the k-th action dimension, and $\boldsymbol{x}_{-i}$ denotes the action profile of all agents other than a_i.

Each agent a_i has an action constraint $\boldsymbol{x}_i \in Q_i = \prod_{k=1}^K Q_i^{(k)}$, where $Q_i^{(k)} := [0, B_i^{(k)}]$ such that $B_i^{(k)}$ captures the physical or financial constraints (budgets) on the k-th action dimension.

We consider games with utility functions consisting of an individual component and a network component:

$$u_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}, G) = \ d_i(\boldsymbol{x}_i) + f_i(\boldsymbol{x}_i, G^{(1)}\boldsymbol{x}^{(1)}, \ldots, G^{(K)}\boldsymbol{x}^{(K)}). \tag{3.1}$$

Here, $d_i(\cdot)$ is the individual component, which only depends on a_i's own actions; conventionally, it contains a standalone benefit and cost of taking action $\boldsymbol{x}_i$. The network component $f_i(\cdot)$ depends on not only the agent's own but also others' actions. Here, the network influence on the k-th action dimension is captured by $G^{(k)}\boldsymbol{x}^{(k)}$. Throughout the paper, we make the following assumption on the game.

Assumption 4. *The utility functions $u_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}, G)$ are concave in $\boldsymbol{x}_i$ and twice continuously differentiable in $\boldsymbol{x}_i$ and $\boldsymbol{x}_{-i}$, for all i.*

An example of this type of utility function is given below.

Example 1. *The multi-relational extension of linear-quadratic utility functions, studied in e.g., [30], has the following form,*

$$u_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}, G) = \boldsymbol{x}_i^T \boldsymbol{b}_i + \sum_{k=1}^{K} \sum_{j \neq i} x_i^{(k)} g_{ij}^{(k)} x_j^{(k)} - \frac{1}{2} \boldsymbol{x}_i^T C_i \boldsymbol{x}_i,$$

where $\boldsymbol{x}_i^T \boldsymbol{b}_i - \frac{1}{2}\boldsymbol{x}_i^T C_i \boldsymbol{x}_i$ is the individual component containing the standalone benefit term and the cost term, and $\sum_{k=1}^{K} \sum_{j \neq i} x_i^{(k)} g_{ij}^{(k)} x_j^{(k)}$ is the (multi-relational) network influence component. (We can further expand this utility function with cross-relational influence terms $\sum_{k=1}^{K} \sum_{j \neq i} x_i^{(k)} g_{ij}^{(k,l)} x_j^{(l)}$, where $G^{(k,l)}$ is a submatrix in G modeling the network influence from the l-th action to the k-th action.)

A Nash equilibrium of the described structured network game can be found as a fixed point of the agents' *best response* mappings. The best response of an agent in the network game is defined as the action an agent takes to maximize its own utility, given other agents' actions and the network topology. For our model, we denote the best response of agent a_i as

$$BR_i(\boldsymbol{x}_{-i}, G) := \arg \max_{\boldsymbol{x}_i \in Q_i} \; u_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}, G). \tag{3.2}$$

We also define an operator F_i as follows,

$$F_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}) = -\nabla_{\boldsymbol{x}_i} u_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}, G) \in \mathbb{R}^K. \tag{3.3}$$

Next, we introduce the Variational Inequality (VI) framework and its relation to Nash equilibria in network games.

3.3.2 The Variational Inequality (VI) Problem

Variational Inequalities (VIs) are a class of optimization problems with applications in game theory. In particular, the Nash equilibria of many games can be found as solutions to a corresponding VI problem [79, 88]. We state the VI problem formally below, using the set of notations introduced earlier so that the correspondence between the game model and the VI problem is clear.

Definition 2. *A variational inequality $VI(Q, F)$ consists of a set $Q \subseteq \mathbb{R}^N$ and a mapping $F : Q \to$*

$\mathbb{R}^N$*, and is the problem of finding a vector* $\boldsymbol{x}^* \in Q$ *such that,*

$$(\boldsymbol{x} - \boldsymbol{x}^*)^T F(\boldsymbol{x}^*) \geq 0, \; \forall \boldsymbol{x} \in Q. \tag{3.4}$$

Finding the Nash equilibrium of a structured network game is equivalent to solving a variational inequality problem $VI(Q, F(\boldsymbol{x}))$ with the appropriate choice of Q and F. An example choice of Q and F is: $Q = Q_1 \times Q_2 \times \ldots \times Q_N \subseteq \mathbb{R}^{NK}$, with Q_i being the action space of a_i, and $F(\boldsymbol{x}) = (F_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}))_{i=1}^N \in \mathbb{R}^{NK}$, with $F_i(\cdot)$ given in (3.3). Then, since finding an NE is the problem of finding a fixed point of the best response mappings in (3.2), it is equivalent to solving the VI problem given in (3.4) with the choice of F and Q stated above (see, e.g., [28, Proposition 1.4.2]). Intuitively, if the equilibrium $\boldsymbol{x}_i^*$ is an interior point of Q_i, then $F_i(\boldsymbol{x}_i^*, \boldsymbol{x}_{-i}^*) = \mathbf{0}$ if and only if $\boldsymbol{x}_i^*$ is a best response to $\boldsymbol{x}_{-i}^*$; if $\boldsymbol{x}_i^*$ is on the boundary, we can still show $(\boldsymbol{x}_i - {\boldsymbol{x}_i}^*)^T F_i(\boldsymbol{x}^*) \geq 0$ holds. Note also that the above definition holds for Q and F that are permutations of the preceding example choice, as long as the permutations are consistent.

3.3.3 Partitions

We next formally define partitions in a structured network game. As shown in Section 3.2, a partition is essentially a set of indices, where each index corresponds to an agent-action pair (which also corresponds to a column or row in the extended adjacency/interaction matrix G). Figures 3.1, 3.2, and 3.4 show how, by grouping certain indices together (rearranging rows and columns), block structures may emerge in G.

It turns out a block structure in G translates to a similar block structure in the Jacobian of the operator F, when F is a suitable permutation of the operator $(F_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}))_{i=1}^N$ consistent with the partitions.

As defined in (3.3), we can think of operator F_i as the best response direction vector of a_i, where each element in F_i corresponds to an agent-action component. Therefore, operator $F(\boldsymbol{x})$ is the global best response direction vector containing all agent-action components; these components can be arranged in an arbitrary order, and $(F_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}))_{i=1}^N$ corresponds to a common order.

While F captures important first-order derivative information of the utility functions with respect to the agent-action components, the Jacobian of F, denoted as $\triangledown F$ captures important second-order information of the utility functions and first-order information of the operator F. It is common, see e.g., [88, 73, 79], to use the properties of $\triangledown F$ to derive unstructured conditions for equilibrium analysis.

When $K = 1$, $\triangledown F \in \mathbb{R}^{N \times N}$, where the off-diagonal elements in $\triangledown F$ measure how an agent's action influences another agent's best response. For $K > 1$, $\triangledown F \in \mathbb{R}^{NK \times NK}$, and now the off-diagonal elements in $\triangledown F$ measure how an agent-action component influences the best response direction of another agent-action component.

There is a one-to-one mapping between the dimensions in $\triangledown F$ and G, since each dimension in $\triangledown F$ and G corresponds to an agent-action component, and thus any partition on G can equally apply to $\triangledown F$, as partitions are nothing more than separating the agent-action components into disjoint sets. Using the same 3-agent, 2-action example in Figures 3.3 and 3.4, while $(F_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}))_{i=1}^N$ orders the agent-action components as $x_1^{(1)}, x_1^{(2)}, x_2^{(1)}, x_2^{(2)}, x_3^{(1)}, x_3^{(2)}$, F orders them as $x_1^{(1)}, x_2^{(1)}, x_3^{(1)}, x_1^{(2)}, x_2^{(2)}, x_3^{(2)}$. Then, if we partition the network as shown in Figure 3.3, we can partition $\triangledown F$ the same way, as illustrated in Figure 3.4.

We note that the partition based on (agent) group structure and the partition based on action dimensions are special cases. For the remainder of the paper, we will discuss network games with general, arbitrary partition structures. Specifically, for an arbitrary structure, we can partition all the $N \times K$ agent-action components (and their corresponding indices) into an arbitrary number (M) of disjoint sets, denoted by $\mathcal{P}_1, \ldots, \mathcal{P}_M$. Accordingly, we will also denote by $N_i = |\mathcal{P}_i|$ the size of the partitions, where $\sum_{i=1}^M N_i = NK$.

3.4 Existence and Uniqueness

We now identify conditions for the existence and uniqueness of Nash equilibrium based on the VI formulation of the game.

3.4.1 Existence of NE

We first state the conditions under which a Nash equilibrium exists. From [88], we have the following theorem that guarantees the existence of NE:

Theorem 4. *([88, Theorem 3]) If F is continuous on Q, and Q is nonempty, compact and convex, then $VI(Q, F)$ has a nonempty and compact solution set.*

This is because in our problem, $Q = Q_1 \times Q_2 \times \ldots \times Q_N$, with $Q_i = [0, B_i^{(1)}] \times \ldots \times [0, B_i^{(K)}]$, so that together with Assumption 4, the conditions of Theorem 4 are satisfied for the network game defined in Section 3.3.

3.4.2 Uniqueness of NE

We next introduce sufficient conditions under which the network game defined in Section 3.3.1 has a unique Nash equilibrium. We begin by introducing the following definitions.

Definition 3. *P-Matrix: A matrix $A \in \mathbb{R}^{N \times N}$ is a P-matrix if every principal minor of A has a positive determinant.*

Definition 4. *A mapping $F : Q \mapsto \mathbb{R}^{NK}$, where $Q \subseteq \mathbb{R}^{NK}$ is nonempty, compact and convex, and F is continuously differentiable on Q, is strongly monotone if there exists a constant $c > 0$ such that*

$$(\boldsymbol{x} - \boldsymbol{y})^T (F(\boldsymbol{x}) - F(\boldsymbol{y})) \geq c\|\boldsymbol{x} - \boldsymbol{y}\|_2^2,\ \forall \boldsymbol{x}, \boldsymbol{y} \in Q\ . \tag{3.5}$$

Further, $F = (F_1, F_2, \ldots, F_N)$ is a uniform block P-function w.r.t. the (agent-level) partition $Q = Q_1 \times Q_2 \times \cdots \times Q_N$ if there exists a constant $b > 0$ such that

$$\max_{i \in \mathbb{N}[1,M]} (\boldsymbol{x}_i - \boldsymbol{y}_i)^T [F_i(\boldsymbol{x}) - F_i(\boldsymbol{y})] \geq b\|\boldsymbol{x} - \boldsymbol{y}\|_2^2,\ \forall \boldsymbol{x}, \boldsymbol{y} \in Q\ . \tag{3.6}$$

By setting $b = c/N$, it is easy to see that strong monotonicity is a sufficient condition for the uniform block P-condition. Parise and Ozdaglar [79] show that if $F(\boldsymbol{x})$ is a uniform block P-function, then $VI(Q, F)$ has a unique solution, and the Nash equilibrium of the network game corresponding to $VI(Q, F)$ is unique.

Unfortunately, it is computationally costly to verify these conditions for the function $F(\boldsymbol{x})$, since checking whether a square matrix is a P-matrix is co-NP-complete [24], and typically, the complexity grows (faster than polynomial time) in the size of the matrix. Below, we show that it is possible to take advantage of the block structure when it is present, and identify conditions for the uniqueness of Nash equilibrium that are of lower computational complexity to verify.

To do so, we define a structured matrix Υ^S and its components, the *internal* and *external* impact levels of partitions, as follows:

Definition 5. *We say a partition $\mathcal{P}_i$ receives internal impact, α_i^S, and external impact, β_{ij}^S, defined as follows:*

$$\begin{aligned} \alpha_i^S &= \inf_{\boldsymbol{x} \in Q} \|\nabla_i F_i(\boldsymbol{x})\|_2,\ \forall i \in \mathbb{N}[1, M] \\ \beta_{ij}^S &= \sup_{\boldsymbol{x} \in Q} \|\nabla_j F_i(\boldsymbol{x})\|_2,\ \forall i, j \in \mathbb{N}[1, M],\ i \neq j\ , \end{aligned} \tag{3.7}$$

where $\nabla_j F_i(\boldsymbol{x}) \in \mathbb{R}^{N_i \times N_j}$ is a matrix with k, l-th entry $\frac{\partial F_i^{(k)}(\boldsymbol{x})}{\partial x_j^{(l)}}$. The structured matrix Υ^S for the network game is defined accordingly as:

$$\Upsilon^S = \begin{bmatrix} \alpha_1^S & -\beta_{12}^S & \ldots & -\beta_{1M}^S \\ -\beta_{21}^S & \alpha_2^S & \ldots & -\beta_{2M}^S \\ \vdots & \vdots & \ddots & \vdots \\ -\beta_{M1}^S & -\beta_{M2}^S & \ldots & \alpha_M^S \end{bmatrix}. \tag{3.8}$$

We note that these impact measures are only defined between partitions and not agents. Accordingly, in this definition, the subscripts are indices of partitions instead of agents.

To motivate the above definition, it helps to understand the matrix Υ^S in the context of what is typically used in prior works for checking the uniqueness of the Nash equilibrium. If we ignore the structure in the network and simply view each agent-action pair as a singleton partition, then using an existing methodology (such as in [79, 73, 88]) will give us the Jacobian Υ^U, which is a $NK \times NK$ matrix. Specifically, Υ^U is given by

$$\Upsilon^U = \begin{bmatrix} \alpha_1^U & -\beta_{12}^U & \dots & -\beta_{1,NK}^U \\ -\beta_{21}^U & \alpha_2^U & \dots & -\beta_{2,NK}^U \\ \vdots & \vdots & \ddots & \vdots \\ -\beta_{NK,1}^U & -\beta_{NK,2}^U & \dots & \alpha_{NK,NK}^U \end{bmatrix}, \tag{3.9}$$

and contains the following elements:

$$\begin{aligned} \alpha_k^U &= \inf_{\boldsymbol{x} \in Q} |\nabla_k F_k(\boldsymbol{x})|, \ \forall k \in \mathbb{N}[1, NK] \\ \beta_{kl}^U &= \sup_{\boldsymbol{x} \in Q} |\nabla_l F_k(\boldsymbol{x})|, \ \forall k, l \in \mathbb{N}[1, NK], \ k \neq l. \end{aligned} \tag{3.10}$$

To clarify, these will be referred to as the *component-level* internal and external impact, respectively. Suppose we rearrange the rows and columns of Υ^U in the following way: group together rows whose action dimensions are in $\mathcal{P}_i$, $i = 1, \cdots, M$; group together columns whose agents are in $\mathcal{P}_j$, $j = 1, \cdots, M$.

We can now view this rearranged Υ^U in blocks/submatrices denoted by $\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_j}$, and there are $M \times M$ blocks. The matrix Υ^S is essentially a condensed version of this rearranged matrix, summarizing or abstracting each block into a single quantity as defined in Definition 5: α_i^S for the diagonal block $\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_i}$ and β_{ij}^S for the off-diagonal block $\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_j}$.

This abstraction aims at capturing the dependence relationship between partitions rather than between individual agent-action pairs. In particular, β_{ij}^S represents the largest influence level of partition $\mathcal{P}_j$ on partition $\mathcal{P}_i$, and α_i^S represents the minimum influence level of $\mathcal{P}_i$ on itself. This is formally established in the following lemma.

Lemma 1. *We have* $\|\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_j}\|_2 \geq \beta_{ij}^S$ *and* $\|\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_i}\|_2 \leq \alpha_i^S$.

For detailed proof, see Appendix B.2, and please refer to Section 3.4.3 for additional interpretation.

In what follows, we show that the matrix Υ^S can be used to provide sufficient conditions on the uniqueness of NE in network games with partition structures. Such an abstraction takes advantage of the partition structure and reduces the dimension of the matrix used to check conditions for the uniqueness from $NK \times NK$ to a single $M \times M$ matrix and M matrices of size $|\mathcal{P}_k| \times |\mathcal{P}_k|, k \in [1, M]$. This greatly reduces the complexity of condition verification, since as mentioned earlier, P-matrix verification is co-NP-complete [24], and the complexity typically grows (faster than polynomial) in the size of the matrix. For example, in a special case where the matrix is symmetric, P-matrix verification is equivalent to examining its positive definiteness and the complexity becomes $O(N^3)$ for an $N \times N$ matrix. If there are $N = 20$ agents on the network with $K = 5$ action dimensions, then the verification complexity of unstructured conditions is $O(10^6)$; if we partition the game by action dimensions, the complexity of the structured conditions is $5 \cdot 8 \cdot O(10^3) + O(5^3)$, much lower than $O(10^6)$. This complexity gap will only increase if the matrix is asymmetric.

However, we also note that the strengths of the conditions obtained from the structured network and the unstructured network are not equivalent. Specifically, we will later show that the conditions obtained from the structured network are stronger (sufficient conditions to) their counterparts in the unstructured network. Numerical results presented in Section 3.7 also highlight the gap between these two sets of conditions.

The following result identifies a condition for the uniqueness of the Nash equilibrium in games with the structured network.

Theorem 5. *If the following two conditions are satisfied,*

1. Υ^S *is a P-matrix, and*

2. $\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_i}, \forall i$ *are P-matrices (the diagonal blocks of* Υ^U *are P-matrices),*

then the network game has a unique Nash equilibrium.

Moreover, when both conditions are satisfied, then Υ^U *is also a P-matrix; i.e., the uniqueness conditions (1 & 2) on the structured network are also a sufficient condition for the uniqueness of the Nash equilibrium in the underlying unstructured network (i.e., one that ignores the partitioned structures).*

Proof. (Sketch)

- First, we show that if Υ^S is a P-matrix, then $F(\boldsymbol{x})$ is a uniform block P-function with respect to the partitions $\mathcal{P}_1, \ldots, \mathcal{P}_M$.

- By [79], if $F(\boldsymbol{x})$ is a uniform block P-function with respect to the partitions, then $VI(Q, F)$ has a unique solution which implies that the network game has a unique equilibrium.

- Finally, using Lemma 1, we show that if $F(\boldsymbol{x})$ is a uniform block P-function with respect to the partitions and $\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_i}, \forall i$ are P-matrices, then in the unstructured network, if we treat each agent-action component as a singleton partition, the counterpart of $F(\boldsymbol{x})$ is also a uniform block P-function with respect to the singleton partition.

For detailed proof, see Appendix B.3. □

An interpretation of the above result is as follows. As noted earlier, β_{ij}^S represents the external impact of partition $\mathcal{P}_j$ on partition $\mathcal{P}_i$, while α_i^S is the internal impact of $\mathcal{P}_i$. Typically, when β_{ij}^S has a relatively small value compared to α_i^S, then Υ^S is a P-matrix. Moreover, when Υ^S is (row or column) diagonally dominant, then Υ^S is a P-matrix [93]. In these types of networks, partitions' action profiles have a bounded influence on each other. On the other hand, if at least one partition's action profile has an out-sized effect on other partitions, then its decision can shift the state of the network substantially and result in possibly multiple equilibria.

Remark 1. *It is worth mentioning that the structured (resp. unstructured) network condition verification is of a much lower complexity if Υ^S (resp. Υ^U) is symmetric. By [69], a symmetric matrix is a P-matrix if and only if it is positive definite, which means that instead of checking the determinant for every principal minor, we only need to do an eigendecomposition. This reduces the complexity from solving a co-NP-complete problem down to polynomial time.*

We next present two corollaries of Theorem 5 which provide alternative ways for verifying the uniqueness of NE in structured network games. We define

$$\Gamma^S = \begin{bmatrix} 0 & -\beta_{12}^S/\alpha_1^S & \dots & -\beta_{1M}^S/\alpha_1^S \\ -\beta_{21}^S/\alpha_2^S & 0 & \dots & -\beta_{2M}^S/\alpha_2^S \\ \vdots & \vdots & \ddots & \vdots \\ -\beta_{M1}^S/\alpha_M^S & -\beta_{M2}^S/\alpha_M^S & \dots & 0 \end{bmatrix}, \tag{3.11}$$

where β_{ij}^S and α_i^S are as in Definition 5. Similar to before, by treating each agent as a singleton partition, we can obtain Γ^S's unstructured counterpart, $\Gamma^U \in \mathbb{R}^{NK\times NK}$. We also denote the spectral radius of Γ^S as $\rho(\Gamma^S)$. By [90], if $\rho(\Gamma^S) < 1$, then Υ^S is a P-matrix. Therefore, we have the following Corollary of Theorem 5.

Corollary 1. *Assume the following two conditions hold simultaneously*

1. $\rho(\Gamma^S) < 1$,
2. $\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_i}, \forall i$ *are P-matrices.*

Then, the network game has a unique Nash equilibrium. Moreover, under these conditions, Υ^U is a P-matrix.

Since Υ^S is a P-matrix under the conditions of Corollary 1, the sufficiency is a direct result of Theorem 5.

Our second corollary is on the special case where Γ^S is a symmetric matrix. As mentioned above, by [69], we know that a symmetric matrix is a P-matrix if and only if it is positive definite. Therefore, we have the following.

Corollary 2. *Assume Γ^S is a symmetric matrix. Denote the eigenvalues of Γ^S by $\lambda_1(\Gamma^S) \leq \lambda_2(\Gamma^S) \leq \cdots \leq \lambda_M(\Gamma^S)$. Then, the network game has a unique Nash equilibrium if the following two conditions hold simultaneously,*

1. *Eigenvalues of Γ^S are larger than -1, i.e., $\lambda_1(\Gamma^S) > -1$,*

2. *$\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_i}, \forall i$ are P-matrices.*

Moreover, under these conditions, Υ^U is a P-matrix.

The proof is given in Appendix B.4.

These corollaries provide two alternative ways to check for the uniqueness of the Nash equilibrium. In terms of complexity, both finding the spectral radius and the eigendecomposition are of complexity $O(N^3)$ for an $N \times N$ matrix; these corollaries' conditions are therefore computationally easier to verify than the co-NP-complete problem. However, the trade-off is that the conditions in Corollary 1 are stronger than those of Theorem 5, while Corollary 2 can only be used given a symmetric Γ^S (resp. Γ^U) on structured (resp. unstructured) networks. It is also worth mentioning that even when Υ^S (resp. Υ^U) are asymmetric, we can still have symmetric Γ^S (resp. Γ^U) matrices, and thus, Corollary 2 could still provide a computationally lighter alternative verification in these cases.

Lastly, we again note that $\Gamma^U \in \mathbb{R}^{NK \times NK}$ could be formed by treating each agent as a singleton partition. When we take this viewpoint, Corollary 2 reduces to Proposition 3 of [73]. On the other hand, by using the partition structure, Γ^S is an $M \times M$ matrix, and the conditions in Corollary 2 are computationally easier to verify as compared to those in Proposition 3 of [73]. Specifically, checking the eigenvalues of a matrix requires performing eigendecomposition over it. To elaborate on the comparison, suppose $N_i = \overline{N}$ for all $\mathcal{P}_i$ (all partitions have the same size). Then, using the unstructured network, the complexity of the eigendecomposition on Υ^U or Γ^U is $O(K^3N^3)$ while using the partition structure, the complexity on Υ^S or Γ^S is reduced to $O(M^3)$. Of course, using the partition structure, we have to compute the α_i^S and β_{ij}^S values as well, which is of complexity $O(M^2\overline{N}^3)$. Altogether, the complexity of checking whether Υ^S is a P-matrix under the conditions in the above corollary is $O(M^3+M^2\overline{N}^3)$, which can be much lower than $O(K^3N^3)$ in the unstructured case.

3.4.3 Sufficiency Gaps on the Uniqueness Conditions

To close this section, we elaborate on the difference between using and not using information about partition structures in checking for the uniqueness of Nash equilibria.

The first thing to note is that, as mentioned earlier, the set of sufficient conditions derived when accounting for partition structures are generally stronger than their counterparts derived without using information about the structure: as Theorem 5, Corollary 1, and Corollary 2 indicate, if a structured network satisfies these uniqueness conditions, then it also satisfies the corresponding unstructured uniqueness conditions, but the opposite is in general not true. This means that there is a sufficiency gap between the conditions obtained from the structured and unstructured networks. The most important reason behind this sufficiency gap has to do with the way partition structures are abstracted. There are two forms of abstractions made during the creation of the Υ^S matrix; both are for partitions and each summarizes the component (agent-action) level internal and external impact, respectively:

1. The internal impact α_i^S of partition $\mathcal{P}_i$, is an abstraction of the component-level internal impact α_k^U for components (and corresponding indices) in $\mathcal{P}_i$, and component level external impact β_{kl}^U between different indices in $\mathcal{P}_i$;

2. The external impact β_{ij}^S of $\mathcal{P}_j$ on $\mathcal{P}_i$, is an abstraction of component-level external impact of indices in $\mathcal{P}_j$ to indices in $\mathcal{P}_i$.

These types of abstractions inevitably introduce gaps in the sufficiency conditions. As mentioned in Lemma 1, the value of α_i^S is lower-bounded by $\|\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_i}\|$ and highly depends on the agents with component-level internal impact in $\mathcal{P}_i$. Meanwhile, the β_{ij}^S value is upper bounded by $\|\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_j}\|$, and highly depends on the strongest component-level external impact from one index in $\mathcal{P}_j$ to another in $\mathcal{P}_i$. The significance of α_i^S in verifying conditions for the uniqueness of the Nash equilibria is akin to the observation "a chain is as strong as its weakest link"; we refer to this as the "weakest link effect". Similarly, the significance of the β_{ij}^S values is referred to as the "strongest link effect".

Recall an earlier observation in Theorem 5 that when each agent has stronger component-level internal impact than component-level external impact, the NE is unique. Similarly, in terms of structure, when each partition has stronger internal impact than external impact, the NE is unique and the conditions in Theorem 5 and Corollaries 1, 2 are sufficient to guarantee such impact differentials.

In some games, the indices k with weak component-level internal impact α_k^U also have weak component-level external impact β_{kl}^U while the indices l with strong component-level internal impact α_l^U have similarly strong component-level external impact β_{lk}^U. While we may be able to guarantee the uniqueness of an NE using the unstructured network, we may not be able to do so using structures when a partition contains both types of (strong and weak) indices. This is because the abstraction becomes inaccurate as the partition's internal impact α_i^S is weak and the external impact β_{ij}^S strong; thus the conditions obtained from the structured network may fail to guarantee the uniqueness of NE. The following example highlights these observations.

Example 2. *Consider a 4-agent, single-action dimension, 2-partition game where agents a_1, a_2 form $\mathcal{P}_1$ and agents a_3, a_4 form $\mathcal{P}_2$, with utility functions: $u_1(\boldsymbol{x}) = x_1(x_2 + 5x_3) - 5x_1^2, u_2(\boldsymbol{x}) = x_2(x_1 + x_4) - 2x_2^2, u_3(\boldsymbol{x}) = x_3(x_4 + 5x_1) - 5x_3^2, u_4(\boldsymbol{x}) = x_4(x_3 + x_2) - 2x_4^2$. Then, we have*

$$\Upsilon^U = \begin{bmatrix} 10 & -1 & -5 & 0 \\ -1 & 4 & 0 & -1 \\ -5 & 0 & 10 & -1 \\ 0 & -1 & -1 & 4 \end{bmatrix} \succ \mathbf{0}, \ \ \Upsilon^S = \begin{bmatrix} 3.838 & -5 \\ -5 & 3.838 \end{bmatrix} \prec \mathbf{0},$$

which means that the unstructured condition guarantees the uniqueness of the NE in this game, yet the game fails to satisfy the sufficient structured condition. This is an example when the internal impact of a partition is weak but the external impact between partitions is strong.

We end this section by summarizing the types of games and partition structures that result in small sufficiency gaps between the two sets of sufficient conditions. The sufficiency gap is small if partition members have similar component-level internal impact; or if members have similar connections to agents in other partitions and have similar component-level external impact level; or if the member with the weakest component-level internal impact also has the strongest component-level external impact. We present numerical experiments in Section 3.7 to further elaborate on these comparisons.

3.5 Stability

We next examine conditions for the stability of the Nash equilibrium in these games. When small changes occur to the underlying model parameters, a new Nash equilibrium may result. Intuitively, if the new Nash equilibrium is close enough to the original one, then we say the original Nash equilibrium is stable.

Formally, we generalize our utility functions in Eqn (3.1) to the family of parameterized functions $u_i(\boldsymbol{x}_i, \boldsymbol{x}_{-i}, \boldsymbol{p}_i)$, where $\boldsymbol{p}_i = [p_i^{(1)}, \cdots, p_i^{(K)}] \in \mathbb{R}^K$ is a vector valued *perturbation parameter* or *shock* on a_i, and $\boldsymbol{p} = [\boldsymbol{p}_1, \cdots \boldsymbol{p}_N] \in \mathbb{R}^{NK}$ denotes the vector of all perturbations/shocks. Moreover, let $\boldsymbol{x}^*(\boldsymbol{p})$ be the action profile at the Nash equilibrium of the game under perturbation vector $\boldsymbol{p}$ and $\boldsymbol{x}^*$ be the Nash equilibrium of the unperturbed game ($\boldsymbol{x}^* := \boldsymbol{x}^*(\mathbf{0})$).

We denote a ball of radius $r > 0$ centered at $\boldsymbol{x} \in \mathbb{R}^N$ by $B(\boldsymbol{x}, r) := \{\boldsymbol{y} \in \mathbb{R}^N : ||\boldsymbol{x} - \boldsymbol{y}||_2 < r\}$.

Definition 6. *([55]) A Nash equilibrium $\boldsymbol{x}^*$ is stable if $\exists r > 0, d > 0$ such that $\forall \boldsymbol{p} \in B(\boldsymbol{0}, r)$, the Nash equilibrium $\boldsymbol{x}^*(\boldsymbol{p})$ exists and satisfies*

$$||\boldsymbol{x}^*(\boldsymbol{p}) - \boldsymbol{x}^*||_2 \leq d||H(\boldsymbol{x}^*(\boldsymbol{p}), \boldsymbol{p}) - H(\boldsymbol{x}^*(\boldsymbol{p}), \boldsymbol{0})||_2 \ ,$$

where $H(\boldsymbol{x}, \boldsymbol{p}) = (H_i(\boldsymbol{x}, \boldsymbol{p}))_{i=1}^M$ with

$$H_i(\boldsymbol{x}, \boldsymbol{p}) = \boldsymbol{x}_i - BR_i(\boldsymbol{x}_{-i}, G, \boldsymbol{p}) \ .$$

Definition 6 states that if an NE $\boldsymbol{x}^*$ is stable, the Nash equilibrium of the perturbed game ($\boldsymbol{x}^*(\boldsymbol{p})$) remains close to the Nash equilibrium of the unperturbed game ($\boldsymbol{x}^*(\boldsymbol{0})$).

3.5.1 Stability Condition Without Network Structure

In order to determine whether a Nash equilibrium $\boldsymbol{x}^*$ is stable, [73] proposed dividing the agents' action indices into three disjoint sets based on $\boldsymbol{x}^*$:

$$\begin{aligned} A(\boldsymbol{x}^*) :=& \{j = \psi(i,k) \mid x_i^{(k)*} > 0, \ x_i^{(k)*} = \tilde{BR}_i^{(k)}(\boldsymbol{x}_{-i}^*, u_i)\}, \\ I(\boldsymbol{x}^*) :=& \{j = \psi(i,k) \mid x_i^{(k)*} = 0, \ x_i^{(k)*} > \tilde{BR}_i^{(k)}(\boldsymbol{x}_{-i}^*, u_i)\}, \\ B(\boldsymbol{x}^*) :=& \{j = \psi(i,k) \mid x_i^{(k)*} = 0, \ x_i^{(k)*} = \tilde{BR}_i^{(k)}(\boldsymbol{x}_{-i}^*, u_i)\}, \end{aligned}$$

where $\tilde{BR}_i(\boldsymbol{x}_{-i}^*, u_i)$ is the *unbounded* best response and can take negative values, $\tilde{BR}_i^{(k)}$ denotes the k-th action dimension unbounded best response, and $\psi : \{1, \ldots, N\} \times \{1, \ldots, K\} \mapsto \{1, \ldots, KN\}$ maps the $N \times K$ agent-action indices pair to the KN indices in the unstructured operator F. $A(\boldsymbol{x}^*)$ is referred to as the set of active indices, $I(\boldsymbol{x}^*)$ the set of strictly inactive indices, and $B(\boldsymbol{x}^*)$ the set of borderline inactive indices. Intuitively, with a small parametric perturbation $\boldsymbol{p}$, agent action indices in $A(\boldsymbol{x}^*)$ remain active ($x_i^{(k)*}(\boldsymbol{p}) > 0$) and agent action indices in $I(\boldsymbol{x}^*)$ remain inactive ($x_i^{(k)*}(\boldsymbol{p}) = 0$), while agent action indices in $B(\boldsymbol{x}^*)$ can transform from inactive to active ($x_i^{(k)*}(\boldsymbol{p}) > x_i^{(k)*}(\boldsymbol{0}) = 0$).

Under these definitions, [73] established the following sufficient condition for the solution to $VI(Q, F)$ to be stable in the sense of Definition 6.

Theorem 6. *([73]) Consider the matrix*

$$\nabla_{A,B} F_{A,B}(\boldsymbol{x}^*) = \begin{bmatrix} \nabla_A F_A(\boldsymbol{x}^*) & \nabla_B F_A(\boldsymbol{x}^*) \\ \nabla_A F_B(\boldsymbol{x}^*) & \nabla_B F_B(\boldsymbol{x}^*) \end{bmatrix} \tag{3.12}$$

where $\nabla_{S_1} F_{S_2}(\boldsymbol{x}^*)$ is a sub-matrix of $\nabla F(\boldsymbol{x}^*)$ with rows and columns corresponding to the agent *action indices in sets S_1 and S_2 (not necessarily groups), respectively, and $\nabla_{A,B}F_{A,B}(\boldsymbol{x}^*)$ is generated by selecting rows and columns corresponding to $A \cup B$ from the game Jacobian $\nabla F(\boldsymbol{x}^*)$. If $\nabla_{A,B}F_{A,B}(\boldsymbol{x}^*)$ is positive definite on Q, then the solution $\boldsymbol{x}^*$ to $VI(Q, F)$ is stable.*

Below we provide a condition for stability which is easier to verify as compared to that in Theorem 6 by taking the partition structure into account.

3.5.2 Stability Condition with Partition Structure

Similar to [73], we divide *partitions* into active, strictly inactive, and borderline inactive sets. Specifically: (1) a partition is active if at least one agent action index in that partition is active at NE $\boldsymbol{x}^*$; (2) if all agent action indices in a partition are strictly inactive, then the partition is strictly inactive; (3) if all agent action indices of a partition are inactive and at least one of them is borderline inactive, then the partition is considered as a borderline inactive partition. Formally, we have,

$$\begin{aligned} A_S(\boldsymbol{x}^*) &:= \{\mathcal{P}_i \mid \boldsymbol{x}^*_{\mathcal{P}_i} \neq \mathbf{0}\}, \\ I_S(\boldsymbol{x}^*) &:= \{\mathcal{P}_i \mid \boldsymbol{x}^*_{\mathcal{P}_i} = \mathbf{0}, \boldsymbol{x}^*_{\mathcal{P}_i} > \tilde{BR}_{\mathcal{P}_i}(\boldsymbol{x}^*)\}, \\ B_S(\boldsymbol{x}^*) &:= \{\mathcal{P}_i \mid \boldsymbol{x}^*_{\mathcal{P}_i} = \mathbf{0}\} - I_S(\boldsymbol{x}^*), \end{aligned} \tag{3.13}$$

where $A_S(\boldsymbol{x}^*)$, $I_S(\boldsymbol{x}^*)$, $B_S(\boldsymbol{x}^*)$ denote the set of active, strictly inactive and borderline inactive partitions, respectively. We use $\boldsymbol{x}^*_{\mathcal{P}_i}$, $\tilde{BR}_{\mathcal{P}_i}(\boldsymbol{x}^*)$ denote the vectors by choosing all indices in partition $\mathcal{P}_i$ from the NE $\boldsymbol{x}^*$, and the unbounded best response $\tilde{BR}(\boldsymbol{x}^*)$.

Theorem 7. *Consider a Nash equilibrium $\boldsymbol{x}^*$ of the network game. Re-index all partitions in $A_S(\boldsymbol{x}^*) \cup B_S(\boldsymbol{x}^*)$ with indices $1, 2, \cdots, Z$, $Z = |A_S(\boldsymbol{x}^*)| + |B_S(\boldsymbol{x}^*)|$. Then, define*

$$G^S(\boldsymbol{x}^*) = \begin{bmatrix} \theta^S_1(\boldsymbol{x}^*) & -\delta^S_{12}(\boldsymbol{x}^*) & \dots & -\delta^S_{1Z}(\boldsymbol{x}^*) \\ -\delta^S_{21}(\boldsymbol{x}^*) & \theta^S_2(\boldsymbol{x}^*) & \dots & -\delta^S_{2Z}(\boldsymbol{x}^*) \\ \vdots & \vdots & \ddots & \vdots \\ -\delta^S_{Z1}(\boldsymbol{x}^*) & -\delta^S_{Z2}(\boldsymbol{x}^*) & \dots & \theta^S_Z(\boldsymbol{x}^*) \end{bmatrix}$$

where $\delta^S_{ij}(\boldsymbol{x}^*) = ||\nabla_j F_i(\boldsymbol{x}^*)||_2$, $\theta^S_i(\boldsymbol{x}^*) = ||\nabla_i F_i(\boldsymbol{x}^*)||_2$. If the following conditions hold simultaneously:

1. $G^S(\boldsymbol{x}^*) \succ \mathbf{0}$,
2. $\nabla_i F_i(\boldsymbol{x}^*) \succ \mathbf{0}, \forall \mathcal{P}_i \in A_S \bigcup B_S$,

then $\boldsymbol{x}^*$ is stable. Moreover, these conditions are sufficient for $\nabla_{A,B}F_{A,B}(\boldsymbol{x}^*) \succ \mathbf{0}$, i.e., the condition for stability on an unstructured network (Theorem 6) holds under these conditions.

For detailed proof, see Appendix B.5.

Intuitively, the matrix $G^S(\boldsymbol{x}^*)$ captures the mutual influence between active and borderline inactive partitions at the current Nash equilibrium profile. The borderline inactive partitions can turn into active partitions under parametric perturbations. When such flips are significant, large fluctuations can appear in the network, which can be further amplified through rebounds and reflections. In this case, new equilibria may not exist, and even if they do, they may be far away from the original equilibrium. However, when $G^S(\boldsymbol{x}^*) \succ \mathbf{0}$ holds, the maximum impacts of flipping partitions from (borderline) inactive to active are bounded, and therefore the current Nash equilibrium remains stable.

In terms of the complexity of verifying these conditions, note that G^S is a $Z \times Z$ matrix. Similar to the comparison shown in Section 3.4, if we denote $Y = |A(\boldsymbol{x}^*)|+|B(\boldsymbol{x}^*)|$, then the computational complexity of condition verification in Proposition 7 vs. Theorem 6 are $O(Z^3 + Z^2\overline{N}^3)$ vs. $O(K^3Y^3)$, where $\overline{N}$ is the average group size. Therefore, since $Z < KY$ ($Z\overline{N} \approx KY$), the computational complexity of condition verification in Proposition 7 is lower than that of Theorem 6.

We conclude this section with a condition on Υ^S leading to stable Nash equilibrium.

Theorem 8. *Assume Υ^S is symmetric. Then if the following two conditions hold simultaneously:*

1. $\Upsilon^S \succ \mathbf{0}$,
2. $\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_i}, \forall i$ *are P-matrices,*

the network game's Nash equilibrium is unique and stable.

For detailed proof, see Appendix B.6.

3.6 Centrality

In network games, notions of node centrality are used to measure the influence of individual nodes on network-level outcomes. Degree centrality is one of the centrality metrics which has gained attention in the literature [13, 78]. In a directed graph, two different measures of degree centrality are considered for each node: in-degree centrality, which is a count of edges directed to a given node, and out-degree centrality, which is the number of outward edges from the given node. In this section, we propose a generalization of the degree centrality measure for disjoint partitions.

Recall that we capture the influence of a partition using the Jacobian matrix $\nabla F(\boldsymbol{x})$. Matrix $\nabla_j F_i(\boldsymbol{x})$ measures the sensitivity of agents in partition i to the action profile of agents in partition j. Accordingly, we define our generalized centrality measure as follows.

Definition 7. *Generalized Degree Centrality (GDC): Following Definition 5, denote* $\beta_{ij}^S = \sup_{\boldsymbol{x}\in Q} ||\nabla_j F_i(\boldsymbol{x})||_2$*, and* $\alpha_i^S = \inf_{\boldsymbol{x}\in Q} ||\nabla_i F_i(\boldsymbol{x})||_2$*. The generalized degree centralities for partition* $\mathcal{P}_i$ *are given by:*

$$D_i^{in} = \sum_{j:j\neq i} \frac{\beta_{ij}^S}{\alpha_i^S}, \quad D_i^{out} = \sum_{j:j\neq i} \frac{\beta_{ji}^S}{\alpha_j^S}, \ \forall i, j \in \mathbb{N}[1, M] \ .$$

Moreover, the maximum GDCs are defined as follows:

$$D_{max}^{in} = \max_{i\in\mathbb{N}[1,M]} D_i^{in}, \quad D_{max}^{out} = \max_{i\in\mathbb{N}[1,M]} D_i^{out} \ .$$

The above definition can be interpreted as follows: out-degree centrality measures the influence of a given partition $\mathcal{P}_i$ on the network based on three factors, (1) connectivity, or the number of links directed outward from $\mathcal{P}_i$, (2) the internal impact of the target partitions that receive impact from $\mathcal{P}_i$, and (3) the external impact $\mathcal{P}_i$ has for every target group. In Definition 7, D_i^{out} captures these factor through the summation of $\frac{\beta_{ji}^S}{\alpha_j^S}$. D_i^{in} can be interpreted similarly.

In addition to capturing importance due to their roles in the network, each partition can be endowed with certain *exogenous* (non-network related) importance (not to be confused with external impact from another partition). This will result in an extended centrality measure. The following is a generalization of the extended centrality measure defined in [84].

Definition 8. *Generalized Extended Degree Centrality (GEDC): Let* $\boldsymbol{e} \in \mathbb{R}_{>0}^M$ *denote the vector of external importance, where* $(\boldsymbol{e})_i = e_i > 0$ *is* $\mathcal{P}_i$*'s external importance. The generalized extended degree centralities for* $\mathcal{P}_i$ *are given by*

$$D_i^{in}(\boldsymbol{e}) = \sum_{j:j\neq i} \frac{\beta_{ij}^S}{\alpha_i^S}\frac{e_j}{e_i}, \ D_i^{out}(\boldsymbol{e}) = \sum_{j:j\neq i} \frac{\beta_{ji}^S}{\alpha_j^S}\frac{e_i}{e_j}, \forall i, j \in \mathbb{N}[1, M]$$

and the maximum GEDCs are defined as

$$D_{max}^{in}(\boldsymbol{e}) = \max_{i\in\mathbb{N}[1,M]} D_i^{in}(\boldsymbol{e}), \ D_{max}^{out}(\boldsymbol{e}) = \max_{i\in\mathbb{N}[1,M]} D_i^{out}(\boldsymbol{e}) \ .$$

When $\boldsymbol{e} = \alpha\mathbf{1}, \alpha > 0$, Definitions 7 and 8 are equivalent. We now show the connection between our centrality measure and the uniqueness of the Nash equilibrium.

Theorem 9. *If the following two conditions hold simultaneously:*

1. $\Upsilon_{\mathcal{P}_i,\mathcal{P}_i}^U, \forall i$ *are P-matrices,*

2. *there exists* $\boldsymbol{e} \succ \mathbf{0}$ *such that* $D_{max}^{in}(\boldsymbol{e}) < 1$*, or* $D_{max}^{in}(\boldsymbol{e}) < 1$*,*

then the Nash equilibrium is unique. If in addition Υ^S is symmetric, the Nash equilibrium is unique and stable.

For a detailed proof, see Appendix B.7.

Theorem 9 implies that if either the in-degree or out-degree GEDCs are bounded, then the Nash equilibrium is unique. On the other hand, if neither the indegree nor outdegree is bounded, then at least one partition has an outsized effect on the network. This partition's decision can change the state of the network significantly, resulting in possibly multiple equilibria.

Theorem 9 is similar to Proposition 7 in [88], but differs in the following aspect. In our work, the β_{ij}^S represent the influence of partitions on each other, while β_{ij} in [88] represents the component-level influence of agents on each other. Moreover, when both conditions in Theorem 9 hold, $\Upsilon^S \succ 0$; this then becomes a special case of the condition in Corollary 2 (where Υ^S is symmetric).

Moreover, Theorem 9 shows that if Υ^S is symmetric and the degree centralities of the partitions are bounded, then the unique Nash equilibrium is also stable. Intuitively, this is because if the degree centrality of a given partition is not bounded, the partition has a considerable impact on the network, and the Nash equilibrium may not be stable as a small perturbation affecting this partition can influence the network dramatically.

3.7 Numerical Results

In this section, we present numerical results that are closely related to the analytical results shown in previous sections. We first show the computational complexity and sufficiency gaps between the structured and unstructured conditions, using single-action dimension (single-relational) network games; the results are applicable to multi-relational network games as our analysis has shown. We then use a multi-relational network game example to demonstrate an interesting setting where some agents are important in one type of relationship but not in another. We will visualize this phenomenon with the corresponding centrality measures defined in Section 3.6.

3.7.1 Procedure for Game Instance Generation

Our results on computational complexity gaps and the sufficiency gaps are obtained from a large number of game instances randomly generated using the following procedure. A game is generated for a specified size (number of agents N, number of action dimensions $K = 1$, number of partitions M, and size of each partition N_i) and using utility functions $u_i(\boldsymbol{x}, G) = x_i(b_i + \sum_j g_{ij}x_j) - \frac{c_i}{2}x_i^2$. The rest of the game is given by the interaction matrix G, and the vectors $\boldsymbol{b} = (b_i)_{i=1}^N$ and $\boldsymbol{c} = (c_i)_{i=1}^N$. In generating a random G, the diagonal elements are set to 0 without loss of generality. Each partition is associated with consecutive agent indices and thus the diagonal blocks represent each partition and the off-diagonal blocks represent cross-partition interdependencies. The off-diagonal elements in a diagonal block are generated using a Bernoulli distribution with parameter P^{in}_{exist}, the probability for a connection (non-zero element) to exist between any pair of agent-action indices. If such a connection exists, its value (strength of the connection), g_{ij}, is drawn from a uniform distribution on the interval $[S^{in}_{low}, S^{in}_{high}]$. We generate g_{ij} and g_{ji} independently. These elements also determine the off-diagonal elements in the diagonal blocks of Υ^U and the diagonal elements α^S_i in the Υ^S matrix.

The off-diagonal blocks of G are generated using the same approach, with parameters P^{out}_{exist} and $[S^{out}_{low}, S^{out}_{high}]$. These determine the connection frequencies and strengths between groups. These elements also determine the off-diagonal blocks of the Υ^U matrix as well as the off-diagonal elements β^S_{ij} in the Υ^S matrix.

In subsequent numerical results, a *dense* matrix G refers to both $P^{in}_{exist} = 1$ and $P^{in}_{exist} = 1$, i.e., the entries in Υ^U and Υ^S are non-zero with probability 1.

The vector of individual cost $\boldsymbol{c}$ is generated by first choosing a fixed mean value $\bar{c} = \frac{1}{2}$ (so that $\mathbb{E}[\alpha^U_k] = 1$) for all the cost terms. We then generate a partition mean $\bar{c}_{\mathcal{P}_i}$ by sampling uniformly at random from an interval $[c^{low}, c^{high}]$, where $c^{low} + c^{high} = 1$. Next, within each group, we choose an interval $[c^{low}_{\mathcal{P}_i}, c^{high}_{\mathcal{P}_i}]$, where $(c^{low}_{\mathcal{P}_i} + c^{high}_{\mathcal{P}_i})/2 = \bar{c}_{\mathcal{P}_i}$ and then sample uniformly at random the individual cost terms from this. All individual cost terms are set to strictly positive values, otherwise neither structured nor unstructured conditions hold, making the game instances trivial. The individual benefit is set to $\boldsymbol{b} = \mathbf{1}$ as it does not affect either Υ^U or Υ^S. In generating these values we fix the global mean but change the variance.

Using game instances generated through this procedure, we will separately compare the verification complexity and sufficiency gaps on the two sets of structured and unstructured conditions. We first use games where both sets of conditions are satisfied to allow a comparison of verification complexity. For the sufficiency gap comparison, we generate games using different parameter settings described above, and measure how frequently the structured conditions fail while the unstructured conditions are satisfied.

3.7.2 The Computational Complexity Gap

As discussed in Sections 3.4, the verification of uniqueness conditions on structured networks is of lower complexity. There are several factors that determine how big this complexity gap is, which we examine in this section.

The first factor is the size of the network (total number of agents). Specifically, the complexity gap increases with the number of agents, and is at least quadratic in the number of groups. Table 3.1 lists the verification complexity (in floating point operations, flops) of the conditions in Corollary 2 with a dense Γ^U matrix. We see that the verification complexity of structured conditions is orders of magnitude lower than that of the unstructured conditions, and the gap increases with the size of the network. Table 3.2 shows the difference in CPU times, where the results are averaged over

50 different instances. Moreover, the complexity reduction in verifying the conditions in Theorem 5 is much more significant than those in Corollary 2. For instance, the verification complexity of conditions in Theorem 5 on a game of size 10×10 (10 partitions of 10 agents each) on a dense Υ^U matrix is 1.08×10^{35} flops while that on the corresponding Υ^S matrix is 1.26×10^6 flops. We refer the interested reader to Appendix B.8 for additional comparisons.

Size ($N_i \times M$)	**Unstructured**	**Structured**
10×10	$6.67{\times}10^5$	$\mathbf{5.83{\times}10^4}$
20×20	$4.27{\times}10^7$	$\mathbf{1.99{\times}10^6}$
50×50	$1.04{\times}10^{10}$	$\mathbf{2.02{\times}10^8}$
$100{\times}100$	$6.67{\times}10^{11}$	$\mathbf{6.57{\times}10^9}$
$200{\times}200$	$4.27{\times}10^{13}$	$\mathbf{2.12{\times}10^{11}}$

Table 3.1: Verification Complexity in Number of FLOPs of Conditions in Corollary 2 over the Number of Agents.

Size ($N_i \times M$)	**Unstructured (sec)**	**Structured (sec)**
10×10	0.0031±0.0011	**0.003±0.0002**
20×20	0.0575±0.0074	**0.0213±0.0019**
50×50	5.851±0.197	**0.6768±0.0442**
100×100	332.8±1.6	**7.052±0.203**
200×200	Memory Overflow	**116.8±6.9**

Table 3.2: Verification Complexity in CPU Times of Conditions in Corollary 2 over the Number of Agents; All Times are in Seconds. All Experiments were Performed on a Machine with a 6-core 2.60/4.50 GHz CPU with Hyperthreaded Cores, 12MB Cache, and 16GB RAM.

The second factor affecting the complexity gap is how (a given number of) agents are partitioned into groups. Figure 3.5 shows the complexity (in flops) of verifying the structured condition (of Υ^S being a P-matrix) in two games of size 50 and 100 agents, respectively, both with a dense Υ^U matrix, as we vary the number of groups. Here, we set to groups to be of equal size, rounding off to the nearest integer if needed; e.g., with 10 agents and 3 groups, the partition sizes are 3, 3, 4.[1]

We see that in each game the complexity has a V shape, reaching a minimum when $M \approx \sqrt{N}$. To explain this, note that we can approximate the complexity of LU decomposition by $\frac{2}{3}p^3$ for a matrix of size $p \times p$; for singular value decomposition the complexity can be approximated by $2pk^2 - \frac{2}{3}k^2$, $p > k$, ([31, p. 75], based on QR decomposition using Householder transformations). Then the complexity for checking the conditions in Theorem 5 is given by ($S = N/M$):

$$\binom{M}{2}\left(2S^3 - \frac{2}{3}S^2\right) + M\sum_{k=1}^{S}\binom{S}{k}\left(\frac{2}{3}k^3\right) + \sum_{k=1}^{M}\binom{M}{k}\left(\frac{2}{3}k^3\right).$$

As $\max\{M!, S!\}$ will be the dominant term in the (expanded) expressions, the minimum is achieved

[1]Here we are comparing the structured complexity between games of different sizes. For a given size N with M equal-sized partitions, the structured and unstructured conditions have the same complexity when $M = 1$ or $M = N$.

when $M = S = \sqrt{N}$.

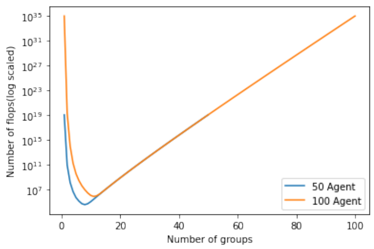


Figure 3.5: Verification Complexity (FLOPs) of Conditions in Theorem 5 over the Number of Groups.

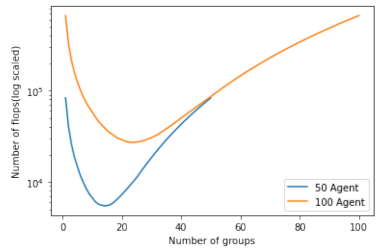


Figure 3.6: Verification Complexity (FLOPs) of Conditions in Corollary 2 over the Number of Groups.

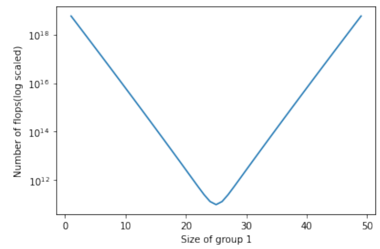


Figure 3.7: Complexity (FLOPs) over Different Partitions in Games with 50 Agents, 2 Groups and a Dense Υ^U Matrix.

Note also that in Figure 3.5 the two curves overlap almost completely beyond the minimum. This is because verifying whether Υ^S is a P-matrix has two phases: generating the Υ^S matrix followed by an eigendecomposition on it. When M is small the first phase is dominant, whereas when M is larger than $\sqrt{N}$ the second becomes dominant. Since Υ^S is of size $M \times M$, when M is large the complexity depends much more on the value M than on the network size N.

In another experiment, we verify the conditions in Corollary 2 on two games of sizes 50 and 100, respectively, both with a dense Υ^U matrix. The results are shown in Figure 3.6. This time the minimum occurs at some $M > \sqrt{N}$. This is because the approximated complexity is given by

$$\binom{M}{2}\left(2S^3 - \frac{2}{3}S^2\right) + \frac{2}{3}M^3 + \frac{2}{3}S^3,$$

which has a minimum at $M > \sqrt{N}$.

A third factor affecting the complexity gap is the size distribution of groups, given fixed N and M. Figure 3.7 shows the complexity of verifying whether Υ^S is a P-matrix in a 50-agent, 2-partition game with a dense Υ^U matrix, as we vary the size of the first group. We see that the complexity reaches its minimum when the two groups are equal sized.

More generally, when we have more action dimensions and create partitions based on them, we can expect that some of the off-diagonal elements in the structured matrix Υ^S will be computed with much lower complexity. To see why, consider two games, one with $N = 100$, $K = 1$ and 10 groups of size 10 each, the other with $N = 50$, $K = 2$ and 5 groups of size 10 each. If we create a partition for all agent-action components from the same group in the same action, we get 10 partitions for both games. While the first game is similar to the samples in this section, the second game has 20 off-diagonal elements in Υ^S computed from the coupled cost. Computing these 20 elements are easier than computing the 2-norm of a matrix since the corresponding block is a diagonal matrix. Moreover, if all agents have the same cost function, then these 20 elements are the same and based on one utility function, which makes the computation even easier.

3.7.3 The Sufficiency Gap

The structured conditions are stronger than their unstructured counterparts, and thus may fail to discover the uniqueness and stability of an NE in a game that fails the former while satisfying the latter. We showed this using an example in Section 3.4. In what follows we will use numerical results to measure this sufficiency gap. We will focus on the uniqueness conditions, and note that the comparison of stability conditions is very similar.

Each of the next set of figures is a heat map showing how often these two conditions in Corollary 2 (over Γ^S and Γ^U, respectively, with sample games that guarantee both Γ^S and Γ^U are symmetric) yield the same or different result. The former means either both are satisfied or not satisfied, while the latter necessarily means the structured condition fails and the unstructured condition holds. Each game is of size 400, with 20 groups of 20 members each. For each cell in the heat map, 50 sample games are generated using a set of parameters corresponding to the cell indicated on the figure; the cell color indicates the fraction of these games that resulted in a difference (sufficiency gap), the higher the fraction, the darker the color. In each of the heat maps, we can see regions of darker cells (clearly) separating the map into two lighter regions. In general, the bottom left represents parameter settings where both structured and unstructured conditions are satisfied and the top right represents settings where both conditions do not hold.

Specifically, in Figures 3.8, 3.9, and 3.10 we hold component-level external impact fixed (at weak, medium, and strong levels, respectively, corresponding to $P^{in}_{exist}, P^{out}_{exist}$ at 0.2, 0.5, 0.8 respectively; $(S^{in}_{low} + S^{in}_{high})/2$ at $0.2/N_i, 0.5/N_i, 0.8/N_i$ respectively, for partition $\mathcal{P}_i$ (normalizing the strengths to make external and internal impact comparable); $(S^{out}_{low} + S^{out}_{high})/2$ fixed at $0.2/N, 0.5/N, 0.8/N$ (normalize the strengths, similar as above), while changing the variances of component-level internal impact (within-partition along the x-axis by changing the value of $c^{high}_{\mathcal{P}_i} - c^{low}_{\mathcal{P}_i}$, and between-partition along the y-axis by changing the value of $c^{high} - c^{low}$). We note that the normalized upper bound for the model parameters are chosen at 1 when we study the sufficiency gaps, because higher values cause both sets of conditions to fail thereby reducing the significance of such sample games. Please refer to Appendix B.9 for more details.

Overall the sufficiency gap is quite low, i.e., the two types of conditions yield the same outcome in the vast majority of parameter settings (as evidenced by mostly 0 values/light-colored cells on these heat maps). The measured difference (obtained by adding the number of different results in every cell and dividing by the total number of sample games) in Figures 3.8, 3.9, and 3.10 are $0.26\%, 5.58\%, 0.05\%$, respectively.

Figures 3.8 and 3.9 show a similar pattern. In the top right corners where component-level internal impact has large variance both between and within groups, neither condition is true, while in the lower left corners both conditions are satisfied, giving rise to the broad agreement (light regions). In comparing the two, we see that the dark region expands and shifts leftward and downward in Figure 3.9, suggesting that the gap between the two conditions are bigger and triggered by lower variances in component-level internal impact when the component-level external impact increases from weak to medium. Furthermore, high within-partition variance combined with low between-partition variances in component-level internal impact results in the largest gap. The reason is the "weakest member effect" discussed in Section 3.4, where the α_i^S values depend highly on the minimum component-level internal impact α_k^U of members in $\mathcal{P}_i$, which makes the abstraction Υ^S inaccurate, causing a larger gap. Interestingly, an increase in between-partition variances in component-level internal impact can mitigate the above effect and reduce the gap. On the other hand, when the component-level external impact is sufficiently high, Figure 3.10 shows that the sufficiency gap all but disappears as now most game instances do not satisfy either condition.

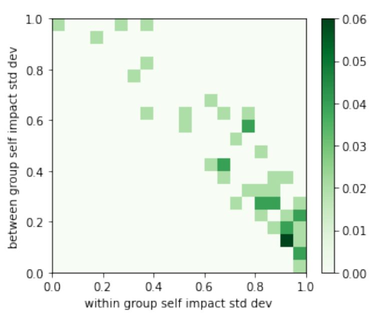


Figure 3.8: Sufficiency Gap Frequency over the Variance of Internal Impact, Weak External Impact.

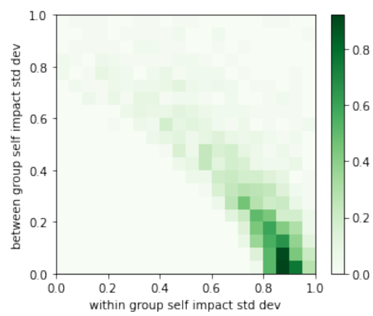


Figure 3.9: Sufficiency Gap Frequency over the Variance of Internal Impact, Medium External Impact.

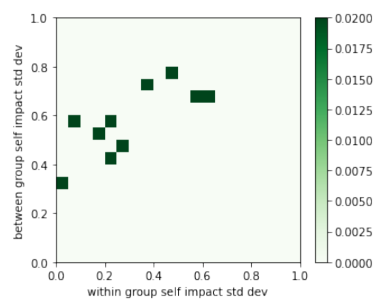


Figure 3.10: Sufficiency Gap Frequency over the Variance of Internal Impact, Strong External Impact.

We next fix the agent's component-level internal impact at 1, the between-partition (resp. within-group) component-level external impact at strong, and vary the within-partition (resp. between-group) component-level external impact strength (x-axis) and connection frequency (y-axis), shown in Figure 3.11 (resp. Figure 3.12). We see that the gap is generally small (2% gap for both). In both figures the disagreement is concentrated around a reciprocal curve, suggesting that when the product of the two parameters is around a critical level, the sufficiency gaps occur. They also suggest that the role of between-partition and within-partition external impact is very similar under these two sets of conditions. The dark regions in these two figures suggest that when individuals have a homogeneous internal impact, then the games where the expected sum of component-level external impact is about 10% higher than the expected sum of component-level internal impact are most likely to have sufficiency gaps. Please refer to Appendix B.9 for a detailed discussion on this phenomenon. Moreover, the dark cell curve in Figure 3.12 shows that when the between-partition connection frequency gets lower, the chance of having a sufficiency gap is higher. This is because a mismatch similar to Example 2 (an agent with weak component-level internal impact does not receive strong component-level external impact but a member in the same partition with strong component-level internal impact does) is more likely to happen when the between-partition connection frequency is lower.

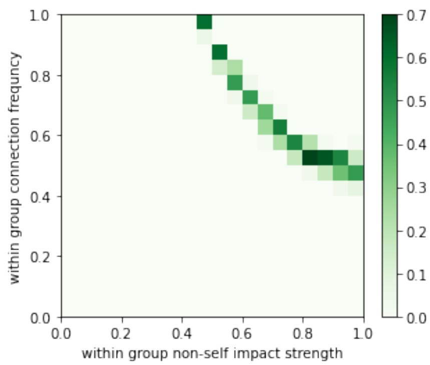


Figure 3.11: Sufficiency Comparison over the Within Partition Connections, when Every Agent Has the Same Internal Impact and Strong Between Partition Influence.

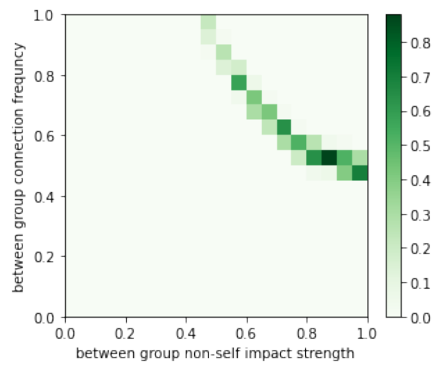


Figure 3.12: Sufficiency Comparison over the Between Partition Connections when Every Agent Has the same Internal Impact, Medium Within Partition Influence.

These numerical results suggest that in general, when network communities are formed around homogeneous agents, i.e., those with similar component-level internal impact, component-level external impact and connectivity, then the two types of conditions yield identical verification outcomes.

More generally, when we have more action dimensions and create partitions based on them, we can expect some of the off-diagonal elements in the structured matrix Υ^S to be computed from the coupled cost function. Similar to Section 3.7.2, we compare the sufficiency gap in two games, one with $N = 100$, $K = 1$ and 10 groups of size 10 each, the other with $N = 50$, $K = 2$ and 5 groups of size 10 each, and create 10 partitions for both games. Again, the first game is similar to the samples in this section, while the second game has 20 off-diagonal elements in Υ^S computed from the coupled cost. When all agents have the same cost functions, these 20 elements do not suffer from the strongest link effect and thus further reduce the sufficiency gap.

3.7.4 Sufficiency Gap on email-Eu-core Network Data

We next perform a similar set of experiments using a real-world directed graph. This dataset is from the email-Eu-core network [58] and provides a binary directed graph with edges indicating whether an email was ever sent from one node to another, where nodes represent people from a large European research institution and each node has a department affiliation. We will use this graph to run game simulations and test the identified conditions on this network's structured and unstructured Υ matrices. In doing so, we add an edge weight ϕ to all the edges; this is shown as the x-axis "external connection strength" in the figures below. We also need to equip the game with a utility function (detailed shortly).

This graph is highly asymmetric. As discussed in Section 3.4, the conditions on asymmetric Υ matrices are much more computationally costly to verify than the conditions on symmetric Γ matrices of the same size. As a result, verifying the conditions on the entire network with 1005 nodes leads to memory overflow. Consequently, our experiments are on down-sampled versions of this graph. We employ the following three types of network sampling methods to generate sub-graphs for our experiments:

- Sample M departments (groups) uniformly at random with all nodes in each;
- Sample N_{sample} nodes uniformly at random;
- Sample M departments (groups) with at least M members uniformly at random first, and then sample M nodes from each department.

We will use the following utility function of a node a_i (an agent) given the sampled network:

$$u_i(x_i, \boldsymbol{x}_{-i}) = \frac{\phi}{\overline{D}} \sum_{j \neq i} (G_{sample})_{ij} x_i x_j - \frac{1}{2} x_i^2,$$

where $\overline{D} = \frac{\sum_{i=1}^{N_{sample}} D_i^{in} + D_i^{out}}{2N_{sample}}$, N_{sample} is the number of sampled nodes, and G_{sample} is the corresponding sampled sub-matrix.

We want to compare different ways of partitioning the agent-action space. Since each node is associated with a department, we can naturally partition the network in terms of departments; we will call this the *original partition*. We can also create partitions on the nodes using the following criteria:

- Degree-based: rank the nodes by the average of the in-degree and out-degree of a node, and then create partitions that include nodes with similar degrees in the same partition;

- Connection-type-based: rank the nodes by the fraction of their within-department connections (number of connections in the department over the node's total number of connections), and create partitions that include nodes with similar fractions in the same partition.

As results in Figures 3.13-3.21 show, degree-based partition results in a significantly lower sufficiency gap and clearer patterns on when the structured conditions fail compared to the original partition. In contrast, the connection-type-based partition results in higher sufficiency gaps and thus is not ideal. The average sufficiency gap is given for each figure in the caption. We can see from the numerical results that when we re-group the nodes based on their degree information, not only the overall sufficiency gap is lower, but the sufficiency gaps are also less sensitive to the network size and only sensitive to the relevant internal/external impact strengths.

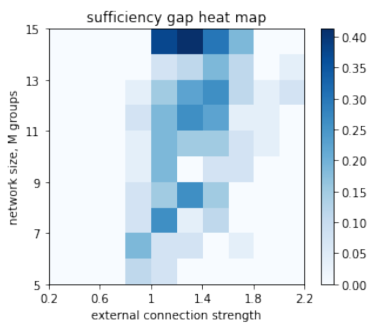


Figure 3.13: Sufficiency Gap when Sampling M Departments, Original Partition, 6.22%.

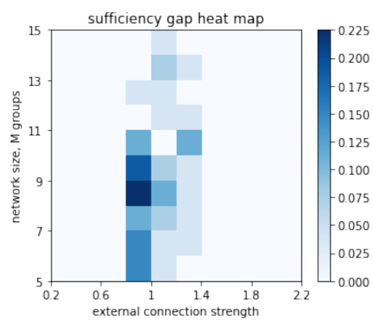


Figure 3.14: Sufficiency Gap when Sampling M Departments, Degree-based Partition, 1.84%.

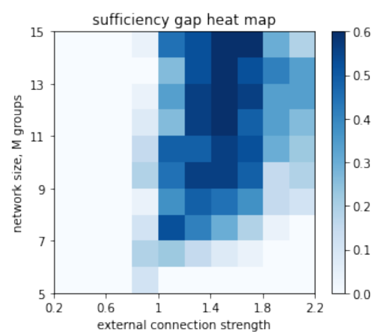


Figure 3.15: Sufficiency Gap when Sampling M Departments, Connection-type-based Partition, 20.40%.

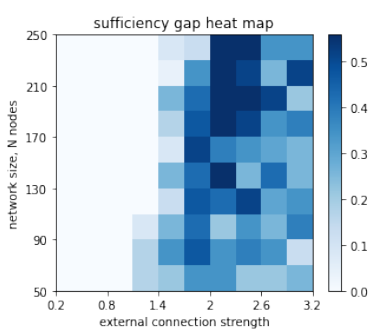


Figure 3.16: Sufficiency Gap when Sampling N Nodes, Original Partition, 21.39%.

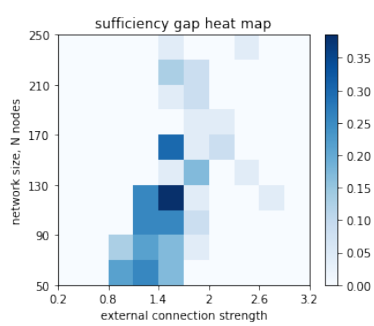


Figure 3.17: Sufficiency Gap when Sampling N Nodes, Degree-based Partition, 3.73%.

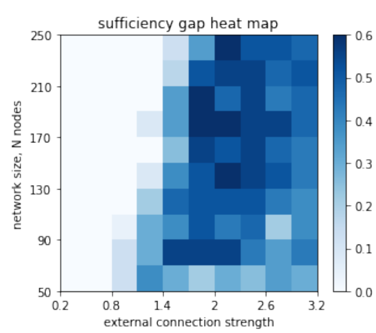


Figure 3.18: Sufficiency Gap when Sampling N Nodes, Connection-type-based partition, 28.63%.

Intuitively, these observations can be explained as follows. The high degree nodes in our sample network are frequently connected with each other. Therefore, the degree-based partition perform

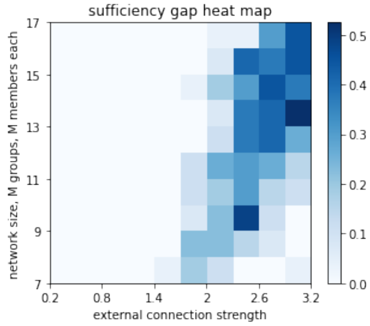


Figure 3.19: Sufficiency Gap when Sampling M Departments, Each with M Nodes, Original Partition, 9.94%.

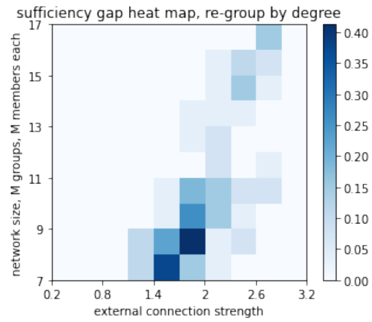


Figure 3.20: Sufficiency Gap when Sampling M Departments, Each with M Nodes, Degree-based Partition, 3.49%.

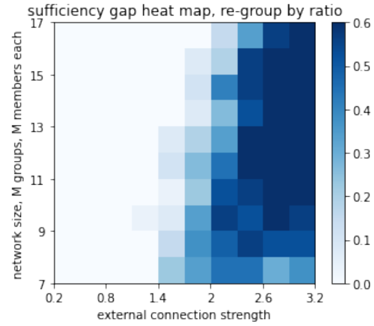


Figure 3.21: Sufficiency Gap when Sampling M Departments, Each with M Nodes, Connection-type-based Partition, 23.06%.

best, as they successfully identify and retain the importance of such strongly connected sub-graphs in our reduced conditions. Moreover, we note that the singleton nodes that have no connections to others are put in the same partition. From the previous analysis, this means that the structured and unstructured conditions hold if and only if the corresponding conditions hold on the non-singleton nodes' sub-graph.

3.7.5 Visualizing the Partition Centrality

We next create a relatively small multi-relational network game of $N = 30$, $K = 2$, and use this to visualize the centrality measures defined in Section 3.6. We generate the game to have symmetric inter-dependencies so that $D^{in} = D^{out}$ and compute the centrality with exogenous importance set to $\boldsymbol{e} = \mathbf{1}$.

The agents form 3 (pre-defined and color-coded) groups, each with a size of 10: we generate interaction graphs where group 1 has a high network influence levels on action dimension 1 but a low influence on dimension 2; group 2 has a low influence on action dimension 1 but a high influence on dimension 2; group 3 has low influence levels on both action dimensions. Figures 3.22 and 3.23 depict the interaction graphs on each action dimension. As can be seen, the red group is highly connected to both the yellow and blue groups in action 1, while the blue group is highly connected to the red and yellow in action 2.

With 3 groups and 2 action dimensions, we create 6 partitions, each consisting of agent-action components of one group on one action dimension. A partition is illustrated as a colored circle in Figures 3.24 and 3.25: the color corresponds to the group identity of the agents in that partition, with a label ("1" or "2") indicating the action dimension. The size of the circle indicates the value of that partition's generalized degree centrality (GDC) given in Definition 7, the larger the circle the higher the centrality. The links (their existence and thickness) between two partitions mirror the off-diagonal entries in the Υ^S matrix.

Figures 3.24 and 3.25 each represent a different type of cost function, coupled and decoupled. In the coupled cost, the action dimensions influence each other through the cost function $c(x^{(1)}, x^{(2)}) = \frac{1}{2}(x^{(1)})^2 + \frac{1}{2}(x^{(2)})^2 + \frac{1}{10}x^{(1)}x^{(2)}$. In the decoupled cost, the cost function is $c(x^{(1)}, x^{(2)}) = \frac{1}{2}(x^{(1)})^2 + \frac{1}{2}(x^{(2)})^2$. We see that since the red group has frequent and strong connections in action 1, the GDC of the partition R1 is high, so is B2. When the cost is coupled, the partitions of the same group of agents on different action dimensions are connected; otherwise the structured graph consists of disconnected components and we can solve the subgames on each dimension independently.

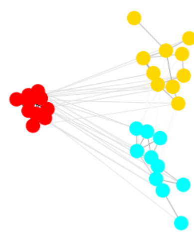


Figure 3.22: Action 1 Interaction Graph for the Experiments in Section 3.7.5. The Red Group Has a High Influence on Action Dimension 1.

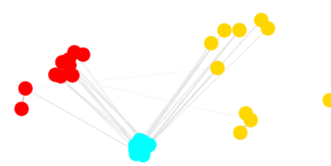


Figure 3.23: Action 2 Interaction Graph for the Experiments in Section 3.7.5. The Blue Group Has a High Influence on Action Dimension 2.

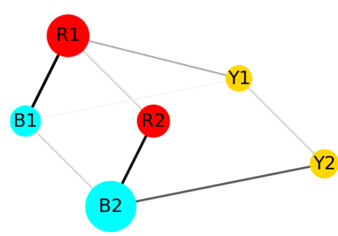


Figure 3.24: Visualization of Partitions' GDC with *Coupled* Cost. The Largest Nodes, "R1" and "B2", Denoting the Partition of the Agent-action Components from Group Red on Action 1 and Group Blue on Action 2, Respectively, Have the Highest GDC.

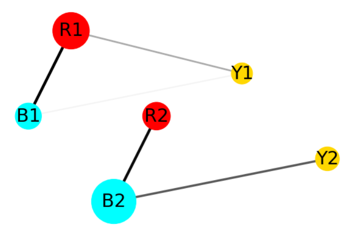


Figure 3.25: Visualization of Partitions' GDC with *Decoupled* Cost. The Sizes of the Nodes are Again Proportional to Each Partition's GDC. As the Action Dimension Costs are Decoupled, the Graph Consists of Two Subgraphs, One for Each Action Dimension.

3.8 Chapter Conclusions

In this chapter, we introduced and studied a family of structured network games with non-linear best response functions. Prior works on network games have found sufficient conditions for the uniqueness and stability of Nash equilibria which are mostly difficult to verify. We also showed that the existence of structure in the network (e.g., in the form of communities, or when there are multi-relational dependencies between agents), helps us find alternatives for such conditions, which we refer to as “structured conditions” as opposed to the “unstructured conditions” in previous works. In particular, we show that the structured conditions for the uniqueness and stability of Nash equilibria are related to matrices which are possibly lower dimensional, with their dimensions depending on the number of partitions naturally arising in a network due to its structured nature. We also demonstrated both analytically and numerically that the structured conditions are sufficient conditions to the unstructured conditions, and that their verification is of much lower computational complexity. We used numerical experiment results to show that the sufficiency gap between the structured conditions and unstructured conditions is small in general and typically occurs in games with some specific characteristics. Moreover, we proposed a new notion of degree centrality to evaluate the influence of a partition in the network, and used it to identify additional conditions for the uniqueness and stability.

Part II

Intervention and Mechanisms in Network Games, Multi-group Systems and Multi-Scale Networks

Chapter 4

Multi-planner Intervention in Network Games with Community Structures

4.1 Introduction

Part I of the thesis focused on modeling and analysis of multi-scale network games. Part II will now focus on control and intervention. In this chapter, we study the problem of intervention in network games where the network has a group structure with local planners, each associated with a group. The agents play a non-cooperative game while the planners may or may not have the same optimization objective. We model this problem using a sequential move game where planners make interventions followed by agents playing the intervened game. We provide equilibrium analysis and algorithms that find the Stackelberg equilibrium. We also propose a two-level efficiency definition to study the efficiency loss of equilibrium actions in this type of games.

As discussed in previous chapters, the strategic decision making of (physically or logically) connected agents is often studied as a network game, where the utility of an agent depends on its own actions as well as that of those in its neighborhood as defined by an interaction graph or adjacency matrix.

Within this context, *intervention* in a network game typically refers to changes in certain game parameters made by a utilitarian welfare maximizer with a budget constraint, who wishes to induce a more socially desirable outcome (in terms of social welfare) under the revised game. A prime example is the study presented in [30], where interventions take the form of changing the agents' standalone marginal benefit terms (in a linear quadratic utility model) and changes are costly; this is done by a central/global planner, who wishes to find the set of interventions that lead to the highest equilibrium social welfare subject to a cost constraint.

As discussed in Chapter 1, finding optimal interventions could be viewed as a form of mechanism design, because in both cases the design or intervention essentially induces a new game form with desirable properties. But there are a few novelties in interventions compared to the conventional mechanism design framework. Specifically, conventional mechanism design is often not limited to a specific game form, the latter being the outcome of the design, while interventions typically start from specified game forms and seek improvement through local changes. Moreover, interventions

[30] aim to optimize the planner's objectives under the constraints of a budget and specified forms of intervention.

In this chapter, we are interested in intervention in a network game where the network exhibits a group or community structure and each group or community has its own local group planner. Since group structures are a common phenomenon across networks of all types, be it social, technological, political, or economic, this modeling consideration allows us to investigate a number of interesting features that often arise in realistic strategic and decentralized decision making. For instance, a single global budget may be first divided into separate chunks of local budgets at the local planners' disposal; these local budgets may or may not be transferred from one community to another, and the local planners' decisions may or may not take into account the connectivity between themselves and other neighboring communities; local planners may or may not wish to cooperate with each other; and so on.

Of particular interest to our study is the issue of efficiency in this type of decision making systems. A standard notion used to measure efficiency loss in a strategic game is the Price of Anarchy (POA); this is defined as the ratio of the maximum social welfare (sum utility) divided by the social welfare attained at the worst case NE (in terms of social welfare). The numerator is what a social planner aims for, while the denominator is the result of agents optimizing their own utilities and best-responding to each other. POA has been extensively studied in a variety of games, including in interdependent security games such as [87, 70], where agents' incentive to free-ride or over-consume contributes to the efficiency loss; in routing and congestion games [86, 21]; and in network creation games [25].

It is not hard to see additional sources of efficiency loss exist in the community intervention problem we are interested in: in addition to agents' self-interested decision making, local planners' non-cooperation, as well as sub-optimal budget allocation among groups, can both results in efficiency loss. The main findings of the chapter are summarized as follows:

1. We show that through backward induction the planners can obtain a reduced version of the planners' game that only depends on each other's intervention profiles. Regardless of being cooperative or not, the sequential game always has a unique Stackelberg equilibrium. Moreover, this equilibrium can be achieved through a decentralized algorithm based on the best responses of the planners.

2. We introduce a two-level definition of efficiency loss that allows us to discuss how the planners' actions influence the outcome of the game separately from the agents' actions, and we show that the efficiency loss due to the planners' non-cooperation can be characterized with the budget constraints and shadow prices.

3. We present numerical results on welfare and efficiency in several commonly seen types of

interaction graphs and commonly used budget allocation rules.

The remainder of the chapter is organized as follows. Section 4.2 introduces our intervention game model and presents the objectives for agents and group planners in different scenarios. Then in section 4.3, we show our analysis and characterization of the Stackelberg equilibrium of the intervention game. In section 4.4, we study the Level-1 and Level-2 efficiencies of the Stackelberg equilibrium. We present our numerical experiment results in section 4.5. Finally, section 4.6 concludes this chapter.

4.2 Game Model

We consider a network game among N agents, denoted by $a_1, \ldots, a_N$, represented by a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is the set of nodes/agents and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ the set of edges. Let $G = (g_{ij})_{i,j}$ denote the adjacency matrix, assumed to be symmetric and as a convention $g_{ii} = 0$; $g_{ij} \neq 0$ implies dependence between a_i and a_j, $i \neq j$.

Agents are divided into M disjoint communities, the kth community denoted as S_k with size N_k. Agent a_i takes an action $x_i \in \mathbf{R}$. Let $\boldsymbol{x}_{-i} = [x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_N]^T$ denote the action profile of all except a_i, $\boldsymbol{x}_{S_k} = (x_i)_{a_i \in S_k}$ the action profile of members in community S_k, and $\boldsymbol{x}_{-S_k}$ the action profile of all agents other than members of S_k.

We consider a family of games with utility:

$$u_i(x_i, \boldsymbol{x}_{-i}, y_i) = (b_i + y_i)x_i - \frac{1}{2}x_i^2 + x_i\left(\sum_{j\neq i} g_{ij}x_j\right), \tag{4.1}$$

which depends on the action profile and a real valued parameter y_i controlled by a planner. This utility function with intervention is studied in [30, 18]. The $-\frac{1}{2}x_i^2$ is the individual cost for a_i, the $b_i x_i$ term is the initial individual marginal benefit, and $x_i\left(\sum_{j\neq i} g_{ij}x_j\right)$ models the network influence. The intervention component can be seen as a linear subsidy (discount) term if $y_i > 0$ and a linear penalty (price) term if $y_i < 0$. In this non-cooperative game, the optimization problem of agent a_i for given intervention is

$$\underset{x_i}{\text{maximize}} \;\; u_i(x_i, \boldsymbol{x}_{-i}, y_i). \tag{4.2}$$

The NE of the game $\boldsymbol{x}^*$, is the action profile where no agent has an incentive to unilaterally deviate, i.e.,

$$x_i^* = \underset{x_i}{\text{argmax}} \;\; u_i(x_i, \boldsymbol{x}_{-i}^*). \tag{4.3}$$

We denote the planner for S_k as p_k, which has a budget constraint $C_k > 0$: $\sum_{a_i \in S_k} y_i^2 \leq C_k$.

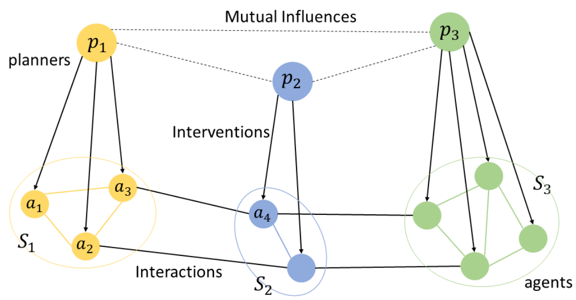


Figure 4.1: An Intervention Game with 3 Communities.

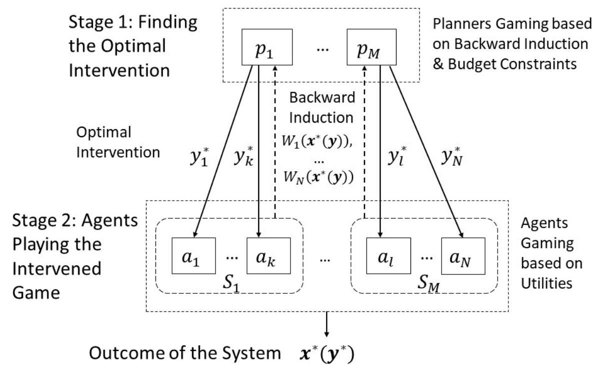


Figure 4.2: A Flow Chart of the Intervention Game.

We denote $\boldsymbol{y}_{S_k} = (y_i)_{a_i \in S_k}$ as the intervention profile of p_k and $\boldsymbol{y}_{-S_k}$ the intervention profile of planners other than p_i. Denote $Q_k = \{\boldsymbol{y}_{S_k} | \sum_{a_i \in S_k} y_i^2 \leq C_k\}$; thus Q_k is nonempty, convex and compact. Finally, $Q = \prod_{i=1}^M Q_i$.

We consider two cases. In the first, planner p_k is a *group-welfare maximizer*, whose objective is to maximize the sum of its members' utilities at the NE, formally

$$\underset{\boldsymbol{y}_{S_k} \in Q_k}{\text{maximize}} \;\; U_k(\boldsymbol{x}^*, \boldsymbol{y}_{S_k}) = \sum_{a_i \in S_k} u_i(\boldsymbol{x}^*, y_i), \tag{4.4}$$

and we denote $\boldsymbol{y}^*_{S_k} = \underset{\boldsymbol{y}_{S_k} \in Q_k}{\text{argmax}} \; U_k(\boldsymbol{x}^*, \boldsymbol{y}_{S_k})$. When all planners are group-welfare maximizers, we say they are *non-cooperative*.

In the second case, planner p_k is a *social-welfare maximizer*, whose objective is to maximize the sum of all agents' utilities at the NE, formally

$$\underset{\boldsymbol{y}_{S_k} \in Q_k}{\text{maximize}} \;\; U(\boldsymbol{x}^*, \boldsymbol{y}_{S_k}, \boldsymbol{y}_{-S_k}) = \sum_{i=1}^{N} u_i(\boldsymbol{x}^*, y_i), \tag{4.5}$$

and we denote $\overline{\boldsymbol{y}}_{S_k} = \underset{\boldsymbol{y}_{S_k} \in Q_k}{\text{argmax}} \; U(\boldsymbol{x}^*, \boldsymbol{y}_{S_k}, \overline{\boldsymbol{y}}_{-S_k})$. When all planners are social-welfare maximizers, we say they are *cooperative*, i.e., they have a common interest.

Figure 4.1 shows the structure of the intervention game described in this section. It's easy to see that with a single planner ($M = 1$), the above may be viewed as a two-stage game: the first mover the planner chooses the intervention actions $\mathbf{y}$ in anticipation of the (simultaneous) second movers the agents playing the induced game with actions $\mathbf{x}$. There is a similar two-stage sequentiality in the case of M local planners as shown in Figure 4.2: the local planners are simultaneous first movers in choosing interventions for their respective communities, in anticipation of interventions by other local planners and actions by the simultaneous second movers the agents. For this reason, the solution concept we employ in this study is the Stackelberg equilibrium.

4.3 The Stackelberg Equilibrium

In this section, we characterize the Stackelberg equilibrium of the system and introduce an algorithm to compute it. We assume the following holds throughout this chapter:

Assumption 5. *Matrix $2diag(\boldsymbol{c}) - W$ is positive definite.*

We start by computing the NE under an arbitrary intervention, we can compute the first order derivatives as follows

$$\frac{\partial u_i}{\partial x_i} = -x_i + \sum_{j \neq i} g_{ij} x_j + (b_i + y_i), \tag{4.6}$$

then by computing the fixed point, we know the unique NE of the game is

$$\boldsymbol{x}^* = (I - G)^{-1}(\boldsymbol{b} + \boldsymbol{y}), \tag{4.7}$$

Denote $A = (I - G)^{-2}$ for simplicity of notation. This NE is known to all planners through backward induction.

4.3.1 Finding the Stackelberg Equilibrium

We denote G_{S_k,S_l} as the block of G corresponding to the rows in S_k and columns in S_l, and $G_{S_k,\star}$ as the block of G corresponding to the rows in S_k and all columns. It's worth noting that given the representation of $\boldsymbol{x}^*$ in Eqn (4.7), the objective of a group welfare maximizer p_k is (See Appendix)

$$U_k(\boldsymbol{x}^*, \boldsymbol{y}_{S_k}) = \frac{1}{2}(\boldsymbol{x}^*_{S_k})^T \boldsymbol{x}^*_{S_k}. \tag{4.8}$$

We can then rewrite the objective of $p_k, \forall k$, and the non-cooperative optimization problem (P-NCk) as

$$\begin{aligned} \text{maximize} \;\; & W_k(\boldsymbol{y}) = \frac{1}{2}\|(A^{1/2}(\boldsymbol{y} + \boldsymbol{b}))_{S_k}\|_2^2 \\ \text{subject to} \;\; & \sum_{i: a_i \in S_k} (y_i)^2 \leq C_k \; . \end{aligned} \tag{4.9}$$

It's worth noting that this doesn't imply the planners' optimization problems are independent, since we can write $\boldsymbol{x}^*_{S_k}$ as $\boldsymbol{x}^*_{S_k} = (A^{1/2}_{S_k,\star})(\boldsymbol{y} + \boldsymbol{b})$, which depends on $\boldsymbol{y}_{-S_k}$ unless S_k is isolated.

Similarly, we can rewrite the cooperative optimization problem (P-C) where all planners are social welfare maximizers

$$\begin{aligned} &\text{maximize} \;\; W(\boldsymbol{y}) = \frac{1}{2}(\boldsymbol{y}+\boldsymbol{b})^T A(\boldsymbol{y}+\boldsymbol{b}) \\ &\text{subject to} \;\; \sum_{i:a_i \in S_k} y_i^2 \le C_k, \;\; \forall k \end{aligned} \tag{4.10}$$

We can also write out the decentralized version of (P-C) where each planner p_k has its own optimization problem (P-Ck) given other planners' intervention profile $\boldsymbol{y}_{-S_k}$

$$\begin{aligned} &\text{maximize} \;\; W(\boldsymbol{y}) = \frac{1}{2}(\boldsymbol{y}+\boldsymbol{b})^T A(\boldsymbol{y}+\boldsymbol{b}) \\ &\text{subject to} \;\; \sum_{i:a_i \in S_k} (y_i)^2 \le C_k. \end{aligned} \tag{4.11}$$

We have the following result.

Theorem 10. *If all planners are group-welfare maximizers or if they are all social-welfare maximizers, then in each case there is a unique optimal intervention, i.e., unique Stackelberg equilibrium, and under the optimal intervention, the budget constraints are tight.*

This is obvious when all planners are social-welfare maximizers from Eqn (4.10); for the other case see Appendix. We also propose the following decentralized algorithm based on best response dynamics (BRD) that computes the Stackelberg equilibrium in both cases. Note that the planners' best-response computation utilizes Eqn (4.9) and (4.10).

Algorithm 5: Planners' BRD

```
Initialize: y(0) = y_0, t = 0
while y not converged do
    for k = 1 : M do
        y_{S_k}(t+1) = argmax_{y_{S_k} ∈ Q_k} U_k(y_{S_k}, y_{-S_k}(t))
        (Best response w.r.t objective U_k, which can be either group or social welfare)
    t ← t + 1
Set optimal intervention profile as y* = y(t)
x* = (I − G)^{-1}(b + y*)
```

Theorem 11. *If all planners are group-welfare maximizers or if they are all social-welfare maximizers, then in each case Algorithm 5 converges to the unique Stackelberg equilibrium under Assumption 5.*

Proof. *(Sketch)* The proof is based on the Jacobian of the best response mappings of the planners. The Jacobian matrix is positive definite when Assumption 5 is true, and thus there is a unique fixed point, and the best response mappings have contraction properties. Therefore, the algorithm converges to the unique fixed point, i.e., the unique optimal intervention profile, which then leads to the unique Stackelberg equilibrium in the game. □

Please see Appendix for the full proof.

Proposition 1. *The following cooperative optimization problem (P-NC-alt) has the same optimal intervention outcome as the original non-cooperative problem (P-NCk) where all planners are group-welfare maximizers:*

$$\begin{aligned} &\textit{maximize} \;\; \tilde{W}(\boldsymbol{y}) = \frac{1}{2}(\boldsymbol{y}+\boldsymbol{b})^T \tilde{A}(\boldsymbol{y}+\boldsymbol{b}) \\ &\textit{subject to} \;\; \sum_{i:a_i \in S_k} y_i^2 \leq C_k, \forall k \end{aligned} \tag{4.12}$$

where

$$\tilde{A} = \begin{bmatrix} A_{S_1,S_1} & \frac{1}{2}A_{S_1,S_2} & \cdots & \frac{1}{2}A_{S_1,S_M} \\ \frac{1}{2}A_{S_2,S_1} & A_{S_2,S_2} & \cdots & \frac{1}{2}A_{S_2,S_M} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2}A_{S_M,S_1} & \frac{1}{2}A_{S_M,S_2} & \cdots & A_{S_M,S_M} \end{bmatrix}$$

Proof. (Sketch) This is obtained by studying the gradient of each planner's objective function. The above cooperative optimization problem in Eqn (4.12) has the exact same gradient for every planner in the original non-cooperative optimization problem in Eqn (4.9). □

Please see the appendix for the full proof. We also characterize the direction of the optimal intervention profile in the appendix.

4.3.2 Lagrangian Dual and Shadow Prices

Next, we introduce some concepts related to the Lagrangian dual variables and *shadow prices*, which will be used to characterize efficiency budget sharing in the next section.

Since Q clearly satisfies Slater's Constraint Qualification [9], the planners' optimization problems are convex regardless of whether they are group or social welfare maximizers. Then based on the Karush–Kuhn–Tucker(KKT) [9] conditions, we know that strong duality holds for both cooperative and non-cooperative planners' optimization problems. If we define the Lagrangian as

$$L(\boldsymbol{y}, \boldsymbol{\lambda}) = W(\boldsymbol{y}) + \sum_{k=1}^{M} \lambda_k (C_k - \|\boldsymbol{y}_{S_k}\|^2),$$

and $W(\boldsymbol{y})$ is either social or group welfare, then we can obtain an optimal dual λ_k^*, the shadow price for p_k. We can then equivalently think of p_k's problem as maximizing the above Lagrangian with a cost of intervention $\lambda_k \sum_{a_i \in S_k} y_i^2$. For the convenience of notation, we use λ_k^* (resp. $\overline{\lambda}_k$) to denote the dual optimal variable corresponding to group (resp. social) welfare maximization problem in the (P-NCk) (resp. (P-C)) problems for the planners. We will later use the Lagrangian values λ_k^* and $\overline{\lambda}_k$ to characterize the efficiency of the Stackelberg equilibrium in Section 4.4.

4.4 Efficiency and the Budget Allocation

In this section, we discuss the efficiency of the subgame perfect equilibria under a fixed budget allocation and then study the impact of different budget allocations on the equilibrium and its efficiency.

4.4.1 Efficiency of the Stackelberg Equilibrium

For conventional single-planner multi-agent systems, the efficiency of an NE is characterized as the ratio of the social objective value in the NE divided by the socially optimal outcome, formally

$$e(\boldsymbol{x}^*) = \frac{U(\boldsymbol{x}^*)}{\max_{\boldsymbol{x} \succeq \boldsymbol{0}} U(\boldsymbol{x})},$$

and an upper bound on its reciprocal is referred to as the *price of anarchy* (PoA) if the objective $U(\boldsymbol{x}) = \sum_{i=1}^N u_i(\boldsymbol{x})$. This maxima is achievable if the agents' utility functions are strictly individually concave and always have a zero point in the first order derivative.

The introduction of group planners in our intervention problem means there are now multiple sources of efficiency loss. Accordingly, we will decompose this into a level-1 (L1) component and a level-2 (L2) component, caused by the non-cooperative nature of agents and planners, respectively. Following the notation of $\boldsymbol{y}^*$ and $\overline{\boldsymbol{y}}$ in Eqn (4.4) and (4.5), we formally define the two efficiency loss measures as

$$e_{L1}(\boldsymbol{y}) = \frac{U(\boldsymbol{x}^*, \boldsymbol{y})}{\max_{\boldsymbol{x}} U(\boldsymbol{x}, \boldsymbol{y})}, \quad e_{L2} = \frac{W(\boldsymbol{x}^*, \boldsymbol{y}^*)}{W(\boldsymbol{x}^*, \overline{\boldsymbol{y}})}. \tag{4.13}$$

Thus the overall efficiency, which resembles the conventional definition, can be written as

$$e(\boldsymbol{x}^*, \boldsymbol{y}^*) = \frac{U(\boldsymbol{x}^*, \boldsymbol{y}^*)}{\max_{\boldsymbol{x} \succeq \boldsymbol{0}, \boldsymbol{y} \in Q} U(\boldsymbol{x}, \boldsymbol{y})} = e_{L2} \cdot e_{L1}(\overline{\boldsymbol{y}}) \; .$$

The L1 efficiency has been well studied in the literature, e.g., [46]. For an arbitrary intervention profile $\boldsymbol{y}$, if $I - 2G \succ \mathbf{0}$, then L1 efficiency can be written as

$$e_{L1}(\boldsymbol{y}) = \frac{(\boldsymbol{b}+\boldsymbol{y})^T (I-G)^{-2} (\boldsymbol{b}+\boldsymbol{y})}{(\boldsymbol{b}+\boldsymbol{y})^T (I-2G)^{-2} (\boldsymbol{b}+\boldsymbol{y})} \ ,$$

since

$$\frac{\partial(\sum_{i=1}^N u_i)}{\partial x_i} = -x_i + 2\sum_{j\neq i} g_{ij} x_j + (b_i + y_i),$$

and by computing the fixed point we know the action profile maximizing the social welfare is $\overline{\boldsymbol{x}} = [I - 2G]^{-1}(\boldsymbol{b}+\boldsymbol{y})$.

We have the following result on the L2 efficiency.

Theorem 12. *When $\boldsymbol{b} = \mathbf{0}$ or $C_k \gg \|\boldsymbol{b}_{S_k}\|_2^2, \forall k$, the welfare in (P-C) can be computed by $W = \sum_{k=1}^M \overline{\lambda}_k C_k$, and a lower bound on the L2 efficiency for a given set of budgets is*

$$e_{L2} \geq \frac{\sum_{k=1}^M (2\lambda_k^* - \frac{1}{2}\rho_k) C_k}{\sum_{k=1}^M \overline{\lambda}_k C_k}, \tag{4.14}$$

where ρ_k denotes the spectral radius of A_{S_k,S_k}.

Proof. (Sketch) When $\boldsymbol{b} = \mathbf{0}$ or $C_k \gg \|\boldsymbol{b}_{S_k}\|_2^2, \forall k$, the optimal intervention $\boldsymbol{y}^*$ becomes the significant part in deciding the L2 efficiency. The shadow prices and current interventions jointly determine $\boldsymbol{x}_{S_k}^*$. Since the budget is binding, we can replace the lengths of optimal interventions with budget values and thus shadow prices and budgets jointly determine the efficiency. □

Please see the appendix for the full proof.

4.4.2 Budget Allocation and Budget Transferability

We say the budget is transferable if the individual budgets C_k are fungible and only the aggregate budget constraint (C) has to be satisfied. We say the budget is non-transferable if C_k is fixed and cannot be violated for all p_k. When all planners are social welfare maximizers and the budget is transferable, the optimization problem reduces to

$$\begin{aligned} \text{maximize} \ \ & W(\boldsymbol{y}) = \frac{1}{2}(\boldsymbol{y}+\boldsymbol{b})^T A (\boldsymbol{y}+\boldsymbol{b}) \\ \text{subject to} \ \ & \sum_{i=1}^N y_i^2 \leq C \ . \end{aligned} \tag{4.15}$$

This problem is well studied in [30].

It is obvious that when group planners are social-welfare maximizing, they have incentives to share the budget since they have a common objective. However, it turns out that even when planners are selfish, group-welfare maximizers, they may still have incentives to share the budget. Intuitively, this is because each group S_k has a decreasing marginal benefit in investing in itself, and if a neighboring group S_l has a strong enough positive externality on S_k and has a relatively low budget C_l compared to C_k, then p_k will have the incentive to transfer some of its budget to p_l.

Proposition 2. *Between two neighboring groups S_k and S_l, where $W_{S_k,S_l} \neq \mathbf{0}$, if the following inequality holds, then p_k has an incentive to share its budget with p_l:*

$$(C_k - C_l)(\nabla_{\boldsymbol{y}_{S_l}} W_l)^T \boldsymbol{y}_{S_k} \geq C_l (\nabla_{\boldsymbol{y}_{S_k}} W_k)^T \boldsymbol{y}_{S_k}.$$

Please see the appendix for the proof.

We note that compared to non-transferable budget, transferable budget can enable Pareto superior solutions to the system, where every agent and every planner gets a higher payoff.

Example 3. *Consider the following game with only two agents, each as a singleton group, and the following utility functions*

$$u_1(x_1, x_2, y_1) = x_1 - \frac{1}{2}x_1^2 + \frac{1}{2}x_1 x_2 + y_1 x_1,$$
$$u_1(x_1, x_2, y_2) = x_2 - \frac{1}{2}x_2^2 + \frac{1}{2}x_1 x_2 + y_2 x_2.$$

In this case, we have

$$(I - G)^{-1} = \frac{2}{3}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix},$$

$$x_1^* = 2 + \frac{2}{3}(2y_1 + y_2), \quad x_2^* = 2 + \frac{2}{3}(y_1 + 2y_2).$$

Suppose the initial budget is $C_1 = 25$, $C_2 = 0$. When not sharing the budget, the planners will fully invest in y_1 and y_2 respectively and the resulting Stackelberg equilibrium is $x_1^* = 2 + \frac{2}{3}(10+0) = \frac{26}{3}, x_2^* = 2 + \frac{2}{3}(5+0) = \frac{16}{3}$. But if p_1 shares the budget and make it $C_1 = 16, C_2 = 9$, we will have the equilibrium at $x_1^* = 2 + \frac{2}{3}(8+3) = \frac{28}{3}, x_2^* = 2 + \frac{2}{3}(4+6) = \frac{20}{3}$. So with budget sharing, we obtain a uniformly better outcome for all involved.

4.5 Numerical Results

We present numerical results in this section.

4.5.1 Budget Allocation and Network Types

Our focus is on examining the L2 efficiency with a number of commonly used budget allocation rules under the following types of networks/interaction graphs.

1. Type 1: strong within-group connection, weak between-group connection. In this type of networks, groups are used to model local and regional organizations formed by individuals; within each organization, agents interact much more frequently and have higher dependencies on local neighbors' decisions. Mathematically, this means that the diagonal blocks G_{S_k,S_k} have more non-zero elements and the non-zero elements have larger absolute values compared to the off-diagonal blocks G_{S_k,S_l}.

2. Type 2: weak within-group connection, strong between-group connection. This is the opposite of Type 1; in this case the off-diagonal blocks are now more frequently filled with larger elements. This type of networks can be used to model logical connectivity, where a group represents a set of agents playing the same role in a game. For example, in a network of sellers and buyers of a set of goods, a seller may interact more frequently with buyers than another seller. In the extreme case where sellers (resp. buyers) only interact with buyers but not with other sellers (resp. buyers), a multipartite graph can be used to capture their interactions.

3. Type 3: evenly distributed connections. Here groups become a rather arbitrarily constructed concept that may not correspond to agent interactions in a game.

We will consider the following three types of budget allocation.

1. Proportional: each group is assigned a budget proportional to its size, i.e., $C_k = \frac{N_k}{N}C$

2. Identical: each group is assigned an equal share of the total budget, i.e., $C_k = C/M$.

3. Cooperative socially optimal: the allocation in the optimal solution of the cooperative optimization problem (Eqn (4.5)), where the shadow prices λ_k^* are the same for all k.

Sample games used in the numerical experiments are generated as follows. In generating a random symmetric G, the diagonal elements are set to 0 as previously described in Section 2.3. The off-diagonal elements in the diagonal blocks are generated using a Bernoulli distribution with parameter P_{exist}^{in}, the probability for an edge (non-zero element) to exist between a pair of agents. The absolute value of a non-zero element (strength of a connection) $|g_{ij}|$ is drawn from a uniform distribution on the interval $[S_{low}^{in}, S_{high}^{in}]$. The off-diagonal blocks of G are similarly generated using the same approach, with parameters P_{exist}^{out} and $[S_{low}^{out}, S_{high}^{out}]$, respectively. The signs of the connections are assigned to yield the following two types of games. In the first, within-group

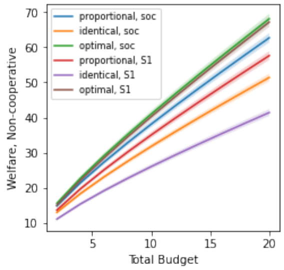


Figure 4.3: Type 1 All Positive Network.

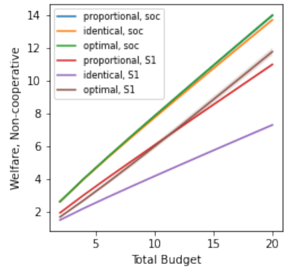


Figure 4.4: Type 1 Conflicting Groups.

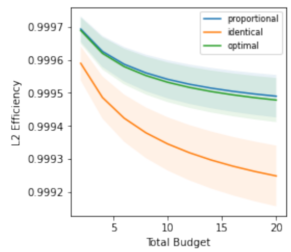


Figure 4.5: Type 1 All Positive Network.

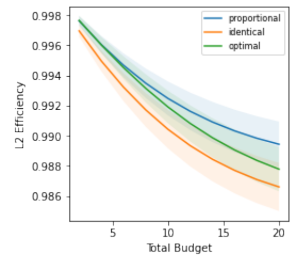


Figure 4.6: Type 1 Conflicting Groups.

connections and between-group connections have the same sign (all positive); in the second, they have opposite signs (positive within-group, negative between-group; this is also referred to as conflicting groups below). The $\boldsymbol{b}$ vector is generated by sampling every element uniformly from an interval $[b_{low}, b_{high}]$.

For strong connections, $P_{exist} = 0.8$ and $S_{low} = 0.7, S_{high} = 0.9$. For weak connections, $P_{exist} = 0.2$ and $S_{low} = 0.1, S_{high} = 0.3$. For evenly distributed networks, $P^{in}_{exist} = P^{out}_{exist} = 0.5$ and $S^{in}_{low} = S^{out}_{low} = 0.4, S^{in}_{high} = S^{out}_{high} = 0.6$. We then normalize the generated G by the total number of agents in the game to make sure that Assumption 5 holds [1]. We also choose $b_{low} = 0.1, b_{high} = 0.5$ to make sure that agents will have an initial incentive to take action above 0 and the budget can easily achieve $C_k \gg \|\boldsymbol{b}_{S_k}\|_2^2$. These sample games contain two groups, S_1 with 40 agents and S_2 with 10 agents; we obtained very similar results with more groups and thus will focus on this setting for brevity.

4.5.2 Social Welfare and L2 Efficiency

For each network type, we show the social welfare with non-cooperative planners and the L2 efficiency on example games with different types of budget allocation rules.

In general, the L2 efficiency is fairly high in all cases except for Type 2 networks with conflicting groups. Therefore, we only show the welfare results in the (P-C) problem. The main reason for this phenomenon is Assumption 5, where we require the elements of G to have relatively small values compared to 1 and thus in all except for Type 2 with conflicting groups, the difference between matrices A and $\tilde{A}$ is small. The major cause of welfare differences comes from budget allocation rules.

Figure 4.3 and 4.5 show the welfare of non-cooperative planners and the L2 efficiency in Type 1 network with all positive connections. In this case, the socially optimal budget allocation yields the

[1] If the product of the expected connection strengths and the connection frequency is fixed, the results are very similar. For this and brevity reasons we don't show results with combinations of low/high connection frequency with strong/weak connections.

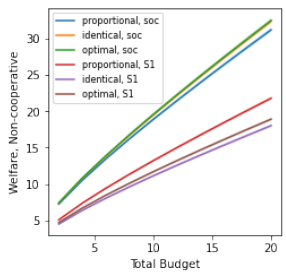


Figure 4.7: Type 2 All Positive Network.

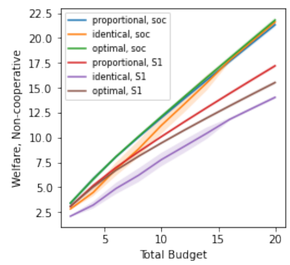


Figure 4.8: Type 2 Conflicting Groups.

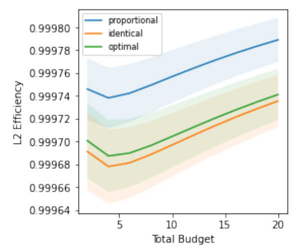


Figure 4.9: Type 2 All Positive Network.

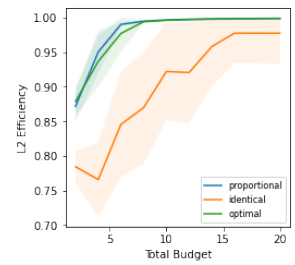


Figure 4.10: Type 2 Conflicting Groups.

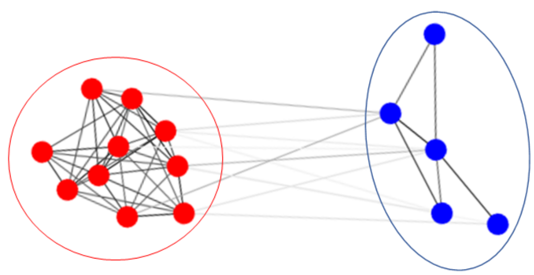


Figure 4.11: An Example of a 15 Agent, Type 1 Network.

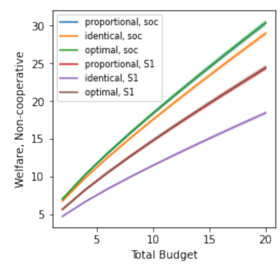


Figure 4.12: Type 3 Network Welfare.

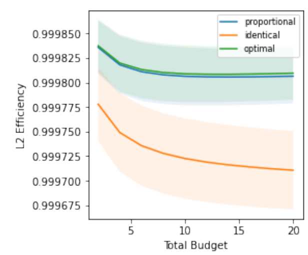


Figure 4.13: Type 3 Network Efficiency.

highest cooperative and non-cooperative welfare assigns almost all budget to S_1.

Figure 4.4 and 4.6 show another case of Type 1 network where between-group connections are all negative, but within-group connections remain positive. This can model that the type of interactions between members in the same group are different from agents in different groups. In a special case of this type of network where every agent is taking a positive action level, an increase in an agent's action level can increase (resp. decrease) the agents' utilities in the same group (resp. other groups). In this case, the proportional allocation rule is almost socially optimal.

In the Type 2 network, where all connections are positive, Figure 4.7 and 4.9 show the welfare with non-cooperative planners. Interestingly, the identical budget allocation rule is actually closer to the socially optimal allocation. For the same Type 2 network but with negative between-group connections, results shown in Figure 4.8 and 4.10 are very different from other network types since the efficiency is now significantly below 1 when we have small budgets.

Figure 4.12 and 4.13 show the results in the Type 3 network with all positive connections. In fact, for all combinations of connection signs, the trends are very similar, but the social welfare is significantly lower when we have negative connections. We also see that the proportional allocation rule is almost socially optimal.

Empirically, we observe that for all types of networks, when the budget grows larger, under any type of budget allocation rule the welfare grows approximately linearly and the efficiency approximately converges to a fixed value.

We also measured the tightness of the theoretical lower bound on the L2 efficiency. For all above introduced network types except for Type 1 with conflicting groups, the gaps between the lower bounds and the actual L2 efficiencies are less than 0.006, for all three budget allocation rules. For Type 1 network with conflicting groups, the gap is around 0.07 for all three budget allocation rules. All gaps are almost invariant in the total budget.

4.6 Chapter Conclusions

In this chapter, we studied an intervention problem in network games with community structures and multiple planners. We showed that given any intervention action, the agents will always have a unique NE. The planners can thus use backward induction and design (locally) optimal interventions. We find that for both cooperative and non-cooperative planners, the system always has a unique Stackelberg equilibrium that fully spends the budget and is Pareto efficient. We also studied the efficiency of the outcomes under different settings in this system, including whether the planners are cooperative and whether the budget is transferable both analytically and numerically. Our analysis shows that we can use the Lagrangian dual optimal variable values to characterize the efficiency, and planners have incentives to share budgets even when they are non-cooperative. The budget transferability also enables uniformly better outcomes than the non-transferable case. Empirically, we observe that the type of network determines which type of (commonly used) budget allocation rule is the most efficient.

Chapter 5

Subsidy Mechanisms for Strategic Classification and Regression Problems

5.1 Introduction

The previous chapter studies how to design interventions on multi-scale networks that induce ideal actions from the agents. The agents' actions were treated as scalars, and changes in those scalars directly impact the planners' objectives. In reality, actions by the agents may not be directly observable; what is observable may be an intermediate layer of *features* that a planner has to rely on for assessment. As a consequence, agents are motivated to take actions to improve their features in order to improve their assessment results. For example, agents can take different types of actions, and only some, e.g., *honest* efforts will influence the planner's objective, while others, e.g., *dishonest* efforts will not benefit the planner's objective but only the agents'. In other words, when an agent exerts honest effort to perform better on a measure designed by a planner, that effort improves a certain true underlying attribute that the planner cares about, while dishonest efforts only improve the proxy feature without actually improving the underlying attribute. The agents can achieve the same observable feature with multiple strategies; it is thus crucial for the planner to induce honest efforts. We leverage the strategic classification and regression framework to address this special type of incomplete information. This strategic learning framework is a Stackelberg game where the decision maker moves first by publishing a decision rule in the form of a classifier or regression function; the agents move next and simultaneously respond to the decision rules by taking actions to manipulate their features. This framework models honest and dishonest efforts in a realistic way as improvement and gaming actions, where improvement actions improve the utilities of both the agents and the decision maker, while gaming only benefits the agents.

This chapter studies the impact of adding a subsidy mechanism in strategic classification and regression problems. Conventional strategic classification and regression model the interaction between a decision maker (algorithm designer) and individuals who are subject to the decision outcomes. While the former benefits from the accuracy of its decisions, the latter may have an incentive to *game* the algorithm into making favorable but erroneous decisions. Specifically, the agents may make superficial changes to their features that lead to them receiving more desirable decisions, but these feature changes do not improve their true attributes nor their true outcomes. Recognizing the potential for such misuse, prior works tend to focus on designing an algorithm that is more robust to such strategic maneuvering, see e.g., [35, 67, 42, 15, 16, 26, 12, 20, 65]. Equally important, however, is the possibility for a mechanism designer to *incentivize* effort by the users who genuinely improve their true label; this would benefit the users and the decision maker by preserving the algorithm performance at the same time.

Toward this end, we present a strategic learning problem augmented by a subsidy mechanism (augmented strategic learning problem) modeled as a Stackelberg game between the decision maker, the mechanism designer (which could be the decision maker itself or a third party) and individuals from different demographic groups who are subject to the classifiers' decisions (the agents). The decision maker and the mechanism designer move in the first stage by publishing and committing to a decision rule (a binary classifier or a regression function) and an incentive mechanism. The published decision rule takes as input the agents' *observable* features and outputs decision outcomes that impact the agents' utilities. The agents are (simultaneous) second movers and best respond to the published decision rule and incentive mechanism. To capture the agent's ability to both game the decision rule and make real changes, we assume each agent has an endowed pre-response attribute (endowed private information), that is causal [65] to a set of observable features as well as its true label, also referred to as its *qualification status* in the context of the strategic learning problem. Here causal means these attributes directly impact the true label, precisely defined in Section 5.2.1.

An agent can exert effort to improve this causal state, thereby improving its features and its underlying attributes, or choose to game the classifier by employing non-causal schemes to improve only its features without changing its underlying attributes [65], or use a combination of them. Both choices of action, referred to as *improvement* (or honest effort) and *gaming* (or cheating, or dishonest effort), respectively, come at a cost to the agent. As pointed out in [67], gaming is much more frequently seen and studied due to its much lower cost compared to improvement. This difference in cost results in Goodhart's Law ("Once a measure becomes a target, it ceases to be a good measure" [92]), since gaming invariably degrades the performance of a classifier. The goal of this study is to see whether, beyond preventing gaming, the incentive mechanism can elicit sufficient *improvement* from the agents.

The decision maker derives its utility from the prediction accuracy, thus even a selfish decision maker may have an incentive to motivate the agents to choose improvements over gaming. When the decision maker is also the mechanism designer, one such incentive mechanism is for the decision maker to subsidize the agents' improvement costs, thereby making improvement more appealing compared to gaming. We characterize the Stackelberg equilibrium in this setting, where the decision maker determines the optimal decision rule as well as the incentive mechanism (a subsidy policy) in anticipation of the agents' best response. In addition, we also study the impact of the equilibrium classifier and incentive mechanism on the fairness and qualification status, when agents come from different demographic groups which differ in their pre-response attribute distribution (e.g., an advantaged group may have higher pre-response attributes that map to higher qualification rates and features) or action cost (e.g., an advantaged group may have lower action cost than a disadvantaged group). Alternatively, we also study the case where the mechanism designer is a third party (e.g., a government) with social well-being metrics as its objective. The third party designs a mechanism that incentivizes agents' improvement action and charges a price to the decision maker for this *improvement service* to ensure budget balance, while also making sure that incentive compatibility and individual rationality constraints are satisfied for both the agents and the decision maker. We compare the outcomes of the augmented strategic learning in these two settings, as well as the conventional strategic learning problem without an incentive mechanism, and investigate how the mechanism designer's objectives influence the fairness and qualification status.

Our work differs from previous works on incentive mechanisms in the presence of strategic agents [34, 54, 91, 42] in the following ways. Firstly, subsidies are also used in [42] for strategic classification; however, all actions considered in [42] are gaming and thus all subsidies go toward gaming. Both our work and [42] show that subsidizing gaming is a strictly dominated strategy for the decision maker, but our work further shows the potential benefit of subsidizing improvement actions. Secondly, [34, 54] use the classifier decision rule as a proxy for designing incentives, while we take a combination of the decision rule and an incentive mechanism choice to provide incentives; this is noteworthy because there are cases where the decision rule alone fails to incentivize improvement, such that one can only resort to the incentive mechanism to serve this purpose (see further discussion in Section 5.2). Thirdly, the decision maker in our model is selfish (i.e. profit maximizing) and the third party optimizes social well-being metrics (e.g., social welfare, or fairness metrics); in contrast, the decision maker is welfare maximizing in [34], is either selfish or welfare maximizing in [91], and works toward effort profiles with desired characteristics in [54]. Fourthly, while [91] focuses on the linear regression problem and [34, 54] on binary classification problems, we study both types of problems and elaborate on the similarities and differences between these setups. Finally, while strategic recourse [20] focuses on the ability of agents to systematically reverse unfavorable decisions made by algorithms (e.g., provide practical suggestions to the agents on how to alter their profile to increase their chance of receiving good decision outcomes), our work focuses on a one-shot Stackelberg game where such suggestions are implicitly given in the mechanism (e.g., the fact that an action is subsidized means such an action can improve the decision outcome). Our main contributions are as follows.

1. We formulate the problem of adding a subsidy mechanism in strategic classification and regression problems as a Stackelberg game, where the decision maker and mechanism designer commit to a classifier and an incentive mechanism, and agents follow by choosing

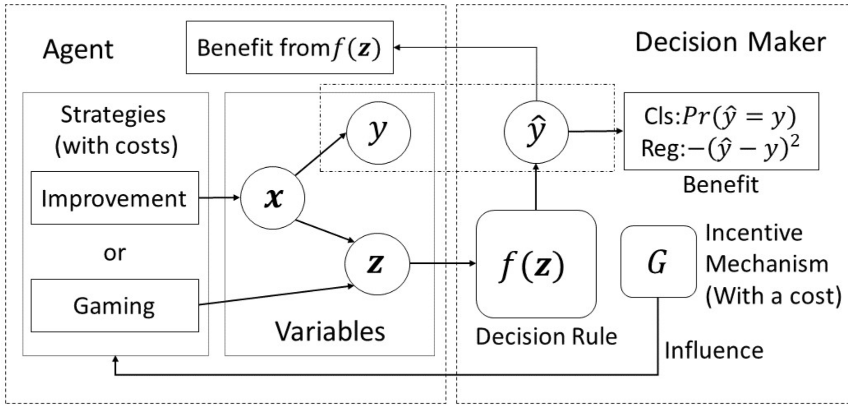


Figure 5.1: The Augmented Strategic Classification/Regression Problem.

an action to best respond (Section 5.2). This model substantially extends existing literature.

2. We begin with the setting in which the decision maker is the mechanism designer, and study the incentive mechanism design and the Stackelberg equilibrium of the classification and regression models (Sections 5.3 and 5.4). We identify conditions under which the incentive mechanisms satisfy individual rationality, incentive compatibility, and budget balance.

3. We study the social well-being of the augmented strategic learning system, focusing on both efficiency and fairness properties (Section 5.5). We also consider the case of a third party mechanism designer, and discuss its influence on these social well-being metrics (Section 5.6).

4. We illustrate our analytical findings through numerical experiments based on the FICO dataset [36] (Section 5.7).

5.2 Model

We first introduce our augmented strategic learning model. In particular, we focus on a single-round, two-stage Stackelberg game, where the decision maker and the mechanism designer move first to design, publish, and commit to a decision rule f combined with an incentive mechanism G; the agents then best respond to both the incentive mechanism and the decision rule in the second stage.

5.2.1 Attributes, Features, and Labels

An agent has an N-dimensional *pre-response attribute* $\boldsymbol{x} \in \mathcal{X}, \mathcal{X} \subseteq \mathbb{R}^N_{\geq 0}$, which is its private information. Its probability density function (pdf) is $p(\boldsymbol{x})$, which is public information. In the response phase, an agent takes an M-dimensional action $\boldsymbol{a} := (\boldsymbol{a}_+, \boldsymbol{a}_-)$, where $\boldsymbol{a}_+ \in \mathbb{R}_{\geq 0}^{M_+}$ denotes an *improvement action* profile while $\boldsymbol{a}_- \in \mathbb{R}_{\geq 0}^{M_-}$ is a *gaming action* profile, with $M_+ + M_- = M$ (with action indices ordered such that $\forall i \leq M_+$ is an improvement action).

The agent's action impacts its attributes as well as features through a *projection matrix* $P = [P_+, P_-], P \geq 0$, where $P_+ \in \mathbb{R}^{N \times M_+}$ (resp. $P_- \in \mathbb{R}^{N \times M_-}$) is the improvement (resp. gaming) projection in the following sense. The action results in the agent having a *post-response attribute* $\boldsymbol{x}' = \boldsymbol{x} + P_+\boldsymbol{a}_+ = \boldsymbol{x} + \hat{P}\boldsymbol{a}$, where $\hat{P} = [P_+, \mathbf{0}] \in \mathbb{R}^{N \times M}$, and a post-response *observable feature* (simply feature for brevity) $\boldsymbol{z} = \boldsymbol{x} + P\boldsymbol{a} = \boldsymbol{x} + P_+\boldsymbol{a}_+ + P_-\boldsymbol{a}_-$. Crucially, the post-response attribute is the agent's private information, whereas the post-response feature is observable by the decision maker. An agent's action may or may not be directly observable to the decision maker, but is anticipated given the game setting and an equilibrium concept.

This model captures the fact that improvement actions can improve an agent's underlying attribute as well as observable feature, while gaming actions only affect the outward feature without changing its underlying attribute. We can think of attributes as actual skills and features as test scores; working hard can be a type of improvement action and cheating can be a type of gaming action.

In general, the projection matrix P is not full rank, which means there are multiple choices of $\boldsymbol{a}$ for the agent to obtain the same feature $\boldsymbol{z}$ and thus the same decision outcome (next subsection).

An agent with pre- (resp. post-)response attribute $\boldsymbol{x}$ (resp. $\boldsymbol{x}'$) has a pre- (resp. post-)response *true label* y (resp. y') that indicates the quality of an agent. For strategic regression, we use the same setting as in [91]:

$$y = q(\boldsymbol{x}) := \boldsymbol{\theta}^T\boldsymbol{x} + \eta,\ y' = q(\boldsymbol{x}') = \boldsymbol{\theta}^T\boldsymbol{x}' + \eta, \tag{5.1}$$

where $\boldsymbol{\theta} \geq 0$ is the quality coefficient vector, and η is a subgaussian noise with 0 mean and variance σ. For strategic classification, $y, y' \in \{0, 1\}$, and we use a similar setting as in [42]:

$$Pr(y = 1) = l(\boldsymbol{\theta}^T\boldsymbol{x}),\ Pr(y' = 1) = l(\boldsymbol{\theta}^T\boldsymbol{x}'),\ y' \geq y, \tag{5.2}$$

where we can interpret $l : \mathbb{R} \mapsto [0, 1]$ as a likelihood function that is weakly increasing (l is a step-function in [42]). We assume that $y' \geq y$ holds for every agent, with improvement actions weakly improving the agent's true label, and gaming actions leaving it unchanged.

Remark 2. *The projection matrix P, the available action dimensions, and the quality coefficients $\boldsymbol{\theta}$ are assumed to be public information for the remainder of the chapter. We discuss in the appendix when these parameters are initially unknown to the decision maker. Parameter acquisition requires multi-round online learning [91, 37], which is different from the model setting in this chapter.*

However, we show that our incentive mechanisms can aid parameter learning in the multi-round online strategic learning models.

5.2.2 The Decision Rule

The decision rule $f : \mathbb{R}^N \mapsto \mathbb{R}$ takes as input an agent's feature $\boldsymbol{z}$ and returns a decision outcome $f(\boldsymbol{z})$. For regression, $f(\boldsymbol{z}) = \boldsymbol{w}^T\boldsymbol{z}$; for classification, $f(\boldsymbol{z}) = \mathbf{1}\{\boldsymbol{w}^T\boldsymbol{z} \geq \tau\}$, for some $\boldsymbol{w} \in \mathbb{R}^N_{\geq 0}$ (since the true labels are weakly increasing in every attribute).

5.2.3 Three Learning/Game Problems

We will consider three different strategic learning systems/game settings:

1. The *conventional strategic (CS)* problem where the agents and the decision maker play the standard Stackelberg game without any added incentive mechanism, both being fully strategic.
2. The *limited strategic (LS)* problem where the agents are fully strategic and expect the decision maker to be strategic, but the latter does not anticipate the agents' strategic behavior and applies the optimal non-strategic decision rule, e.g., $f(\boldsymbol{z}) = \boldsymbol{\theta}^T\boldsymbol{z}$ in regression, as a sub-optimal option.[1]
3. The *augmented strategic (AS)* problem, where the agents and the decision maker play the Stackelberg game with a subsidy mechanism.

We use the CS and LS problems as benchmarks to show how subsidy mechanisms influence the equilibrium system outcome. We next detail the utility functions and the incentive mechanism.

5.2.4 Utilities and Optimal Strategies in Conventional & Limited Strategic Learning

In a conventional strategic learning problem, it is assumed that an agent has the following utility function $u_C(\boldsymbol{x}, \boldsymbol{a}) = f(\boldsymbol{x} + P\boldsymbol{a}) - h(\boldsymbol{a})$, where the agent benefits from the decision outcome $f(\boldsymbol{z})$ and incurs a cost of $h(\boldsymbol{a}) := \boldsymbol{c}^T\boldsymbol{a}$.

Denote by $\boldsymbol{a}^*_C(\boldsymbol{x}) := \arg\max_{\boldsymbol{a}} u_C(\boldsymbol{x}, \boldsymbol{a})$ the agent's *conventional best response* or *CS best response*, with ties broken in favor of its qualification status $\boldsymbol{\theta}^T\boldsymbol{x}'$. In the same problem, denote y'_C as the *CS post-response label*. The decision maker's utility is

$$U_C^{(cls)}(f) = \int_{\mathcal{X}} Pr\big(f(\boldsymbol{x} + P\boldsymbol{a}^*_C(\boldsymbol{x})) = y'_C\big)p(\boldsymbol{x})d\boldsymbol{x};$$
$$U_C^{(reg)}(f) = \int_{\mathcal{X}} \mathbb{E}_\eta\big[-\big(f(\boldsymbol{x} + P\boldsymbol{a}^*_C(\boldsymbol{x})) - y'_C\big)^2\big]p(\boldsymbol{x})d\boldsymbol{x} \tag{5.3}$$

[1]The agents in LS behave the same as in CS problems. One reason to consider LS is that the CS problem is in general NP-hard for the decision maker [54].

for strategic classification and regression, respectively. Here the decision maker aims to maximize the classification accuracy and minimize the mean squared error in regression, respectively. We will use $f_C^* := \arg\max_f U_C(f)$ to denote the decision maker's optimal conventional strategic decision rule, where the type of problem (*cls* vs. *reg*) will be clear from the context. In the limited strategic (LS) problem, the agents' utilities and best responses are the same as the CS problem, but the decision maker instead maximizes, respectively:

$$\begin{aligned} U_L^{(cls)}(f) &= \int_{\mathcal{X}} Pr\big(f(\boldsymbol{x}) = y\big)p(\boldsymbol{x})d\boldsymbol{x}; \\ U_L^{(reg)}(f) &= \int_{\mathcal{X}} \mathbb{E}_\eta\big[-\big(f(\boldsymbol{x}) - y\big)^2\big]p(\boldsymbol{x})d\boldsymbol{x}. \end{aligned} \tag{5.4}$$

Remark 3. *Our findings generalize to other cost functions such as L2 cost $h(\boldsymbol{a}) = \sqrt{\boldsymbol{a}^T C \boldsymbol{a}}$ or quadratic cost $h(\boldsymbol{a}) = \frac{1}{2}||\boldsymbol{a}||_2^2$. More details are provided in the appendix.*

5.2.5 Incentive Mechanisms and Utilities in Augmented Strategic Learning

Different from previous works, we focus on how an incentive mechanism can influence the strategic interaction between the decision maker and the agents. We consider two types of mechanism providers. We will start with the setting where the mechanism provider is the decision maker itself. Our analysis and results are then extended in Section 5.6 to a second setting where the mechanism is provided or implemented by a *third-party* organization, e.g., the government.

We focus on *discount mechanisms* that are based on providing a *discount on actions*, where the mechanism provider has the ability to lower the cost of agents' actions, e.g., making the cost of getting tutoring or exam preparation cheaper during the school admission process.[2] We use G to denote the discount mechanism where the designer chooses a *rate discount* value on each action dimension $\triangle \boldsymbol{c} = (\triangle c_i)_{i=1}^M, \triangle c_i < c_i$, and set a *discount amount range* $[\underline{c}, \overline{c}]$. Then with G, the agent's utility function in the augmented strategic learning problem becomes

$$u_A(\boldsymbol{x}, \boldsymbol{a}) = f(\boldsymbol{x} + P\boldsymbol{a}) - h_A(\boldsymbol{a}), \text{ where } h_A(\boldsymbol{a}) = h(\boldsymbol{a}) - \triangle \boldsymbol{c}^T \boldsymbol{a} \cdot \mathbf{1}\{\triangle \boldsymbol{c}^T \boldsymbol{a} \in [\underline{c}, \overline{c}]\}. \tag{5.5}$$

With G, $\boldsymbol{a}_A^*(\boldsymbol{x}) := \arg\max_{\boldsymbol{a}} u_A(\boldsymbol{x}, \boldsymbol{a})$ denotes the agent's *augmented best response* or *AS best response*, with ties broken in favor of maximizing $\boldsymbol{\theta}^T \boldsymbol{x}'$ unless otherwise suggested by the mechanism designer. The designer incurs a *subsidy cost*

$$H(G) = \int_{\mathcal{X}} \triangle \boldsymbol{c}^T \boldsymbol{a}_A^*(\boldsymbol{x}) \cdot \mathbf{1}\{\triangle \boldsymbol{c}^T \boldsymbol{a}_A^*(\boldsymbol{x}) \in [\underline{c}, \overline{c}]\}p(\boldsymbol{x})d\boldsymbol{x}. \tag{5.6}$$

[2] In the appendix, we discuss an alternative mechanism where the designer cannot change the action cost, and show that the resulting mechanism design problem is computationally intractable.

Denote by y'_A the *AS post-response label*. The augmented utility of the decision maker is then:

$$U_A^{(cls)}(f) = \int_{\mathcal{X}} Pr\big(f(\boldsymbol{x} + P\boldsymbol{a}_A^*(\boldsymbol{x})) = y'_A\big)p(\boldsymbol{x})d\boldsymbol{x} - H(G);$$
$$U_A^{(reg)}(f) = \int_{\mathcal{X}} \mathbb{E}_\eta\big[-\big(f(\boldsymbol{x} + P\boldsymbol{a}_A^*(\boldsymbol{x})) - y'_A\big)^2\big]p(\boldsymbol{x})d\boldsymbol{x} - H(G), \tag{5.7}$$

for the classification and regression problems, respectively. In designing G, we will consider three commonly studied properties in the mechanism design literature:

1. Individual rationality (IR): The agents are better off participating in the mechanism than not.
2. Incentive compatibility (IC): The agents act in self-interest.
3. Budget balance (BB): This only applies to the third party mechanism; see Section 5.6.

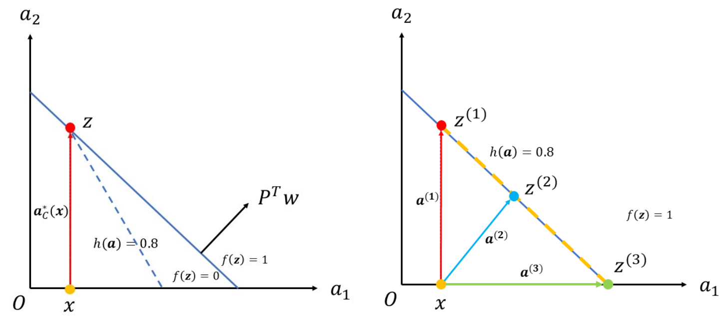
Here we elaborate on our definition of IC. Conventionally, a mechanism is IC if every participant achieves the best outcome by acting according to their true preferences. In our case, the mechanism and the decision rule are *fixed*, and every agent's response to the mechanism and the decision rule is *independent of other agents*. Therefore, every agent only needs to choose the optimal action based on its (subsidized) utility function, and the IC condition is straightforward. The authors in [99] provided another definition of IC classifiers, where the classifier is IC if no manipulation can strictly improve an agent's utility. We consider a different problem setting from [99] but similar to [42, 54, 91, 67], where manipulation can always strictly improve some agents' utilities. In our problem, we encourage agents to take improvement actions over manipulations (instead of trying to stop all manipulations), which benefits all entities in the game and improves the social qualification status. The difference in the definitions of IC found in our work and in [99] is due to very different problem settings.

5.3 Augmented Strategic Classification

In this and the next section, we consider agents from a single demographic group. Throughout our analysis, we will provide pictorial interpretations of our results, using an example with 2 action dimensions: a_1 is an improvement action and a_2 is a gaming action.

We begin with some preliminaries. The next two lemmas characterize the magnitude and direction of the agents' best responses $\boldsymbol{a}_t^*(\boldsymbol{x})$ ($t \in \{C, A\}$) in the conventional and augmented strategic games.

Lemma 2. *For CS and AS classifications,* $\boldsymbol{w}^T(\boldsymbol{x} + P\boldsymbol{a}_t^*(\boldsymbol{x})) = \tau \Leftrightarrow \boldsymbol{a}_t^*(\boldsymbol{x}) \neq \boldsymbol{0}, \forall t.$



(a) CS Best Response (b) Tie Breaking

Figure 5.2: An Illustration of a CS Best Response in Classification, where $P = [1, 1]$, $w = 1$, $P^T w = (1, 1)$. The Solid Blue Line is the Decision Boundary. In (a), the Blue Dashed Line is an Equal Cost Contour; $c_2 < c_1$, and thus Gaming is Cheaper than Improving, Leading to the Best Response Shown in Red. (b) Illustrates Tie Breaking in Best Responses, where $c_1 = c_2$, with the Equal Cost Contour Shown with the Yellow Dashed Line.

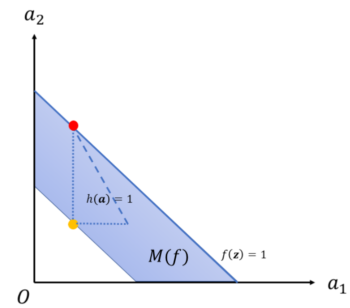


Figure 5.3: An Illustration of the Manipulation Margin in Classification, Given by the Shaded Region; Every Agent Inside can Reach the Boundary with an Action Cost of No More than 1.

Proof. For $\forall \boldsymbol{a}$ such that $\boldsymbol{w}^T(\boldsymbol{x} + P\boldsymbol{a}) < \tau$, $f(\boldsymbol{z}) = 0$; thus it is dominated by $\mathbf{0}$ due to $h(\boldsymbol{a}) \geq h(\mathbf{0}) = 0$ and $h_A(\boldsymbol{a}) \geq h_A(\mathbf{0}) = 0$. On the other hand, for $\forall \boldsymbol{a}$ such that $\boldsymbol{w}^T(\boldsymbol{x} + \boldsymbol{a}^*) > \tau$, there exists an $\alpha \in (0, 1)$ such that $\boldsymbol{w}^T(\boldsymbol{x} + \alpha P\boldsymbol{a}) = \tau$. Both $\boldsymbol{a}$ and $\alpha\boldsymbol{a}$ guarantee $f(\boldsymbol{z}) = 1$, and thus $\boldsymbol{a}$ is dominated by $\alpha\boldsymbol{a}$ due to $h(\boldsymbol{a}) > h(\alpha\boldsymbol{a})$ and $h_A(\boldsymbol{a}) > h_A(\alpha\boldsymbol{a})$ if $\boldsymbol{a} \neq \mathbf{0}$. □

Lemma 2 describes the magnitude of the best response in CS and AS classification: it is such that the feature $\boldsymbol{z}$ reaches the decision boundary but not beyond, as going beyond the boundary only increases the cost without affecting the decision. This is illustrated by the red arrow in Figure 5.2a.

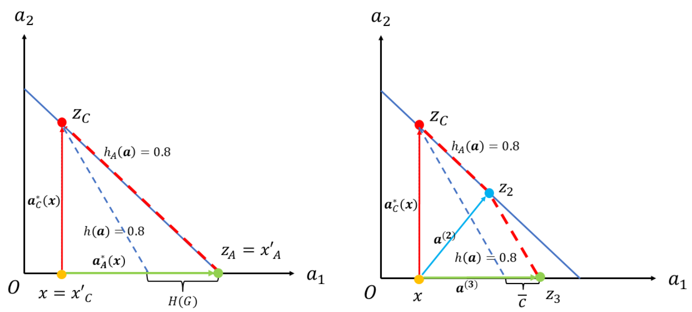
Lemma 3. *For CS and AS classification,*

$$(\boldsymbol{a}_C^*(\boldsymbol{x}))_i \geq 0, \ \textit{if}\ i \in \{\arg\max_j (P^T\boldsymbol{w})_j/c_j\}; \ \ (\boldsymbol{a}_t^*(\boldsymbol{x}))_i = 0, \ \textit{o.w.}, \ \ \forall \boldsymbol{x}.$$

$$(\boldsymbol{a}_A^*(\boldsymbol{x}))_i \geq 0, \ \textit{if}\ i \in \{\arg\max_j (P^T\boldsymbol{w})_j/(c_j - \triangle c_j)\}; \ \ (\boldsymbol{a}_A^*(\boldsymbol{x}))_i = 0, \ \textit{o.w.}, \ \ \forall \boldsymbol{x}. \tag{5.8}$$

Proof. Assume by contradiction $a_j^* > 0, j \neq i_C = \arg\max_k \frac{(P^T\boldsymbol{w})_k}{c_k}$. By Lemma 2, as $\boldsymbol{a}^* \neq \mathbf{0}$ we have $\boldsymbol{w}^T(\boldsymbol{x} + P\boldsymbol{a}^*) = \tau$. Denote $\tilde{\boldsymbol{a}} = \boldsymbol{a}^* - a_j^*\boldsymbol{e}_j + \frac{a_j^*(P^T\boldsymbol{w})_j}{(P^T\boldsymbol{w})_{i_C}}\boldsymbol{e}_{i_C}$, where $\boldsymbol{e}_i$ is the i-th orthonormal base vector of $\mathbb{R}^M$. It is easy to see that $\boldsymbol{w}^T(\boldsymbol{x} + P\tilde{\boldsymbol{a}}) = \tau$ and thus $f(\boldsymbol{z}) = 1$, while $h(\tilde{\boldsymbol{a}}) < h(\boldsymbol{a}^*)$, indicating that $\tilde{\boldsymbol{a}}$ achieves a higher utility than $\boldsymbol{a}^*$, contradicting the optimality of $\boldsymbol{a}^*$. The proof for AS classification is similar. □

Lemma 3 describes the directional properties of the best response: the agent will invest in the action dimension(s) with the highest *return on investment* $(P^T\boldsymbol{w})_j/c_j$ (in CS) or $(P^T\boldsymbol{w})_j/(c_j - \triangle c_j)$



(a) Discount Mechanism (b) Designer's Suggestion

Figure 5.4: An Illustration of the Discount Mechanism in Classification, $P = [1, 1], w = 1, P^T w = (1, 1), c_2 < c_1$, the Red Dashed Line is the Discounted Equal Cost Contour with a Minimum Effective Discount. In Figure 5.4b, the $\bar{c}$ is of a Smaller Value, and the Equal Cost Contour has a Different Shape. The Decision Maker Suggests the Agents Choose $\boldsymbol{a}_C^*(\boldsymbol{x})$ Instead of $\boldsymbol{a}^{(3)}$ in Tie Breaking in Algorithm 6 and 7 When l is Convex.

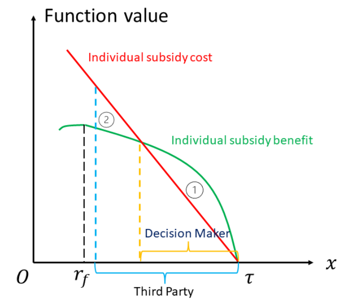


Figure 5.5: A Simplified Illustration of the Individual Subsidy Benefit and Cost in the Mechanism. Region 1 (Resp. 2) Corresponds to Agents with Subsidy Surplus (Resp. Deficit). The Third Party (Section 5.6) Incentivizes Region 2 Agents for Social Well-being Objectives. r_f Represents the Lower Boundary of $\mathcal{M}(f)$.

(in AS). Without loss of generality, we assume that the optimal CS action dimension $i_C := \arg\max_j (P^T \boldsymbol{w})_j / c_j$ is unique. This property is also shown in Figure 5.2a, where $i_C = 2$ is the action with the highest return on investment.

We note that there may be multiple actions that are equal in their return on investment. In such cases, we assume the agent follows the algorithm designer's recommendation if any, and otherwise chooses the one that leads to the maximum improvement (i.e., the one maximizing $\boldsymbol{\theta}^T \hat{P} \boldsymbol{a}$). Figure 5.2b explains this tie breaking: here $c_1 = c_2$ and every point on the yellow contour has equal cost and benefit to the agent, making the agent indifferent between $\boldsymbol{a}^{(1)}, \boldsymbol{a}^{(2)}, \boldsymbol{a}^{(3)}$. We take $\boldsymbol{a}^{(3)}$, the largest improvement, to be the agent's choice.

Using Lemmas 2 and 3, we have

$$\boldsymbol{a}_C^*(\boldsymbol{x}) = \frac{\tau - \boldsymbol{w}^T \boldsymbol{x}}{(P^T \boldsymbol{w})_{i_C}} \boldsymbol{e}_{i_C}, \text{ if } \boldsymbol{x} \in \mathcal{M}(f); \qquad \boldsymbol{a}_C^*(\boldsymbol{x}) = \boldsymbol{0}, \text{ o.w.,} \tag{5.9}$$

where $\mathcal{M}(f) := \left\{ \boldsymbol{x} \middle| \frac{(\tau - \boldsymbol{w}^T \boldsymbol{x}) c_{i_C}}{(P^T \boldsymbol{w})_{i_C}} \in (0, 1] \right\}$ denotes the *manipulation margin* of f: every agent in the manipulation margin has a non-zero best response to improve their decision outcome to 1. This is illustrated in Figure 5.3.

In classification, if $i_C \leq M_+$, we say the decision rule *incentivizes improvement*, otherwise we say the decision rule *incentivizes gaming*. The theorem below shows conditions under which it is impossible for the decision maker to have a decision rule that incentivizes improvement; the proof is given in Appendix D.2.2.

Theorem 13. *Let κ_i denote the substitutability of action dimension i [54, 50]. Formally,*

$$\kappa_i := \min_{\boldsymbol{a}\in\mathbb{R}^M, \boldsymbol{a}\geq 0} \frac{\boldsymbol{c}^T\boldsymbol{a}}{c_i}, \quad \text{s.t. } P\boldsymbol{a} - \boldsymbol{p}_i \geq 0, \tag{5.10}$$

where $\boldsymbol{p}_i$ is the i-th column of P. If $\kappa_i = 1$, then there exists a $\boldsymbol{w}$ that can incentivize action dimension i, and the $\boldsymbol{w}$ can be found in polynomial time. On the other hand, if $\kappa_i < 1, \forall i \leq M_+$, then there always exist linear combinations of gaming actions that weakly dominate every improvement action, in which case there is no f that can incentivize improvement, and the decision maker's CS optimal strategy f_C^* satisfies $\boldsymbol{w} = \boldsymbol{\theta}$.

We next consider designing an incentive mechanism, with the decision rule f treated as given.

Lemma 4. *To induce an agent to take an AS best response with non-zero investment in action dimension $j \leq M_+$, i.e., $[\boldsymbol{a}_A^*(\boldsymbol{x})]_j > 0$, the discount value $\triangle c_j$ should satisfy $(P^T\boldsymbol{w})_j/(c_j - \triangle c_j) \geq (P^T\boldsymbol{w})_{i_C}/(c_{i_C})$, i.e., $\triangle c_j \geq c_j - \frac{(P^T\boldsymbol{w})_j}{(P^T\boldsymbol{w})_{i_C}} c_{i_C}$.*

Based on Lemma 4, we denote the *minimum effective discount value* as

$$\triangle c_j^* := c_j - \frac{(P^T\boldsymbol{w})_j}{(P^T\boldsymbol{w})_{i_C}} c_{i_C}. \tag{5.11}$$

Intuitively, Lemma 4 states that to induce a best response in action j, the discount has to make j the action with the highest (potentially tied) return on investment. Figure 5.4a illustrates an example of how the discount mechanism with minimum effective discount value works. By choosing $\triangle c_1 = \triangle c_1^*$, the two actions have the same return on investment; the agents choose $\boldsymbol{a}_A^*(\boldsymbol{x})$, the maximum improvement action, in this case. In contrast, the CS action $\boldsymbol{a}_C^*(\boldsymbol{x})$ is a gaming action.

Before we move on to the optimal mechanism design, we define the *subsidy surplus*.

Definition 9. *In classification the subsidy surplus is*

$$S(f,G) = \int_{\mathcal{X}} \big[Pr(f(\boldsymbol{x}+P\boldsymbol{a}_A^*(\boldsymbol{x})) = y'_A) - Pr(f(\boldsymbol{x}+P\boldsymbol{a}_C^*(\boldsymbol{x})) = y'_C)\big] p(\boldsymbol{x}) d\boldsymbol{x} - H(G), \tag{5.12}$$

where y'_t denotes the post-response label such that $Pr(y'_t = 1) = l(\boldsymbol{x} + \hat{P}\boldsymbol{a}_t^*(\boldsymbol{x})), \forall t \in \{C, A\}$.

The integral part in $S(f,G)$ is the *benefit gain* of the decision maker and the value in the square bracket is the *individual subsidy benefit*. The decision maker's problem is equivalent to maximizing $S(f,G)$ under IC and IR.

Theorem 14. *For general $f(\boldsymbol{z}) = \mathbf{1}\{\boldsymbol{w}^T\boldsymbol{z} \geq \tau\}$, p, and l functions, finding the optimal IC and IR discount mechanism requires solving non-convex optimization problems and thus is NP-hard.*

While finding the optimal mechanism under IC and IR constraints is NP-hard, we can develop an efficient algorithm (Algorithm 1) for a special case when the likelihood function l is convex.

Theorem 15. *Algorithm 6 runs in polynomial time, and if l is convex on $[0, \max_{\boldsymbol{x}:\boldsymbol{w}^T\boldsymbol{x}=\tau} l(\boldsymbol{x})]$, then any $G \neq 0$ returned by Algorithm 6 is IC, IR, and satisfies $S(f, G) \geq 0$.*

Algorithm 6: Find a $G \neq 0$ that is IC, IR and $S(f, G) > 0$ for Classification

---

$\boldsymbol{x}_1 \leftarrow \arg\min_{\boldsymbol{x}:\boldsymbol{w}^T\boldsymbol{x}=\tau} \boldsymbol{\theta}^T\boldsymbol{x}$;
$i_C \leftarrow \arg\max_j (P^T\boldsymbol{w})_j / c_j$;
for $j = 1 : M_+$ **do**
  $\triangle\boldsymbol{c} \leftarrow \mathbf{0}$; $\overline{c} \leftarrow 0$; $l_+ \leftarrow 0$;
  $\triangle c_j \leftarrow c_j - \frac{(P^T\boldsymbol{w})_j}{(P^T\boldsymbol{w})_{i_C}} c_{i_C}$;
  Define function
  $\boldsymbol{a}(\delta) := \delta\boldsymbol{e}_j - \delta\frac{(P^T\boldsymbol{w})_j}{(P^T\boldsymbol{w})_{i_C}}\boldsymbol{e}_{i_C}$;
  $l_+(\delta) := l(\boldsymbol{\theta}^T\boldsymbol{x}_1) - l(\boldsymbol{\theta}^T\boldsymbol{x}_1 - \boldsymbol{a}(\delta))$;
  $\delta^* \leftarrow \arg\max_\delta \; s.t. \; l_+(\delta) \geq \delta\triangle c_j$;
  if $\delta^* = 0$ **then**
    Go back to for loop
  $\overline{c} \leftarrow \min\{\delta^*, 1/(c_j - \triangle c_j)\} \cdot \triangle c_j$;
  Return $(\triangle\boldsymbol{c}, 0, \overline{c})$
Return $(\mathbf{0}, 0, 0)$

Algorithm 7: A G that is IC, IR and $S(f, G) \geq 0$ for Classification when $\boldsymbol{w} = \boldsymbol{\theta}$

$i_A \leftarrow \arg\max_{j \leq M_+} (P^T\boldsymbol{\theta})_j / c_j$;
$\triangle\boldsymbol{c} \leftarrow \mathbf{0}$; $\triangle c_{i_A} \leftarrow c_{i_A} - \frac{(P^T\boldsymbol{\theta})_{i_A}}{(P^T\boldsymbol{\theta})_{i_C}} c_{i_C}$;
Define functions
$s_1(r) := l(\tau) - l(r) - \frac{(\tau - r)\triangle c_{i_A}}{(P^T\boldsymbol{\theta})_{i_A}}$;
$s_2(r) := l(\tau) + l(r) - 1 - \frac{(\tau - r)\triangle c_{i_A}}{(P^T\boldsymbol{\theta})_{i_A}}$;
$r \leftarrow \arg\min_r \; s.t. \; s_1(r) \geq 0$;
if $l(r) < 0.5$ **then**
  $r \leftarrow \arg\min_r \; s.t. \; s_2(r) \geq 0$;
$\overline{c} = (\tau - r)\triangle c_{i_A} / (P^T\boldsymbol{\theta})_{i_A}$;
Return $(\triangle\boldsymbol{c}, 0, \overline{c})$.

From Algorithm 6 we see that the decision maker prefers subsidizing agents that are "closer" to the boundary when l is convex on $[0, \max_{\boldsymbol{x}:\boldsymbol{w}^T\boldsymbol{x}=\tau} l(\boldsymbol{x})]$. This is because when l is convex, the subsidy benefit becomes concave while the subsidy cost is linear in the "distance to the boundary"; thus the agents close enough to the boundary can have positive individual subsidy surplus; Figure 5.5 provides an illustration of this.

The convexity requirement of l on a low range is satisfied in real-world datasets such as the FICO credit score dataset, in which the likelihood function l frequently has an S-shape (see Section 5.7). We discuss the case of other likelihood function types (including concave) in the appendix. Also note that in Algorithm 6 the mechanism designer places a discount on only one dimension. This is because even though it technically can set the discount $\triangle c_i > 0$ for multiple improvement actions, ultimately the agent either finds the dimension with the highest return on investment or breaks ties in favor of the largest improvement.[3]

The optimal mechanism can be found more efficiently for the special case when $\boldsymbol{w} = \boldsymbol{\theta}$ in f (this happens, e.g., in the optimal LS strategy as shown in Lemma 6 in the appendix, or in the optimal CS strategy when $\kappa_i < 1, \forall i \leq M_+$ in Theorem 13). This can be done in a fixed number of steps (faster than polynomial) using Algorithm 7.

Theorem 16. *If $\boldsymbol{w} = \boldsymbol{\theta}$ in f, f incentivizes gaming, and l is convex on $[0, \tau]$, then Algorithm 7 finds a G that is IC, IR, and satisfies $S(f, G) \geq 0$. In addition, algorithm 7 finds the optimal G if $l(\tau) - l(\underline{r}_f) \leq \frac{(\tau - \underline{r}_f)\triangle c^*_{i_A}}{(P^T\boldsymbol{\theta})_{i_A}}$, where $\underline{r}_f = \min_{\boldsymbol{x} \in \mathcal{M}(f)} \boldsymbol{\theta}^T \boldsymbol{x}$.*

Intuitively, the condition $l(\tau) - l(\underline{r}_f) \leq \frac{(\tau - \underline{r}_f)\triangle c^*_{i_A}}{(P^T\boldsymbol{\theta})_{i_A}}$ indicates the subsidy cost is larger than the subsidy gain for an agent on the "far side" boundary of $\mathcal{M}(f)$ in (5.9). This holds when improvement costs are much larger than gaming costs, so that the discount payment is higher than the resulting benefit from the agent's improvement. Such a condition is needed to enable the efficient calculation of the optimal mechanism for the following reason. If the condition does not hold, the mechanism can further increase the cost discount rate on the actions and let agents with a pre-response attribute such that $\boldsymbol{\theta}^T\boldsymbol{x} < \underline{r}_f$ to also take improvement actions. However, this would again make the problem hard for the decision maker, since it has to jointly optimize $\triangle c_j$ and $\bar{c}$, and such optimization is non-convex.

We note that the s_1 and s_2 functions in Algorithm 7 capture the following properties of individual subsidy surplus: for agents in $\mathcal{M}(f)$, these agents' qualification status improvement equals the individual subsidy benefit $l(\boldsymbol{\theta}^T\boldsymbol{x}'_A) - l(\boldsymbol{\theta}^T\boldsymbol{x}'_C)$, but for agents not in $\mathcal{M}(f)$, the individual subsidy benefit is not the qualification status improvement, but instead $l(\boldsymbol{\theta}^T\boldsymbol{x}'_A) - [1 - l(\boldsymbol{\theta}^T\boldsymbol{x}'_C)]$ since these agents are supposed to receive 0 decision outcomes (rejections) in the CS problem. The green curve in Figure 5.5 also illustrates the above.

5.4 Augmented Strategic Regression

We now turn to the strategic regression problem. For CS and AS regression, the best response directions are the same as CS and AS classification, as given in Lemma 3.

[3] When placing discounts on multiple actions, finding the optimal tie-breaking rule is a non-convex problem.

However, different from the strategic classification problem, the agents can have best responses with infinite magnitude. For example, if $(P^T\boldsymbol{w})_{i_C} \geq c_{i_C}$, the agent will invest an infinite amount in action i_C. To handle this issue, we assume that the agents' actions are bounded by an action budget $h(\boldsymbol{a}) \leq B$ in CS (and LS) regression, and $h_A(\boldsymbol{a}) \leq B$ in AS regression.[4]

Given these bounds on the agents' budgets, the agents' best responses can be characterized as follows: if $(P^T\boldsymbol{w})_{i_C} \geq c_{i_C}$, then $\boldsymbol{a}_C^*(\boldsymbol{x}) = \frac{B}{c_{i_C}}\boldsymbol{e}_{i_C}$; otherwise $\boldsymbol{a}_C^*(\boldsymbol{x}) = \boldsymbol{0}$. Similarly, let $i_A = \arg\max_j (P^T\boldsymbol{w})_j/(c_j - \triangle c_j)$, if $(P^T\boldsymbol{w})_{i_A} \geq c_{i_A} - \triangle c_{i_A}$. Then, the AS-discount best response is $\boldsymbol{a}_A^*(\boldsymbol{x}) = \frac{B}{c_{i_A} - \triangle c_{i_A}}\boldsymbol{e}_{i_A}$; otherwise $\boldsymbol{a}_A^*(\boldsymbol{x}) = \boldsymbol{0}$.

An interesting difference to highlight is that the agents' best responses in strategic classification depend on both the pre-response attributes of the agents and the decision rule, whereas in strategic regression, the best responses are the same for all agents and only depend on the decision rule.

In this strategic regression setting, we will say f incentivizes *0 responses* if $\boldsymbol{a}_C^*(\boldsymbol{x}) = \boldsymbol{0}$. Otherwise, if $i_C \leq M_+$ (resp. $i_C > M_+$), we say f incentivizes improvement (resp. gaming).

If f incentivizes non-zero responses (improvement or gaming), the cost discount rates will again follow Lemma 4, with the minimum effective discount rate still the same as in (5.11); otherwise, the minimum effective cost discount rate on action j will be such that $(P^T\boldsymbol{w})_j = (c_j - \triangle c_j)$, $\triangle c_j^* = \max\{c_j - c_{i_C}(P^T\boldsymbol{w})_j/(P^T\boldsymbol{w})_{i_C}, c_j - (P^T\boldsymbol{w})_j\}$.

Using this, the error incurred by the designer on an agent with pre-response attributes $\boldsymbol{x}$ will consist of two parts, an *equilibrium coefficient error* and an inevitable error due to noises,

$$\mathcal{E}(f, \boldsymbol{a}, \boldsymbol{x}) = [\boldsymbol{w}^T(\boldsymbol{x} + P\boldsymbol{a}) - \boldsymbol{\theta}^T(\boldsymbol{x} + \hat{P}\boldsymbol{a})]^2 + err(\eta). \tag{5.13}$$

Note that although the agents' best responses are independent of $\boldsymbol{x}$, the equilibrium individual errors depend on $\boldsymbol{x}$ for any $\boldsymbol{w} \neq \boldsymbol{\theta}$.

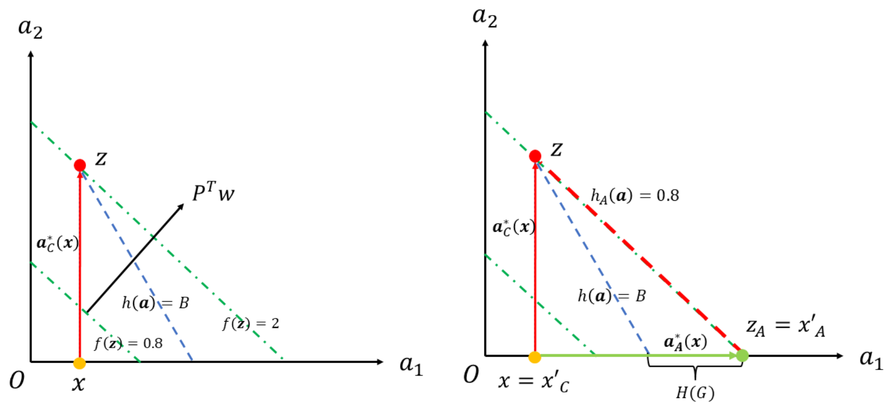
We next consider the problem of designing an incentive (discount) mechanism.

Theorem 17. *For general $f(\boldsymbol{z}) = \boldsymbol{w}^T\boldsymbol{z}$ and $p(\boldsymbol{x})$, finding the optimal IC, IR, and discount mechanism requires solving non-convex optimization problems and thus is NP-hard.*

The difficulty of designing incentive mechanisms for strategic regression problems stems from the fact that the equilibrium individual errors depend on $\boldsymbol{x}$ and thus the overall prediction error depends largely on $p(\boldsymbol{x})$. Moreover, the individual equilibrium error is not monotone in any action dimension for a general $\boldsymbol{w} \neq \boldsymbol{\theta}$. As a result, we can not follow the same methods used in the strategic classification setting to find sufficient conditions that simplify the search for the optimal mechanism.

However, the mechanism designer can now leverage the fact that the agents have identical best

[4]Such bound was not needed in the classification setting, as the fact that $f(\boldsymbol{z}) \leq 1$ naturally provided this.



(a) CS Best Response

(b) Discount Mechanism

Figure 5.6: An Illustration of the CS Best Response and the Discount Mechanism in Regression, where the Green Dashed Lines are Equal Decision Outcome Contours, $P = [1, 1], w = 1, P^T w = (1, 1), c_2 < c_1$, the Red Dashed Line is the Discounted Equal Cost Contour with a Minimum Effective Discount.

responses to facilitate the search for IC and IR discount mechanisms that satisfy $S(f, G) \geq 0$, as shown in the following theorem.

Theorem 18. *Suppose the computation of integration $\int_{\mathcal{X}} \mathcal{E}(f, \boldsymbol{a}, \boldsymbol{x}) p(\boldsymbol{x}) d\boldsymbol{x}, \forall \boldsymbol{a}$ can be done in finite time. Then, Algorithm 8 runs in polynomial time and any $G \neq 0$ it returns is IC, IR and satisfies $S(f, G) > 0$.*

The finite computation time assumption is met, for example, when the distribution $\mathcal{X}$ is discrete or when $p(\boldsymbol{x})$ is uniform.

If f incentivizes non-zero responses, then Algorithm 8 sets $\triangle c_j$ at the minimum effective discount value, and sets no discount on other actions. Then, it chooses $\underline{c} = 0, \overline{c} = \frac{\alpha B \triangle c_j}{c_j - \triangle c_j}$ so that it incentivizes all agents to take an AS best response $\boldsymbol{a}_A^*(\boldsymbol{x}) = \alpha \frac{B}{c_j - \triangle c_j} \boldsymbol{e}_j + (1 - \alpha) \frac{B}{c_{i_C}} \boldsymbol{e}_{i_C}$.[5] If f incentivizes 0 responses, then the decision maker can choose $\triangle c_j = c_j - (P^T \boldsymbol{w})_j$ and set $\overline{c} = \alpha B$ in Algorithm 8 so that $\boldsymbol{a}_A^*(\boldsymbol{x}) = \alpha \boldsymbol{e}_j$.

[5]Similar to the classification setting, we let the algorithm put a discount on one action dimension. Any $\underline{c} \leq \overline{c}$ is equivalent to both the agents and the designer here since the agent will by default use the discount amount $\overline{c}$ for the maximum improvement. The algorithm can return on condition $S > 0$ as well.

Algorithm 8: Grid Search an IC, IR and $S(f, G) > 0$ Mechanism for Regression

Choose $\epsilon > 0$;
$\boldsymbol{a}_C \leftarrow \frac{B}{c_{i_C}} \boldsymbol{e}_{i_C}$; $S_{max} \leftarrow 0$; $ans \leftarrow (\mathbf{0}, [0, 0])$;
$E_C \leftarrow \int_{\mathcal{X}} \mathcal{E}(f, \boldsymbol{a}_C, \boldsymbol{x}) p(\boldsymbol{x}) d\boldsymbol{x}$;
for $j = 1 : M_+$ **do**
  $\triangle \boldsymbol{c} \leftarrow \mathbf{0}$; $S \leftarrow 0$; $\alpha \leftarrow \epsilon$;
  $\triangle c_j \leftarrow c_j - \frac{(P^T \boldsymbol{w})_j}{(P^T \boldsymbol{w})_{i_C}} c_{i_C}$;
  while $S \geq 0$ **do**
    $\alpha \leftarrow \alpha + \epsilon$; $\bar{c} = \frac{\alpha B \triangle c_j}{c_j - \triangle c_j}$;
    $\boldsymbol{a}_A = \alpha \frac{B}{c_j - \triangle c_j} \boldsymbol{e}_j + (1 - \alpha) \frac{B}{c_{i_C}} \boldsymbol{e}_{i_C}$;
    $E_A \leftarrow \int_{\mathcal{X}} \mathcal{E}(f, \boldsymbol{a}_A, \boldsymbol{x}) p(\boldsymbol{x}) d\boldsymbol{x}$;
    $S \leftarrow E_C - E_A - \bar{c}$;
  if $S > S_{max}$ **then**
    $S_{max} \leftarrow S$; $ans \leftarrow (\triangle \boldsymbol{c}, [0, \bar{c}])$;
Return ans.

Below, we also discuss the cases when $\boldsymbol{w} = \boldsymbol{\theta}$, e.g., the decision maker's optimal LS strategy $f_L^*(\boldsymbol{z}) = \boldsymbol{\theta}^T \boldsymbol{z}$.[6]

Lemma 5. *If $\boldsymbol{w} = \boldsymbol{\theta}$ in f and f incentivizes 0 responses or improvement, then the optimal IC and IR discount mechanism is $G = 0$.*

This is straightforward since the decision maker cannot further lower the error from $err(\eta)$ and thus does not want to pay the agents.

If f incentivizes gaming, then the equilibrium individual error becomes, $\mathcal{E}(f, \boldsymbol{a}_C, \boldsymbol{x}) = [\boldsymbol{\theta}^T(\boldsymbol{x} + P\boldsymbol{a}_C^*) - \boldsymbol{\theta}^T \boldsymbol{x}]^2 + err(\eta) = (\boldsymbol{\theta}^T P \boldsymbol{a}_C^*)^2 + err(\eta)$, which is independent of the pre-response attribute $\boldsymbol{x}$.

Theorem 19. *If $\boldsymbol{w} = \boldsymbol{\theta}$ in f, and f incentivizes gaming, then the optimal IC, IR, and BB $G \neq 0$ can be found as follows:*

Choose $i_A = \arg\max_{j \leq M_+} (P^T \boldsymbol{\theta})_j / c_j$ as the target dimension, and set $\triangle c_{i_A} = \triangle c_{i_A}^*$.

Then, derive the alternative form of individual subsidy surplus as $s(\alpha) = (2\alpha - \alpha^2)(\boldsymbol{\theta}^T P \boldsymbol{a}_C^*)^2 - \alpha B \triangle c_{i_A} (c_{i_A} - \triangle c_{i_A})^{-1}$ and get $\alpha^* = \arg\max_{\alpha \leq 1} s(\alpha) = 1 - \frac{B \triangle c_{i_A} (c_{i_A} - \triangle c_{i_A})^{-1}}{2(\boldsymbol{\theta}^T P \boldsymbol{a}_C^*)^2}$. Then find the optimal $\bar{c}$ by $\bar{c} = \alpha^* B \triangle c_{i_A} (c_{i_A} - \triangle c_{i_A})^{-1}$.

An interesting observation is that the decision maker does not try to completely remove gaming with the discount mechanism. This is because when the error drops to a sufficiently low level, the marginal subsidy benefit becomes lower than the marginal subsidy cost, which is a constant.

[6] The optimal CS strategy in regression does not guarantee $\boldsymbol{w} = \boldsymbol{\theta}$ when incentivizing improvement is impossible.

5.5 Demographic Groups and Social Well-Being

Consider now the case where agents come from two demographic groups distinguished by a *sensitive attribute* $d \in \{1, 2\}$ (e.g., gender, race), which is not a part of the N skill-related attributes (not in $\boldsymbol{x}$) and is never influenced by an agent's action $\boldsymbol{a}$. Suppose the decision rule is *not allowed* to use the sensitive attribute as input but that it can be used to design *group specific* subsidies, so that different groups are subject to different incentive mechanisms provided the group identities are truthfully revealed.

We are particularly interested in how the subsidy mechanisms and their corresponding AS outcomes influence the fairness of the system. Below we introduce a number of commonly used definitions of group differences and social well-being measures related to fairness. Here, the term *well-being* is used to refer to a broader set of metrics defined below whereas *welfare* is used in the narrower sense of sum utility.

5.5.1 Group Differences

Without loss of generality, we will refer to group 1 as the *advantaged* group and 2 as the *disadvantaged* group.[7] We consider the following set of definitions; the first is new to the best of our knowledge and the other two were introduced in [67].

Definition 10 (Group Disadvantages)**.** *We say group 2 is*

1. *disadvantaged in attributes in classification if* $F^{(2)}(l) > F^{(1)}(l)$ *for* $l \in (0, 1)$*, where* $F^{(d)}$ *is the cumulative density function (cdf) of the conditional pre-response qualification status conditioned on* $d \in \{1, 2\}$*; the same in regression if* $F^{(2)}(y) > F^{(1)}(y)$ *for* $y \in (0, \max_{\boldsymbol{x}} q(\boldsymbol{x}))$.
2. *disadvantaged in positive individuals (in classification) if* $F_+^{(2)}(l) > F_+^{(1)}(l)$*, where* $F_+^{(d)}$ *is the cdf of conditional pre-response qualification status* $(l(\boldsymbol{x})|Y = 1, D = d), d \in \{1, 2\}$.
3. *disadvantaged in action cost if* $h^{(2)}(\boldsymbol{a}) > h^{(1)}(\boldsymbol{a}), \forall \boldsymbol{a} \neq \mathbf{0}$*, where* $h^{(d)}$ *denotes the action cost functions with sensitive attribute* $d \in \{1, 2\}$*. Moreover, the minimum effective discount values satisfy* $(\triangle c^{(1)})_i^* \leq (\triangle c^{(2)})_i^*, \forall i$.

5.5.2 Social Well-being Metrics

We will use the equilibrium qualification status $\mathbb{E}[y_t'], t \in \{C, A\}$ as an *efficiency* oriented social well-being metric. We also introduce *fairness* oriented well-being metrics.

[7] The group index shows up in superscripts.

Definition 11 (Quality gain)**.** *Quality gain measures the increase in agents' expected qualification status (positive rate in classification) in the response phase:*

$$\triangle Q_t := \mathbb{E}[Y_t'] - \mathbb{E}[Y_t];\ \ \triangle Q_t^d := \mathbb{E}[Y_t'|D=d] - \mathbb{E}[Y|D=d];\ \ \forall d \in \{1,2\}, \forall t \in \{A,C\}. \quad (5.14)$$

$\gamma_t^Q(f,G) := \triangle Q_t^{(1)} - \triangle Q_t^{(2)}$ *further measures the group difference in this gain under game type* t.

Clearly, if f incentivizes improvement, then $\triangle Q_C > 0$; if $G \neq 0$ incentivizes improvement, then $\triangle Q_A > 0$. What's more interesting is to compare the quality gains across different groups and under different game types.

Definition 12 (Classification Fairness)**.** *Considering two commonly used fairness criteria in classification, Equal Opportunity (*`EO`*) (equalized true positive rates) [36] and Demographic Parity (*`DP`*) (equalized positive decision rates), and define their respective group differences:*

$$\gamma_t^{EO}(f,G) := Pr(f(\boldsymbol{z}_t)=1|Y_t'=1, D=1) - Pr(f(\boldsymbol{z}_t)=1|Y_t'=1, D=2), t \in \{A,C\} \quad (5.15)$$

$$\gamma_t^{DP}(f,G) := Pr(f(\boldsymbol{z}_t)=1|D=1) - Pr(f(\boldsymbol{z}_t)=1|D=2). \quad (5.16)$$

5.5.3 Fairness Issues in the CS/LS Equilibrium

We start with a number of fairness limitations of the CS equilibria in classification and regression; the same results apply to LS.

Theorem 20. *In the equilibrium CS outcome of classification where two groups have the same action cost, then (i) if group 2 is disadvantaged in attributes, then there is a* `DP` *gap no matter if* f *incentivizes improvement or gaming; and (ii) if group 2 is disadvantaged in positive individuals, then there is an* `EO` *gap if* f *incentivizes gaming but not necessarily if* f *incentivizes improvement.*

Part (1) is a direct result of $1 - F^{(1)}(l) > 1 - F^{(2)}(l)$, and the two groups have the same implicit threshold, which is the lower side boundary of their manipulation margins (since every agent above it will manipulate to get a positive decision outcome), and $\mathcal{M}^{(1)}(f) = \mathcal{M}^{(2)}(f)$ since the two groups have the same action cost. For part (2), whether there is a quality gain gap entirely depends on whether f incentivizes improvement and the distribution of each group in its manipulation margin $\mathcal{M}^{(d)}(f)$. For example, we can have $Pr(\boldsymbol{x} \in \mathcal{M}^{(2)}(f)|D=2) > Pr(\boldsymbol{x} \in \mathcal{M}^{(1)}(f)|D=1)$ and thus group 2 have more agents to improve and may have an inverse quality gain gap.

Theorem 21. *In the equilibrium CS outcome of classification and regression, if group 2 is disadvantaged in action cost but has the same pre-response attribute distribution as group 1 (for positive individuals as well), then there is (i) a quality gain gap only if* f *incentivizes improvement; (ii) an* `EO` *gap no matter if* f *incentivizes improvement or gaming; and (ii) a* `DP` *gap no matter if* f *incentivizes improvement or gaming.*

To understand the above result, we note that if group 2 is disadvantaged in cost, we have $\mathcal{M}^{(1)}(f) \supseteq \mathcal{M}^{(2)}(f)$, so even when group 2 has the same pre-response attribute distribution, a larger portion of group 1 are accepted in the equilibrium, causing the `DP` gap. This is similar to the reason of an `EO` gap when f incentivizes gaming. If f incentivizes improvement, then a larger portion of group 1 will improve and be accepted in the equilibrium, causing a quality gain gap and an `EO` gap simultaneously.

5.5.4 Influence of the Discount Mechanism on Fairness

Here we analyze how the discount mechanism G alone may influence the fairness.

Theorem 22. *If group 2 is disadvantaged in cost but has the same pre-response attribute distribution, then a rational decision maker will choose a G that widens the quality gain gap in both classification and regression.*

Theorem 22 means that a rational mechanism for the decision maker is always making the system more unfair when the quality gain gap is the metric. The rational mechanism influences the `DP` and `EO` gap but does not always make them worse.

5.6 Third Party Mechanisms

We next discuss an alternative system where the discount mechanism is implemented by a third party, who subsidizes the agents' improvement actions in the same way as described in Section 5.2 and charges the decision maker a tax $\mathcal{T}(G)$ for improved decision performance. The decision maker's AS utility in this alternative system is

$$U_A^{(cls)}(f) = \int_{\mathcal{X}} Pr\big(f(\boldsymbol{x} + P\boldsymbol{a}_A^*(\boldsymbol{x})) = y'_A\big)p(\boldsymbol{x})d\boldsymbol{x} - \mathcal{T}(G),$$
$$U_A^{(reg)}(f) = \int_{\mathcal{X}} \mathbb{E}_\sigma\big[-\big(f(\boldsymbol{x} + P\boldsymbol{a}_A^*(\boldsymbol{x})) - y'_A\big)^2\big]p(\boldsymbol{x})d\boldsymbol{x} - \mathcal{T}(G)\ .$$

The IR condition for the decision maker is $U_A^{(cls)}(f) \geq U_C^{(cls)}(f)$ or $U_A^{(reg)}(f) \geq U_C^{(reg)}(f)$. In addition, we also consider the common mechanism criterion of budget balance: if the charged price is no less than the subsidy cost of the third party, then the mechanism is (weakly) budget balanced:

Definition 13 (Budget Balance)**.** *The third party is considered (weakly) budget balanced if $\mathcal{T}(G) \geq H(G)$.*

The mechanism designer can induce truthful revelation of the sensitive attribute by the agents as follows: (1) Let G consist of two group-specific mechanisms $G^{(1)}$ and $G^{(2)}$; agents who do not

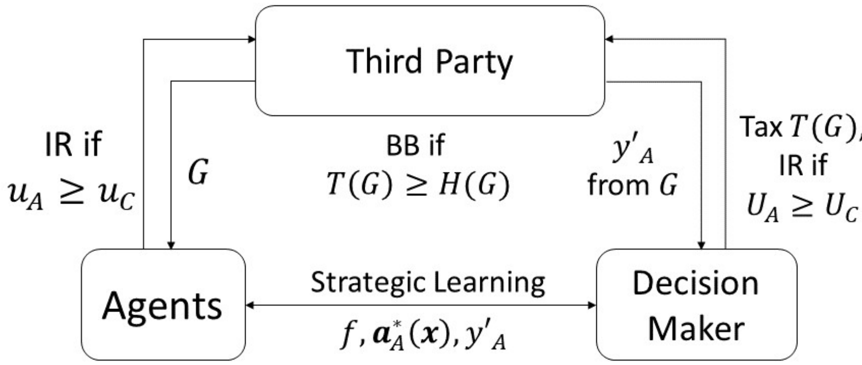


Figure 5.7: An Illustration of the Alternative Three-party Augmented Strategic Learning System.

reveal their d participate in $G^{(1)}$; (2) Ensure that $\triangle c_i^{(1)} \leq \triangle c_i^{(2)}, \forall i$ and $(\triangle \boldsymbol{c}^{(1)})^T \boldsymbol{a} \in [\underline{c}^{(1)}, \overline{c}^{(1)}] \Rightarrow (\triangle \boldsymbol{c}^{(2)})^T \boldsymbol{a} \in [\underline{c}^{(2)}, \overline{c}^{(2)}]$. Then, group 1 agents are indifferent about revealing d while revealing d is the dominant strategy for group 2 agents. Figure 5.7 illustrates the three-party AS learning system.

5.6.1 Objectives of the Third Party

We introduce two types of third party mechanism designers, *efficiency* oriented and *fairness* oriented. An *efficiency* oriented third party tries to maximize the equilibrium social qualification status $W_{eff}^{(cls)}(f, G) := Pr(y'_A = 1)$, $W_{eff}^{(reg)}(f, G) := \mathbb{E}[y'_A]$; we call the corresponding equilibrium AS outcome the *efficient AS outcome* or *AS-eff* in short. On the other hand, a *fairness* oriented third party aims to minimize a non-negative and non-zero linear combination of the fairness gaps, or equivalently, maximizing

$$W_{fair}^{(cls)}(f, G) := -\beta^Q \gamma_A^Q(f, G) - \beta^{DP} \gamma_A^{DP}(f, G) - \beta^{EO} \gamma_A^{EO}(f, G); \;\; W_{fair}^{(reg)}(f, G) := -\gamma_A^Q(f, G),$$

for some $\beta^Q, \beta^{EO}, \beta^{DP} \geq 0, \beta^Q + \beta^{EO} + \beta^{DP} > 0$. We call the corresponding equilibrium AS outcome the *fair* AS outcome or *AS-fair* in short.

For conciseness, we use *AS-dm* to denote the decision maker's equilibrium AS outcome.

Theorem 23. *If there is a mechanism that is IC and IR and satisfies $S(f, G) > 0$, then a mechanism that satisfies IC, IR, and BB criteria exists and weakly improves the third party's social well-being objective (either efficiency or fairness oriented) compared to the original AS equilibrium.*

5.6.2 Influence of Mechanism Designers' Objectives

Finally, we discuss how the objective of the mechanism designer and the corresponding incentive mechanisms influence the equilibrium efficiency and fairness oriented social well-being metrics. We compare the different AS, CS, and LS equilibrium outcomes where they have the same decision rule f and focus on how the incentive mechanisms for different objectives affect the outcome.

Definition 14. *We say a mechanism* $G^{(d)} \neq 0$ *is an* ideal mechanism *if it is IC and IR for group* d *agents and achieves* $S(f, G^{(d)}) > 0$ *on group* d, $\forall d \in \{1, 2\}$.

Theorem 24. *If group 2 is disadvantaged in action cost but has the same pre-response attribute distribution as group 1 (for positive individuals as well), then in the equilibrium,*

1. *the* `DP` *gap in weak ascending order is: AS-fair, CS(LS), AS-dm, AS-eff;*

2. *the* `EO` *gap (or quality gain gap) in weak ascending order is: AS-fair, CS(LS), AS-dm, AS-eff;*

3. *The social quality improvement in weak descending order is: AS-eff, AS-dm, CS(LS).*

If there is an ideal mechanism for group 1, then AS-fair is strictly the lowest in `DP` *gap; the orders in* `EO` *gap (or quality gain gap) and quality improvement becomes strict for CS(LS), AS-dm and AS-eff. Moreover, if there is an ideal mechanism for group 2, AS-fair is strictly the lowest in* `EO` *gap (or quality gain gap).*

Below we provide some explanations of the statements in Theorem 24. For an efficiency oriented third party, the set of agents it incentivizes is a superset of the agents incentivized by the decision maker, making AS-eff the best in part (3). This is because subsidizing the agents with a positive individual subsidy surplus not only helps the third party improve the objective but also raises the budget to subsidize agents with a negative individual subsidy surplus (individual subsidy deficit). Moreover, the efficiency oriented third party tries to incentivize more agents from group 1 since they are "cheaper" to incentivize and thus exacerbates the fairness issues in parts (1) and (2).

For a fairness oriented third party, it can also incentivize a superset of agents incentivized by the decision maker, but that means incentivizing some group 1 agents, which results in two conflicting effects: it helps the third party gather more "funding" to subsidize group 2 agents, but at the same time makes the fairness issue worse. As a result, the social quality improvement in AS-fair is better than CS (LS) and worse than AS-eff, but how it compares to AS-dm depends on the specific game parameters and thus is not discussed in part (3). When there is an ideal mechanism for group 2, the third party can ignore the dilemma of subsidizing group 1 agents and focus on subsidizing only group 2 agents to improve fairness in parts (1) and (2).

The ideal mechanisms in Theorem 24 make the comparison strict. The existence of an ideal $G^{(2)}$ is a sufficient condition for the existence of an ideal $G^{(1)}$ when group 2 is disadvantaged in cost but has the same distribution. This is because $G^{(2)}$ itself is ideal for group 1.

Theorem 25. *In both classification and regression problems, if group 2 is disadvantaged in attributes (resp. positive individuals) but has the same action cost as group 1 then*

1. *the* `DP` *(resp.* `EO`*) gap in AS-fair outcome is weakly the lowest, and is strictly the lowest if there is an ideal mechanism for group 2;*

2. *the social quality improvement in AS-eff outcome is weakly the highest, and is strictly the highest if there is an ideal mechanism for either group.*

When group 2 has the same cost, then an ideal $G^{(2)}$ is no longer sufficient or necessary for an ideal $G^{(1)}$ to exist for general classification problems, and that's why the condition in part (2) looks different from Theorem 25. But the existence of an ideal $G^{(2)}$ is sufficient and necessary for the existence of an ideal $G^{(1)}$ in regression, as well as in a special class of classification problems where $\boldsymbol{w} = \boldsymbol{\theta}$ in f and l is convex on $[0, \tau]$.[8] From Theorem 20, we know that DP and EO gap always exist in the CS (LS) problem, but if there is an ideal $G^{(2)}$, the fairness oriented third party can further incentivize group 2 agents to reduce the gap in part (1) (those not in $\mathcal{M}^{(2)}(f)$ to reduce the DP gap).

Theorem 26. *Suppose group 2 is disadvantaged in cost but has the same pre-response distribution (for positive individuals as well). Denote* $p^{(d)} := Pr(D = d)$*, then an IC, IR, and BB mechanism* $G \neq 0$ *that satisfies* $\gamma_A^Q = \gamma_A^{EO} = \gamma_A^{DP} = 0$ *exists if* $S(f, G^{(1)}) + (1 - p^{(1)})H(G^{(1)}) \geq p^{(2)}H(G^{(2)}),$ *s.t.* $h_A^{(1)}(\boldsymbol{a}) = h_A^{(2)}(\boldsymbol{a}), \forall \boldsymbol{a}$*.*

In general, this condition can hold if $p^{(1)}$ is much larger than $p^{(2)}$, i.e., the disadvantaged group is also the minority group in the population or $S(f, G^{(1)})$ is very high.

Remark 4. *Our results generalize to multiple groups when the definitions of group disadvantages and fairness metrics are consistent.*

5.7 Numerical Results

This section presents numerical results obtained using the FICO score [85] dataset preprocessed in [36]. The credit card holders are considered agents and they have repayment rates that can map to the likelihood function l in our model. The decision maker uses binary classification to predict whether the agents will default. We assume that $\theta = 1$, $P = [1, 1]$, and the agent can either choose a_1 to improve or a_2 to game the classifier $f(z) = \mathbf{1}(z \geq \tau)$, i.e., x is the pre-response normalized FICO score as well as the attribute, $x' = x + a_1$ is the post-response attribute, and $z = x + a_1 + a_2$ is the post-response normalized FICO score. Figure 5.8 shows how the repayment rate $l(x)$ changes with x; it has an S-shape, with $l(x) = 0.5$ approximately corresponding to $x = 0.3$ and $l(x)$ (nearly) convex on $[0, 0.3]$. We assume that the decision maker chooses $w = 1$, which aligns with the LS and CS optimal solution from Section 5.3 when $c_2 < c_1$.

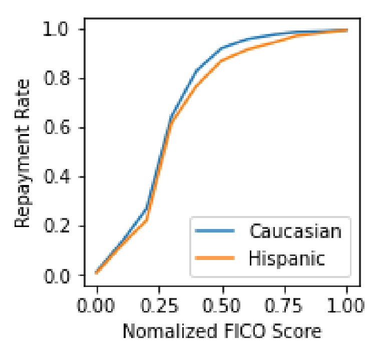
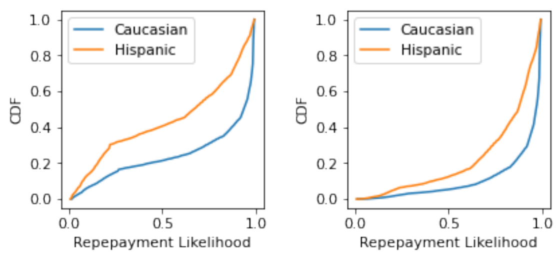


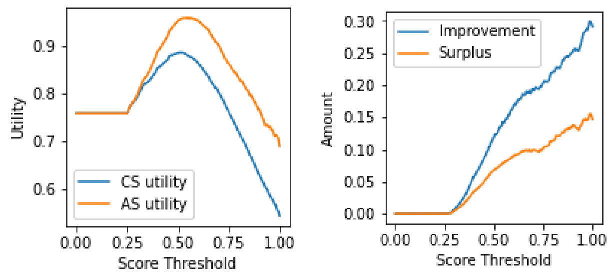
Figure 5.8: Repay Rate $l(x)$

[8] We are excluding extreme distributions in the "iff" claim, e.g., $Pr(\boldsymbol{x} \in \mathcal{M}(f)) = 0$.



(a) Entire Group (b) Positive Individuals

Figure 5.9: The Likelihood CDF



(a) CS/AS Utilities (b) $\triangle Q$ and $S(f, G)$

Figure 5.10: Single Group (Caucasian) Results

We start with the properties of the discount mechanism and show how the decision maker's CS and AS utility changes with different choices of threshold τ. We then show the impact the incentive mechanisms have on social well-being metrics.

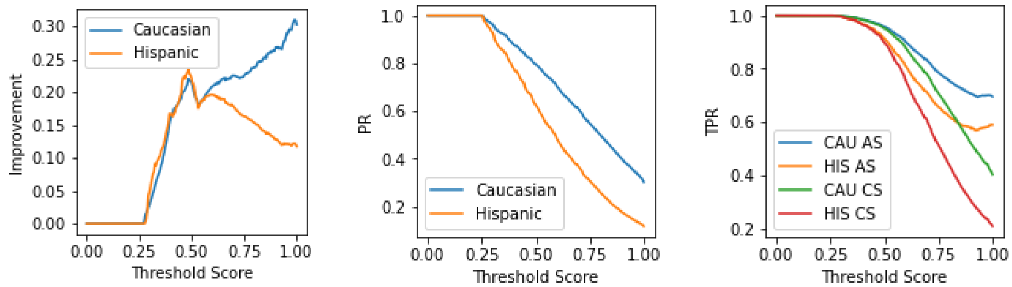
Throughout this section, we use a quadratic outcome likelihood cost function and assume that $c_1^{(1)} = c_1 = 8$ and $c_2^{(1)} = c_2 = 4$ (for the advantaged group if there are action cost differences). For the multiple group case, we make the following two sets of comparisons. (1) Groups with different distributions: the Hispanic group is disadvantaged in features and in positive individuals compared to the Caucasian group (see Figure 5.9). (2) Groups with different costs: we will assume there are two subgroups (A and B) in the Caucasian group, and group 2 has higher action costs $c_1^{(2)} = 10$ and $c_2^{(2)} = 5$. We set $p^{(1)} = 0.8, p^{(2)} = 0.2$ as the population proportions.

As a result, we show the AS-fair equilibrium outcome is the best well-rounded system design for the augmented strategic learning problems.

The decision maker's AS and CS utility. Using only the Caucasian data, the set of results in Figure 5.10 shows how the AS/CS decision maker utilities, subsidy surplus, and qualification status improvement change with the threshold τ.

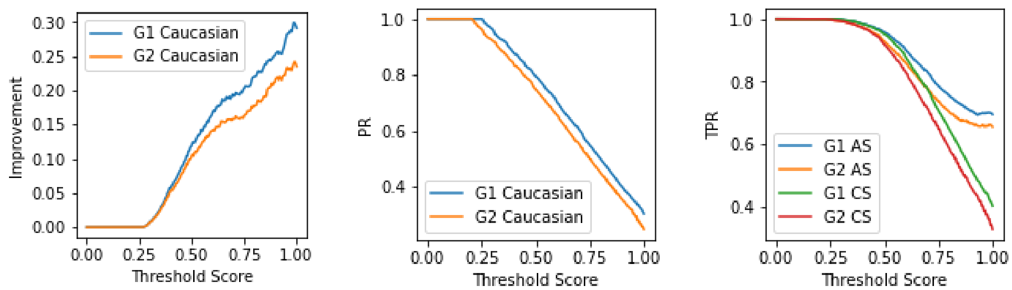
We can see that the AS utility is always higher than the CS utility (Fig. 5.10). This is because their difference is the subsidy surplus, which is non-negative for a rational decision maker. We note that the CS utility should always be single-peaked but the AS utility may have multiple local maxima since the value of subsidy surplus is not monotone in τ and depends on $p(x)$. For other choices of c_1, c_2 values, we find that the larger the difference $c_1 - c_2$, the smaller the utility difference and the closer the optimal thresholds are ($|\tau_{AS}^* - \tau_{CS}^*|$ lower). Both the subsidy surplus in (5.12) and the qualification status improvement in (5.14) are positive, indicating the decision maker's selfish strategy is also benefiting the efficiency oriented social well-being. The improvement and subsidy surplus are also highly positively correlated with a correlation coefficient of 0.92.

Social well-being of the strategic incentive mechanism. Figure 5.11 (resp. Figure 5.12) shows the quality improvement, PR and TPR, (and thus we can see the `DP`, and `EO` gap from the curve differences) when the Hispanic group (resp. Caucasian subgroup 2) is disadvantaged in features and



(a) Improvement (b) PR (c) AS/CS TPR

Figure 5.11: Disadvantaged in Features



(a) Improvement (b) PR (c) TPR

Figure 5.12: Disadvantaged in Costs

positive individuals (resp. costs) compared to the Caucasian group (resp. Caucasian subgroup 1) in the CS(LS) and AS-dm equilibrium. The decision maker does not incentivize agents outside of the manipulation margin and thus the CS and AS PR curves are the same.

We can see from Figure 5.11a that when τ is in the lower score ranges, the Hispanic group has a slightly higher qualification status improvement compared to the Caucasian group, whereas if τ is in the higher score ranges, the Caucasian group has a much higher improvement. Intuitively, this is because the Hispanic (resp. Caucasian) group has a higher probability mass in the lower (resp. higher) score ranges and a low (resp. high) τ incentivizes a higher proportion of agents to improve in the Hispanic (resp. Caucasian) group. Figure 5.11b shows that the PR is 1 when $\tau < 0.25$; this is because all agents can manipulate to get $f(z) = 1$. When $\tau > 0.25$, the PR is strictly decreasing in τ for both groups and the Caucasian group always has a higher PR, i.e., the Hispanic group will suffer from a `DP` gap in both CS and AS-dm equilibrium. This is because the lower side boundary of the manipulation margin becomes an implicit threshold, where all agents above the implicit threshold can manipulate (no matter improvement or gaming) to get accepted. The implicit threshold is the same for both groups since they have the same action cost, and the `DP` gap is caused by the disadvantage in pre-response attribute distribution (Theorem 20 part (1)). For similar reasons, Figure 5.11c shows that the CS and AS TPR is 1 when $\tau < 0.3$. Therefore, we can see that the AS TPR is always higher than the CS TPR for either group, because now some agents improved their qualification status and get accepted at the same time, making the numerator and denominator of the TPR formula increase by the same amount and thus increase the TPR. On the other hand, the Hispanic group suffers from an EO gap in both the CS and the AS-dm equilibrium, as previously discussed in Theorem 20 part (2).

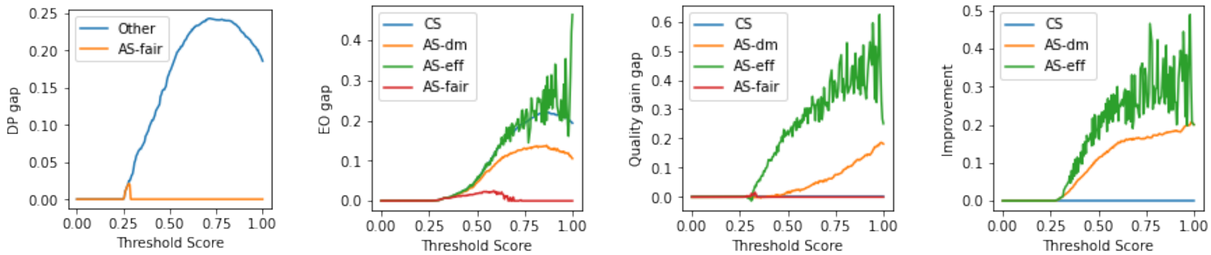
Figure 5.12a and 5.12c support our claims in Theorem 22 part (3), where the incentive mechanism widens the quality gain gap and the EO gap. Figure 5.12b shows PR curves and the DP gap between the two subgroups, which is determined by the pre-response attribute probability mass within $[\tau - 1/c_2^{(1)}, \tau - 1/c_2^{(2)}]$ (the difference between the manipulation margins in the two groups). Figure 5.12c shows the CS and AS TPR curves and the EO gaps; the implicit threshold creates the CS EO gap, and the fact that group 1 agents are cheaper to incentivize jointly creates the AS EO gap.

Social Well-being metrics with the third party incentive. Social well-being results under the third party model are shown in Figure 5.13 where groups have attribute distribution differences (Caucasian and Hispanic group), and in Figure 5.14 where groups have cost differences (Caucasian subgroups).

We can see in both sets of results that the AS-fair equilibrium outcome significantly reduces and even removes the fairness issues in the system, whereas the AS-eff equilibrium outcome has the worst fairness metrics. On the other hand, the AS-eff equilibrium achieves the highest social qualification status improvement. We note that the chosen AS-fair outcomes used mechanisms that incentivized a superset of agents compared to those that are incentivized by the decision maker, and thus it achieves a higher social qualification status improvement than AS-dm as well.

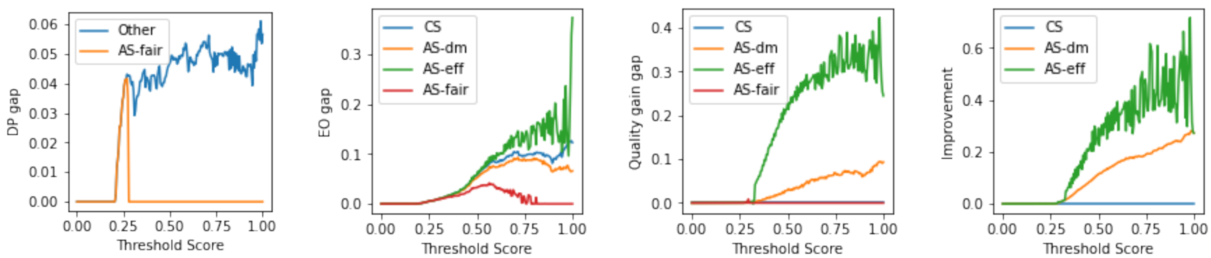
5.8 Chapter Conclusions

In this chapter, we formulated Stackelberg game models to study the strategic classification and regression problem, where the decision maker's strategy combines a decision rule and an incentive mechanism. Our model provides an extension of the previously studied strategic learning problems. We showed how the decision maker can design discount-based incentives mechanisms to use in conjunction with its decision rule, by providing conditions on when this problem is computationally intractable, discussing when and how approximation algorithms can find reasonable mechanisms in polynomial time, and when the optimal mechanism can be found in closed-form. We then discussed the efficiency and fairness oriented social well-being properties of the augmented strategic learning system when multiple demographic groups co-exist. We also examined an alternative model where the incentive mechanism is provided by a third party, whose objective is optimizing some of the social well-being metrics with an IC, IR, and BB mechanism, and showed how an efficiency-oriented and fairness-oriented third party can influence the equilibrium social well-being metrics. We conducted numerical experiments on the FICO dataset to demonstrate the impact of the



(a) DP gap (b) EO gap (c) Quality gain gap (d) Improvement

Figure 5.13: Third Party Outcomes with Attribute Distribution Differences



(a) DP gap (b) EO gap (c) Quality gain gap (d) Improvement

Figure 5.14: Third Party Outcomes with Cost Differences

incentive mechanism on the system. Our findings established that a fairness-oriented third party can provide the best well-rounded equilibrium outcomes compared to a selfish decision maker, an efficiency-oriented third party, or a system without an incentive mechanism.

Part III

Conclusions and Future Directions

Chapter 6

Conclusions and Future Directions

This thesis covers two main directions in network games, with a strong focus on multi-scale network games.

6.1 Conclusions on Equilibrium Computation and Analysis on Multi-Scale Network Games

The first direction is the equilibrium computation and analysis in multi-scale network games. In Chapter 2, we proposed a novel representation of games that have a multi-scale network structure, where the network has a multi-level hierarchical structure. Across the levels of hierarchy, the nodes' utilities at each level only depend on the aggregate strategies of other groups at the same level. We presented several iterative algorithms that leverage the multi-scale game structure, and show that they converge to a pure strategy Nash equilibrium under similar conditions as for best response dynamics in network games. Our experiments demonstrate that the proposed algorithms can yield orders of magnitude scalability improvement over conventional best response dynamics. Our multi-scale algorithms can reveal the extent to which one's group affiliation impacts one's strategic decision making, and how strategic interactions among groups impact strategic interactions among individuals.

In Chapter 3, we introduced and studied a family of structured network games with non-linear best response functions. Prior works on network games have found sufficient conditions for the uniqueness and stability of Nash equilibria, where most of them are difficult to verify. In this work, we showed that the existence of structure in the network (e.g., in the form of communities, or when there are multi-relational dependencies between agents), helps us find alternatives for such conditions, which we refer to as "structured conditions" as opposed to the "unstructured conditions" in previous works. In particular, we show that the structured conditions for the uniqueness and stability of Nash equilibria are related to conditions on matrices which are possibly lower dimensional, with their dimensions depending on the number of partitions naturally arising in a network due to its structured nature. We also demonstrated both analytically and numerically that the structured conditions are sufficient conditions for the unstructured conditions, and that their verification is of much lower computational complexity. We used numerical experiment results to show that the sufficiency gap between the structured conditions and unstructured conditions is small in general and typically occurs in games with some specific characteristics. Moreover, we proposed a new notion of degree centrality to evaluate the influence of a partition in the network, and used it to identify additional conditions for uniqueness and stability.

6.2 Conclusions on Interventions and Mechanism Design on Multi-scale Networks and Multi-population Systems

The second direction is mechanism design and interventions in network games, games with multiple sub-populations and multi-scale networks. In Chapter 4, we studied an intervention problem in network games with community structures and multiple planners. We showed that given any intervention action, the agents will always have a unique NE. The planners can thus use backward induction and design (locally) optimal interventions. We found that no matter if the planners are cooperative or non-cooperative, the system always has a unique subgame perfect equilibrium that fully spends the budget and is Pareto efficient. We also studied the efficiency of the outcomes under different settings in this system, including whether the planners are cooperative and whether the budget is transferable both analytically and numerically. Our analysis shows that we can use the Lagrangian dual optimal variable values to characterize the efficiency, and planners have incentives to share budgets even when they are non-cooperative. The budget transferability also enables uniformly better outcomes than the non-transferable case. Empirically, we observe that the type of network determines which type of (commonly used) budget allocation rule is the most efficient.

In Chapter 5, we formulated Stackelberg game models to study the strategic classification and regression problem, where the decision maker's strategy combines a decision rule and an incentive mechanism. Our model provides an extension of previously studied strategic learning problems. We showed how the decision maker can design discount-based incentives mechanisms to use in conjunction with its decision rule, by providing conditions on when this problem is computationally intractable, discussing when and how approximation algorithms can find reasonable mechanisms in polynomial time, and when the optimal mechanism can be found in closed-form. We then discussed the efficiency and fairness oriented social well-being properties of the augmented strategic learning system when multiple demographic groups co-exist. We also examined an alternative model where the incentive mechanism is provided by a third party, whose objective is optimizing some of the social well-being metrics with an IC, IR, and BB mechanism, and showed how an efficiency-oriented or fairness-oriented third party can influence equilibrium social well-being metrics. We conducted numerical experiments on the FICO dataset to demonstrate the impact of the incentive mechanism on the system. Our findings established that a fairness-oriented third party can provide the best well-rounded equilibrium outcomes compared to any of the following: a selfish decision maker, an efficiency-oriented third party, or a system without an incentive mechanism.

6.3 Future Directions

While some of the issues of multi-scale networks discussed in Chapter 2 have also been considered in the network science literature (e.g., hierarchical clustering, etc.), The latter is primarily concerned with community structure in networks and its detection and identification, rather than modeling how communities *interact*, which is critical in describing a formal multi-scale structure for games. A very important future direction is to identify and obtain relevant field data in order to create realistic benchmarks for multi-scale games and to enable the implementation of our multi-scale algorithms. This would involve identifying ways to obtain data on how communities (and not just individuals) interact. With access to data on interactions at multiple scales (e.g., among members and among groups), we can apply our algorithms to specific multi-scale networks. To use criminal networks (criminal organizations and their members) as an example, given game models constructed with the help of domain expertise, we can:

1. infer utility models from observational data at multiple scales;
2. compute equilibria predicting, say, criminal activity as a function of structural changes to organizations;
3. study policies (including strengthening or weakening connections between agents or groups, endowing agents/groups with more resources (e.g., lower costs of effort, etc.) that would induce more desirable equilibrium outcomes.

We showed in chapter 3 that the structured conditions are computationally much cheaper than their unstructured counterparts with low sufficiency gaps, and some natural ways of partitioning the network empirically work well. However, it remains an open question of how to find the optimal or near-optimal partitions in a more rigorous way.

We are also interested in extending our multi-planner intervention framework in Chapter 4 to include a wider variety of intervention types, e.g., by changing game parameters like the connection strengths or studying a more general intervention cost. It is also an interesting problem when the connection strengths are initially unknown to the planner, who has to learn the game parameters in an iterative manner via repeated interactions.

For Chapter 5, an interesting extension is to study other incentive mechanisms in addition to the decision rules and explore the possibility of utilizing network effects.

APPENDIX A

Chapter 2 Appendix

A.1 Structured Variational Inequalities

A structured variational inequality SVI_n arises when a VI problem has n separable operators. This is used to analyze our game under the multi-scale perspective described in Section 2.3.

We now introduce a particular type of SVI_2 relevant to our model. Suppose the N level-1 agents form M disjoint groups in the game and S_j denotes the jth level-1 group, whereby $i \in S_j$ denotes that a_i is a member of S_j. Consider the following utility function of a_i:

$$u_i(x_i, \boldsymbol{x}_{-i}, y_j, \boldsymbol{y}_{-j}) = u_i^{(1)}(x_i, \boldsymbol{x}_{-i}) + u_j^{(2)}(y_j, \boldsymbol{y}_{-j}), \tag{A.1}$$

where $\boldsymbol{x} \in \mathbf{R}^N$ denotes the level-1 action profile and $\boldsymbol{y} \in \mathbf{R}^M$ denotes the level-2 action profile, and $A\boldsymbol{x} + \boldsymbol{y} = \mathbf{0}$, for

$$A_{ji} = \begin{cases} -1, & \text{if } i \in S_j \\ 0, & \text{else} \end{cases}, j = 1, \ldots, M, i = 1, \ldots, N .$$

Thus $A\boldsymbol{x} + \boldsymbol{y} = \mathbf{0}$ is equivalent to $y_j = \sum_{i \in S_j} x_i$. We say $\boldsymbol{x}$ and $\boldsymbol{y}$ are two separated operators, and define

$$\begin{aligned}
F^{(1)}(\boldsymbol{x}) &:= \Big(-\nabla_{x_i} u_i^{(1)}(\boldsymbol{x}) \Big)_{i=1}^{N}, \quad x_i \in K_i^{(1)}, \\
F^{(2)}(\boldsymbol{y}) &:= \Big(-\nabla_{y_j} u_j^{(2)}(\boldsymbol{y}) \Big)_{j=1}^{M}, \quad y_j \in K_j^{(2)}, \\
K^{(1)} &= \prod_{i=1}^{N} K_i^{(1)}, K^{(2)} = \prod_{j=1}^{M} K_j^{(2)}, \overline{K} = K^{(1)} \times K^{(2)}, \\
\boldsymbol{v} &= \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} \in \overline{K}, \;\; \overline{F}(\boldsymbol{v}) = \begin{bmatrix} F^{(1)}(\boldsymbol{x}) \\ F^{(2)}(\boldsymbol{y}) \end{bmatrix}.
\end{aligned} \tag{A.2}$$

Define $\Omega = \{v \in \overline{K} | A\boldsymbol{x} + \boldsymbol{y} = \boldsymbol{0}\}$. Then the VI($\Omega, \overline{F}$) problem is to find $v^* \in \Omega$, such that: $(\boldsymbol{v} - \boldsymbol{v}^*)^T \overline{F}(\boldsymbol{v}) \geq 0, \ \forall \boldsymbol{v} \in \Omega$. This problem is equivalent to the SVI$_2$ problem VI($\mathcal{W}, Q$) defined in Eqn (A.3)

$$(\boldsymbol{\omega} - \boldsymbol{\omega}^*)^T Q(\boldsymbol{\omega}) \geq 0, \ \forall \boldsymbol{\omega} \in \mathcal{W}, \tag{A.3}$$

where, $\mathcal{W} = \overline{K} \times \mathbf{R}^M$ and

$$\boldsymbol{\omega} = \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ \boldsymbol{\lambda} \end{pmatrix}, Q(\boldsymbol{\omega}) = \begin{pmatrix} F^{(1)}(\boldsymbol{x}) - A^T \boldsymbol{\lambda} \\ F^{(2)}(\boldsymbol{y}) - \boldsymbol{\lambda} \\ A\boldsymbol{x} + \boldsymbol{y} \end{pmatrix}. \tag{A.4}$$

It is easy to see that if we use $\sum_{i \in S_j} x_i$ to replace y_j, then we again have a single operator $\boldsymbol{x}$ and can construct a VI(K, F) as outlined in Section 2.2. There is a one-to-one mapping between a solution $\boldsymbol{x}^*$ to VI(K, F) and a solution $\boldsymbol{\omega}^* = (\boldsymbol{x}^*, -A\boldsymbol{x}^*, \boldsymbol{\lambda}^*)$ to VI($\mathcal{W}, Q$). Therefore, solving either VI(K, F) or VI($\mathcal{W}, Q$) finds the set of NEs.

A.2 Uniqueness of NE

We will introduce some special matrices before we move on to the sufficient conditions for the uniqueness of NE.

Definition 15. *Some special matrices:*

1. *P-matrix: A square matrix is a P-matrix if all its principal components have positive determinant*
2. *Z-matrix: A square matrix is a Z-matrix if all its off-diagonal components are nonpositive*
3. *M-matrix: An M-matrix is a Z-matrix whose eigenvalues' real parts are nonnegative*
4. *L-matrix: An L-matrix is a Z-matrix whose diagonal elements are nonnegative*

For an arbitrary mapping $F : \mathbf{R}^N \to \mathbf{R}^N$, we denote the Jacobian of $F(\boldsymbol{x})$ as $JF(\boldsymbol{x})$. And then $\nabla_j F_i = [JF(\boldsymbol{x})]_{ij}$

Checking if a matrix is P-matrix or not is still not trivial, and we can look at the spectral radius of a matrix instead.

Theorem 27. ***The*** P_Γ ***condition:***

We define the Γ *matrix generated from* F *as follows*

$$\Gamma(F) = \begin{bmatrix} 0 & -\frac{\beta_{1,2}(F)}{\alpha_1(F)} & \cdots & -\frac{\beta_{1,N}(F)}{\alpha_1(F)} \\ -\frac{\beta_{2,1}(F)}{\alpha_2(F)} & 0 & \cdots & -\frac{\beta_{2,N}(F)}{\alpha_2(F)} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{\beta_{N,1}(F)}{\alpha_N(F)} & -\frac{\beta_{N,2}(F)}{\alpha_N(F)} & \cdots & 0 \end{bmatrix}, \tag{A.5}$$

if the spectral radius $\rho(\Gamma(F)) = \|\Gamma(F)\|_2 < 1$*, then we say* F *satisfies the* P_Γ *condition. Then* P_Γ *condition* $\Leftrightarrow$ P_Υ *condition and VI*(K, F) *has a unique solution.*

In [88], the authors mentioned that the P_Υ captures "some kind of diagonal dominance". In fact, the strong diagonal dominance(s.d.d) or weakly chained diagonal dominance(w.c.d.d) of Υ can be an easier yet sufficient condition to check.

Theorem 28. *If* Υ *is s.d.d or w.c.d.d, the NE is unique, since*

Υ *is an s.d.d L-matrix*

$\Rightarrow \Upsilon$ *is a w.c.d.d L-matrix*

$\Leftrightarrow \Upsilon$ *is a nonsigular weakly diagonally dominant(w.d.d)L-matrix*

$\Leftrightarrow \Upsilon$ *is a nonsigular w.d.d M-matrix*

$\Rightarrow \Upsilon$ *is a P-matrix*

Also, when Υ is s.d.d, Γ is a (right, row) substochastic matrix and thus $\rho(\Gamma) < 1$ trivially holds and the NE is unique.

The P_Υ condition guarantees both the uniqueness of NE and the convergence of BRD. Please refer to [79] for more conditions on the uniqueness.

A.3 Proof of Theorem 2

Proof. This algorithm is designed to solve the SVI problem presented in Eqn (A.3) and (A.4). We denote $H = \frac{1}{2}\mathbf{diag}(\boldsymbol{h})$, and the norm $\|\boldsymbol{x}\|_G$, where $G \succ \mathbf{0}$ as

$$\|\boldsymbol{x}\|_G = \boldsymbol{x}^T G \boldsymbol{x}.$$

For simplicity reason, we will use $\boldsymbol{x}$ and $\boldsymbol{y}$ to replace $\boldsymbol{x}^{(1)}$ and $\boldsymbol{x}^{(2)}$ in the remainder of the proof. We can rewrite the steps in Algorithm 3 as follows:

- Step 0: Initialization, given ϵ, μ and $\boldsymbol{x}_0$, let $t = 0$, $\boldsymbol{x}(0) = \boldsymbol{x}_0$, $y_k(0) = \sigma_k(\boldsymbol{x}_{S_k}(0))$; arbitrarily choose $\boldsymbol{\lambda}(0)$.

- Step 1: Find $\boldsymbol{x}^* \in K^{(1)}$ that solves

$$(\boldsymbol{x}' - \boldsymbol{x}^*)^T \Big[f(\boldsymbol{x}^*) - A^T[\boldsymbol{\lambda}(t) - H(A\boldsymbol{x}^* + \boldsymbol{y}(t))] \Big] \geq 0, \tag{A.6}$$

for $\forall \boldsymbol{x}' \in K$, and set $\boldsymbol{x}(t+1) = \boldsymbol{x}^*$.

- Step 2: Find $\boldsymbol{y}^* \in K^{(2)}$ that solves

$$(\boldsymbol{y}' - \boldsymbol{y}^*)^T \Big[f(\boldsymbol{x}^*) - [\boldsymbol{\lambda}(t) - H(A\boldsymbol{x}(t+1) + \boldsymbol{y}^*)] \Big] \geq 0, \tag{A.7}$$

for $\forall \boldsymbol{y}' \in K$, and set $\boldsymbol{y}(t+1) = \boldsymbol{y}^*$.

- Step 3: Set

$$\boldsymbol{\lambda}(t+1) = \boldsymbol{\lambda}(t) - H(A\boldsymbol{x}(t+1) - \boldsymbol{y}(t+1)) \tag{A.8}$$

- Step 4: Convergence verification: If $\|\boldsymbol{\omega}(t+1) - \boldsymbol{\omega}(t)\|_\infty < \epsilon$, then stop. Otherwise let $t \leftarrow t+1$ and go back to Step 1.

When we have $\boldsymbol{y}(t+1) = \boldsymbol{y}(t)$ and $\boldsymbol{\lambda}(t+1) = \boldsymbol{\lambda}(t)$, $\boldsymbol{\omega}(t+1) = (\boldsymbol{x}(t+1), \boldsymbol{y}(t+1), \boldsymbol{\lambda}(t+1))$ is the solution to our SVI_2. We denote the unique solution as $\boldsymbol{\omega}^* = (\boldsymbol{x}^*, \boldsymbol{y}^*, \boldsymbol{\lambda}^*)$. From Eqn (A.7) and (A.8), we have the following from Section 2 of [38],

$$\begin{aligned}
&\|\boldsymbol{y}(t+1) - \boldsymbol{y}^*\|_H^2 + \|\boldsymbol{\lambda}(t+1) - \boldsymbol{\lambda}^*\|_{H^{-1}}^2 \\
\leq &\Big(\|\boldsymbol{y}(t) - \boldsymbol{y}^*\|_H^2 + \|\boldsymbol{\lambda}(t) - \boldsymbol{\lambda}^*\|_{H^{-1}}^2 \Big) - \Big(\|\boldsymbol{y}(t+1) - \boldsymbol{y}(t)\|_H^2 + \|\boldsymbol{\lambda}(t+1) - \boldsymbol{\lambda}(t)\|_{H^{-1}}^2 \Big) \\
< &\|\boldsymbol{y}(t) - \boldsymbol{y}^*\|_H^2 + \|\boldsymbol{\lambda}(t) - \boldsymbol{\lambda}^*\|_{H^{-1}}^2,
\end{aligned} \tag{A.9}$$

which shows the contraction property of the sequence $\{(\boldsymbol{y}(t), \boldsymbol{\lambda}(t))\}$ and thus proves the convergence of the algorithm.

A more detailed proof of convergence of the above steps in Eqn (A.6)-(A.8) is covered in [29, 33], and a more generalized version of the above steps and convergence proofs are covered in [94, 61].

□

A.4 Proof of Theorem 3

A.4.1 Full version of HH-BRD

We will first show the ull version of HH-BRD, suppose the superlevel partitions is taken between level $q-1$ and level q, then for $i = 1, \ldots, N^{(1)}$,

$$\hat{u}_i^{(sl_1)} = \sum_{l=1}^{q-1} u_{k_{il}}^{(l)}(x_{k_{il}}, \boldsymbol{x}_{I_{k_{il}}}) - L_{k_{iq}}^{(sl_1,sl_2)}\left(\sigma_{k_{iq}}^{(1,q)}(\boldsymbol{x}_{S_{k_{iq}}^{(1,q)}}), x_{k_{iq}}^{(q)}\right), \tag{A.10}$$

where

$$S_p^{(1,q)} = \{a_i^{(1)} \mid k_{iq} = p\}, \sigma_p^{(1,q)}(\boldsymbol{x}_{S_p^{(1,q)}}) = \sum_{a_i^{(1)} \in S_p^{(1,q)}} x_i^{(1)}.$$

And for $j = 1, \ldots, N^{(q)}$

$$\hat{u}_j^{(sl_2)} = \sum_{l=q}^{L} u_{k_{jl}}^{(l)}(x_{k_{jl}}, \boldsymbol{x}_{I_{k_{jl}}}) - L_j^{(sl_2,sl_1)}\left(x_j^{(q)}, \sigma_j^{(1,q)}(\boldsymbol{x}_{S_j^{(1,q)}})\right). \tag{A.11}$$

Please refer to Algorithm 9 for the pseudo code of the full version of this algorithm. The loss function updates are similar to that of Algorithm 3.

A.4.2 Proof of Theorem

We will first construct an equivalent 2-level game to the L-level game where $L > 2$, and then show that the action profile update trajectories are the same for the original game and he equivalent game. Finally, the convergence of the equivalent game follows Theorem 2 and thus Algorithm 4 guarantees convergence.

Proof. We define the following counter-part for utility component $u_i^{(l)}(x_i^{(l)}, \boldsymbol{x}_{I_i}^{(l)})$ $(1 < l < q)$

$$\underline{u}_i^{(l)}(\boldsymbol{x}_{S_i^{(1,l)}}, \boldsymbol{x}_{S_{I_i}^{(1,l)}}) = u_i^{(l)}(x_i^{(l)}, \boldsymbol{x}_{I_i}^{(l)}), \tag{A.12}$$

when $x_i^{(l)} = \sigma_i^{(1,l)}(\boldsymbol{x}_{S_i^{(1,l)}}), \forall i, \forall l \in \{2, \ldots, q-1\}$. Both $\boldsymbol{x}_{S_i^{(1,l)}}$ and $\boldsymbol{x}_{S_{I_i}^{(1,l)}}$ are level-1 action profiles. This is exactly how we create the utility functions under the flat perspective, where we expand the higher level aggregate actions down to level-1.

Similarly, we define the following counter-part for utility component $u_j^{(l)}(x_j^{(l)}, \boldsymbol{x}_{I_j}^{(l)})$ $(q < l \leq L)$

$$\underline{u}_j^{(l)}(\boldsymbol{x}_{S_j^{(q,l)}}, \boldsymbol{x}_{S_{I_j}^{(q,l)}}) = u_j^{(l)}(x_j^{(l)}, \boldsymbol{x}_{I_j}^{(l)}), \tag{A.13}$$

Algorithm 9: Hybrid Hierarchical BRD(Full Version)

Initialize the game, $t = 0, x_i^{(1)}(0) = (\boldsymbol{x}_0)_i, i = 1, \ldots, N^{(0)}$
for *l = 2:L* **do**
  for *k = 1:*$N^{(l)}$ **do**
    $\boldsymbol{x}_k^{(l)}(0) = \sigma_k^{(l)}(\boldsymbol{x}_{S_k^{(l)}}(0));$
while *not converged* **do**
  for *k = 1:*$N^{(q)}$ *(Meta-Level-1 Penalty Update)* **do**
    Update $L_k^{(sl_1,sl_2)}$
  for $i = 1 : N^{(1)}$ *(Level-1/Meta-Level-1 Gaming)* **do**
    $x_i^{(1)}(t+1) = BR_i\left(\boldsymbol{x}_{I_i}^{(1)}(t), \boldsymbol{x}_{I_{k_{i2}}}^{(2)}(t), \ldots, x_{k_{iq}}^{(3)}(t), \hat{u}_i^{(sl_1)}\right)$
  for *l = 2:q-1 (Level-2 to Level-q Aggregation)* **do**
    for *j = 1:*$N^{(l)}$ **do**
      $\boldsymbol{x}_j^{(l)}(t+1) = \sigma_j^{(l)}(\boldsymbol{x}_{S_j^{(l)}}(t+1))$
  for *k = 1:*$N^{(q)}$ *(Meta-Level-2 Penalty Update)* **do**
    Update $L_k^{(sl_2,sl_1)}$
  for $j = 1 : N^{(q)}$ *(Level-q/Meta-Level-2 Gaming)* **do**

$$
\begin{aligned}
&x_j^{(q)}(t+1) \\
&= BR_j\Big(\sigma_j^{(1,q)}(\boldsymbol{x}_{S_j^{(1,q)}}), \boldsymbol{x}_{I_j}^{(q)}(t), \boldsymbol{x}_{I_{k_{j(q+1)}}}^{(q+1)}(t), \ldots, \\
&\qquad \boldsymbol{x}_{I_{k_{jL}}}^{(L)}(t), \hat{u}_j^{(sl_2)}\Big)
\end{aligned}
$$

  for *l = q+1:L (Level-2 to Level-q+1 Aggregation)* **do**
    for *p = 1:*$N^{(l)}$ **do**
      $\boldsymbol{x}_p^{(l)}(t+1) = \sigma_p^{(l)}(\boldsymbol{x}_{S_p^{(l)}}(t+1))$
  $t \leftarrow t + 1;$

when $x_j^{(l)} = \sigma_j^{(q,l)}(\boldsymbol{x}_{S_j^{(q,l)}}), \forall j, \forall l \in \{q, \ldots, L\}$. Both $\boldsymbol{x}_{S_i^{(1,l)}}$ and $\boldsymbol{x}_{S_{I_i}^{(1,l)}}$ are level-q action profiles. This time we expand the higher level aggregate actions down to level-q instead of level-1.

So then we can define a "flattened" super-level-1 utility function counterpart for $u_i^{(sl_1)}$ as follows

$$
\overline{u}_i^{(sl_1)}(x_i^{(1)}, x_{\underline{I}_i}^{(1)}) = \sum_{l=1}^{q-1} \underline{u}_{k_{il}}^{(l)}(\boldsymbol{x}_{S_{k_{il}}^{(1,l)}}, \boldsymbol{x}_{S_{I_{k_{il}}}^{(1,l)}}) - L_{k_{iq}}^{(sl_1,sl_2)}\left(\sigma_{k_{iq}}^{(1,q)}(\boldsymbol{x}_{S_{k_{iq}}^{(1,q)}}), x_{k_{iq}}^{(q)}\right), \tag{A.14}
$$

where

$$
\underline{I}_i^{(sl_1)} = \{a_j^{(1)} | k_{jq} = k_{iq}, j \neq i\}.
$$

Similarly, for meta-level 2, we can define a "flattened"(to level-q) function counterpart for $u_j^{(sl_2)}$

as follows

$$\overline{u}_j^{(sl_2)}(x_j^{(q)}, x_{\underline{I}_j^{(sl_2)}}^{(q)}) = \sum_{l=q}^{L} \underline{u}_{k_{jl}}^{(l)}(\boldsymbol{x}_{S_{k_{jl}}^{(q,l)}}, \boldsymbol{x}_{S_{I_{k_{jL}}}^{(q,l)}}) - L_j^{(sl_2,sl_1)}\Big(x_j^{(q)}, \sigma_j^{(1,q)}(\boldsymbol{x}_{S_j^{(1,q)}})\Big), \tag{A.15}$$

where

$$\underline{I}_j^{(sl_2)} = \{a_p^{(q)} | k_{pL} = k_{jL}, p \neq j\}.$$

So now we can create a 2-level game where the level-1(resp. level-q) agents in the original game become the level-1(resp. level-2) agents in the new game with utility functions defined in Eqn (A.14) (resp. Eqn (A.15)). Based on Theorem 2, we know that if we apply SH-BRD, we can converge to the unique NE of the game under Assumptions 1-3.

Then it remains to show that given the same initialization, applying HH-BRD in the original game and the MS-BRD in the new 2-level game generate the same level-1 action profile update trajectory. This can be shown using induction.

We know from initialization that

$$x_i^{(l)}(0) = \sigma_i^{(1,l)}(\boldsymbol{x}_{S_i^{(1,l)}}(0)), \forall i, \forall l \in \{2, \ldots, q-1\},$$

$$x_j^{(l)}(0) = \sigma_j^{(q,l)}(\boldsymbol{x}_{S_j^{(q,l)}}(0)), \forall j, \forall l \in \{q, \ldots, L\}.$$

Then based on Eqn (A.12), we know that

$$\begin{aligned} & u_i^{(sl_1)}(x_i^{(1)}, \boldsymbol{x}_{I_i}^{(1)}(0), \ldots, x_{k_{iq}}^{(q)}(0)) = \overline{u}_i^{(sl_1)}(x_i^{(1)}, x_{\underline{I}_i}^{(1)}(0)) \\ \Leftrightarrow \quad & BR_i(\boldsymbol{x}_{I_i}^{(1)}(0), \ldots, x_{k_{iq}}^{(q)}(0), u_i^{(sl_1)}) = BR_i(x_{\underline{I}_i}^{(1)}(0), \overline{u}_i^{(sl_1)}), \end{aligned}$$

and thus when $t = 1$, $\boldsymbol{x}^{(1)}(t)$ are the same when applying HH-BRD in the original game and the MS-BRD in the new 2-level game. Similarly, $\boldsymbol{x}^{(q)}(1)$ are the same based on Eqn (A.13).

Suppose $\boldsymbol{x}^{(1)}(t)$ and $\boldsymbol{x}^{(q)}(t)$ are the same for the two dynamics for $t = 0, 1, \ldots, T$, we need to show that $\boldsymbol{x}^{(1)}(t)$ and $\boldsymbol{x}^{(q)}(t)$ are the same for $t = T+1$ to complete the proof.

Again, based on Eqn (A.12), we know that

$$\begin{aligned} & u_i^{(sl_1)}(x_i^{(1)}, \boldsymbol{x}_{I_i}^{(1)}(T), \ldots, x_{k_{iq}}^{(q)}(T)) = \overline{u}_i^{(sl_1)}(x_i^{(1)}, x_{\underline{I}_i}^{(1)}(T)) \\ \Leftrightarrow \quad & BR_i(\boldsymbol{x}_{I_i}^{(1)}(T), \ldots, x_{k_{iq}}^{(q)}(T), u_i^{(sl_1)}) = BR_i(x_{\underline{I}_i}^{(1)}(T), \overline{u}_i^{(sl_1)}), \end{aligned}$$

which implies $\boldsymbol{x}^{(1)}(T+1)$ are the same for the two dynamics and similarly $\boldsymbol{x}^{(q)}(T+1)$ are the same based on Eqn (A.13).

□

A.5 Data Generation for Numerical Experiments

We introduce the data generation procedures for both games with linear best response and non-linear best response in this part.

First of all, for both type of games, we create an adjacency matrix for each of the groups on every level. This matrix has 0 diagonal elements and for the off-diagonal elements, the existence of a directed edge subjects to the Bernoulli distribution where there is a fixed P_{exist}. Then if a directed edge exist, the edge weight is generated by choosing a value from $[0, 1]$ uniformly at random. Later, we will multiply these matrices with different scalars to adjust the values so that Assumption 3 holds. These matrices have 0 diagonal elements because they capture the dependencies of agents on each other, or equivalently, they are used to model the external impact the agents receive from the network. The internal impact are modeled by cost functions and marginal benefit terms that only depend on an agent's own action.

A.5.1 Linear Best Response Games

For games with linear best response, we generated a 2-level game with 100 groups and 10,000 level-1 agents. The adjacency matrix generation follows $P_{exist} = 0.1$, which creates a rather sparse network. Each level-2 group $S_k^{(2)}$ contains 100 members, and we use W_k to denote the corresponding adjacency matrix. We use V to denote the level-2 adjacency matrix. From Eqn (2.6), we know that for each level-1 agent, the utility function is

$$u_i(x_i^{(1)}, \boldsymbol{x}_{I_i}^{(1)}, \boldsymbol{x}_{I_{k_{i2}}}^{(2)}) = u_i^{(1)}(x_i^{(1)}, \boldsymbol{x}_{I_i}^{(1)}) + u_{k_{i2}}^{(2)}(x_{k_{i2}}^{(2)}, \boldsymbol{x}_{I_{k_{i2}}}^{(2)}),$$

where

$$u_i^{(1)}(x_i^{(1)}, \boldsymbol{x}_{I_i}^{(1)}) = b_i x_i^{(1)} + x_i^{(1)}\left(\sum_{j\in I_i}(W_{k_{i2}})_{r_i r_j} x_j^{(1)}\right) - c_i(x_i^{(1)})^2,$$

$$u_k^{(2)}(x_{k_{i2}}^{(2)}, \boldsymbol{x}_{I_{k_{i2}}}^{(2)}) = x_p^{(2)}\left(\sum_{p\neq k} V_{kp} x_p^{(2)}\right).$$

We choose the cost coefficients c_i to be large enough so that the $\Upsilon(F)$ satisfies the P_Υ condition(from Appendix A, strong diagonal dominance implies P_Υ condition). In the experiments, the $\rho(\Gamma)$(Se Appendix A for Γ) has a value between $[0.7, 0.8]$.

Then under the flat perspective, a level-1 agent $a_i^{(1)}$ has the following utility function

$$u_i^{flat}(x_i^{(1)}, x_{-i}^{(1)}) = b_i x_i^{(1)} + x_i^{(1)}\left(\sum_{j\neq i} W_{ij}^{flat} x_j^{(1)}\right) - c_i(x_i^{(1)})^2 + d_i,$$

where

$$d_i = \sum_{j \in I_i} x_j^{(1)} \Bigg(\sum_{p \notin S_{k_{i2}}^{(2)}} W_{jp}^{flat} x_p^{(1)} \Bigg),$$

$$W^{flat} = \begin{bmatrix} W_1 & V_{1,2} \cdot \mathbf{1} & \cdots & V_{1,100} \cdot \mathbf{1} \\ V_{2,1} \cdot \mathbf{1} & W_2 & \cdots & V_{2,100} \cdot \mathbf{1} \\ \vdots & \vdots & \ddots & \vdots \\ V_{100,1} \cdot \mathbf{1} & \cdots & V_{100,2} \cdot \mathbf{1} & W_{100} \end{bmatrix},$$

here $\mathbf{1}$ represents the all 1 matrix of suitable size(100×100).

A.5.2 General Best Response Games

For games with general(non-linear) best response, we generated data using the graphical game model similarly like the above. However, this time we use a mixed cost term that is a weighted sum of a quadratic component and an exponential component. Therefore, we can no longer represent the best response functions as linear functions and the best response computing now relies on gradient based optimization steps. In the experiments shown in the main article, the adjacency matrix is generated following $P_{exist} = 0.1$, which creates a sparse network. We also tried $P_{exist} = 1$ and the results on the dense networks are included in this part of the appendix.

We use $W_i^{(l)}$ to denote the adjacency matrix within $S_i^{(l)}$ and $W^{(L+1)}$ to denote the adjacency matrix between highest level agents. For the 2-level games with general best response, the utility components are set as follows

$$u_i^{(1)}(x_i^{(1)}, \boldsymbol{x}_{I_i}^{(1)}) = b_i x_i^{(1)} + x_i^{(1)} \Bigg(\sum_{j \in I_i} (W_{k_{i2}}^{(2)})_{r_i r_j} x_j^{(1)} \Bigg) - c_i (x_i^{(1)})^2 - e^{0.1 x_i^{(1)}},$$

$$u_i^{(2)}(x_i^{(2)}, \boldsymbol{x}_{I_i}^{(2)}) = x_i^{(2)} \Bigg(\sum_{j \neq i} (W_{k_{i3}}^{(3)})_{ij} x_j^{(2)} \Bigg) - |S_i^{(2)}| \cdot e^{0.1 x_i^{(2)} / |S_i^{(2)}|}.$$

For 3-level games with general best response, the components in level-1 and 2 remain the same, and the level-3 components are

$$u_i^{(2)}(x_i^{(3)}, \boldsymbol{x}_{I_i}^{(3)}) = x_i^{(3)} \Bigg(\sum_{j \neq i} W_{ij}^{(4)} x_j^{(3)} \Bigg) - |S_i^{(1,3)}| \cdot e^{0.1 x_i^{(3)} / |S_i^{(1,3)}|}.$$

For the 2-level games with linear/nonlinear best response, the utility components are set as follows

$$u_i^{(1)}(x_i^{(1)}, \boldsymbol{x}_{I_i}^{(1)}) = b_i x_i^{(1)} + x_i^{(1)} \left(\sum_{j \in I_i} (W_{k_{i2}}^{(2)})_{r_i r_j} x_j^{(1)} \right) - c_i (x_i^{(1)})^2,$$

$$u_i^{(2)}(x_i^{(2)}, \boldsymbol{x}_{I_i}^{(2)}) = x_i^{(2)} \left(\sum_{j \neq i} (W_{k_{i3}}^{(3)})_{ij} x_j^{(2)} \right) - |S_i^{(2)}| \cdot e^{0.1 x_i^{(2)} / |S_i^{(2)}|}.$$

Again, the adjacency matrix and the cost terms will be scaled to ensure that Assumption 3 holds, and in the experiments, the $\rho(\Gamma)$(Se Appendix A for Γ) has a value between $[0.7, 0.8]$.

Hyperparameter settings: besides the parameters in the graphical games, the parameter $h_i^{(l)}$ in the loss function updates in Eqn (2.10) is chosen arbitrarily. These parameters can also be referred to as "penalty parameters". In our experiments, the performance over these parameters are rather smooth under assumption 3. The hyperparameters $h_i^{(l)}$ are set to the same value on each level l. In the 2-level case, we perform a binary search on these hyperparameter, where each value is tested for 5 runs to see the average performance. For the 3-level case, we need to determine 2 hyperparameter values, and this is done by a fixed step size search performed iteratively on the two values. We tune the first one, each value is tested for 5 runs like the above, while fixing the second value, after that, we switch to the tuning of the second value and this process keeps iteratively. The parameters we used in the numerical experiments are

- 2-Level game: $h_i^{(2)} = 0.2, 0.1, 0.06, 0.03, 0.01$; for network sizes $30^2, 50^2, 100^2, 200^2, 500^2$ respectively. With tuning range $[0, 0.5]$.

- 3-Level game:

  For SH-BRD: $(h_i^{(2)}, h_j^{(3)}) = (0.65, 0.1)$, $(0.32, 0.03)$, $(0.2, 0.01)$, $(0.12, 0.006)$, $(0.04, 0.003)$; for network sizes $10^3, 20^3, 30^3, 50^3, 100^3$ respectively. With tuning range $[0, 0.5]^2$ and tuning step 0.002.

  For HH-BRD: $h_i^{(sl_1)} = 0.7, 0.3, 0.21, 0.125, 0.063$ for network sizes $10^3, 20^3, 30^3, 50^3, 100^3$ respectively. With tuning range $[0, 0.5]$

Under the current parameter settings, we still haven't bring out the best performances of SH-BRD, and HH-BRD. In act, the performance gap between the current setting and the optimal setting won't be too large since the best response steps are well-posed. And even with their sub-optimal performances, we have seen their advantages over other algorithms.

In [39], the authors mentioned an adaptive method to generate the penalty parameter matrix H which is generally not diagonal, that can speed up the problem solving steps. This will be an interesting direction to generalize our current algorithm when the best response functions become more ill-posed in the future.

A.5.3 CPU Specs:

- CPU: 6 cores, 12 threads, 2.60/4.50 GHz, 12MB Cache
- OS: Windows 10
- Software: Python 3.7
- RAM: 16 GB

A.5.4 Results on Dense Networks

Size	BRD	MS-BRD	SH-BRD
30^2	$(2.97\pm0.24)\times10^7$	$(9.91\pm0.81)\times10^5$	**(8.31±0.66)**$\times10^5$
50^2	$(2.41\pm0.22)\times10^8$	$(4.83\pm0.45)\times10^6$	**(3.27±0.30)**$\times10^6$
100^2	$(4.07\pm0.34)\times10^9$	$(4.07\pm0.34)\times10^7$	**(3.04±0.22)**$\times10^7$
200^2	$(6.66\pm0.62)\times10^{10}$	$(3.33\pm0.31)\times10^8$	**(2.44±0.17)**$\times10^8$
500^2	$(2.72\pm0.29)\times10^{12}$	$(5.53\pm0.49)\times10^9$	**(3.26±0.26)**$\times10^9$

Table A.1: Convergence and complexity (flops) comparison with linear best response under multiple initialization, dense network.

Size	BRD	MS-BRD	SH-BRD
30^2	0.99±0.03	0.49±0.02	**0.24±0.01**
50^2	22.80±0.05	1.83±0.06	**0.69±0.01**
100^2	1351±7	13.28±0.26	**4.70±0.06**
200^2	> 18000	159.9±0.8	**58.07±0.42**
500^2	nan	3505±54	**1286±20**

Table A.2: CPU times on a single machine on 2-Level games with general best response functions, dense network; All times are in seconds.

Size	BRD	MS-BRD	SH-BRD
30^2	1.63±0.12	0.57± 0.02	**0.028±0.002**
50^2	30.65±0.35	1.94±0.03	**0.051±0.003**
100^2	1660±3	13.93±0.25	**0.33±0.02**
200^2	> 18000	163.1±1.4	**1.32±0.04**
500^2	nan	3416±52	**29.37±0.91**

Table A.3: CPU times on a single machine for 2-Level, linear/nonlinear best-response games, dense network; All times are in seconds.

Size	BRD	MS-BRD	SH-BRD	HH-BRD
10^3	1.25±0.02	0.39±0.01	0.57±0.02	**0.34±0.01**
20^3	617.3±4.7	2.85±0.07	4.50±0.06	**2.56±0.06**
30^3	> 18000	10.25±0.25	17.87±0.14	**9.53±0.09**
50^3	nan	58.04±0.32	100.8±0.41	**51.86±0.24**
100^3	nan	926.8±6.4	2131±11	**780.9±3.0**

Table A.4: CPU times in seconds on a single machine on 3-Level, general best response games, dense network; All times are in seconds.

We can see that though the results in linear best response games are very different in sparse and dense networks, the results in games with non-linear best responses are quite similar in both types of networks. In games with linear best responses, the standard deviation results from different initialization. For the same game, one initial action profile's distance(measured in Euclidean norm) to the equilibrium point can be 20 times to the distance of another initial action profile. This results in different number of iterations of the algorithm before convergence. However, it only takes about 20% more iterations for a "distant" initial action profile to reach convergence, which shows that these algorithms have good convergence property under Assumptions 1-3. In games with non-linear best responses, the standard deviations of CPU times are relatively small(around 1%) compared to the mean values, and it shows that the performance of all algorithms are stable with a fixed initial action profile.

A.6 Algorithm Performances and Network Sizes

In this part, we present some results that show the algorithms' performances with different network sizes in 2-level games.

Figure A.1 shows the number of flops per iteration for the three algorithms in $I \times M$ games where I is the number of agents in each group and M the number of groups in the network. Both Algorithms 2 and 4 outperform Algorithm 1. Algorithm 4 generally has lower complexity per iteration compared to Algorithm 2 since it has less input in every sub-problem and the number of sub-problems are similar in Algorithm 2 and 4 when the group sizes are large. However, when group sizes are small compared to the number of groups, Algorithm 2 and 4 are similar per iteration.

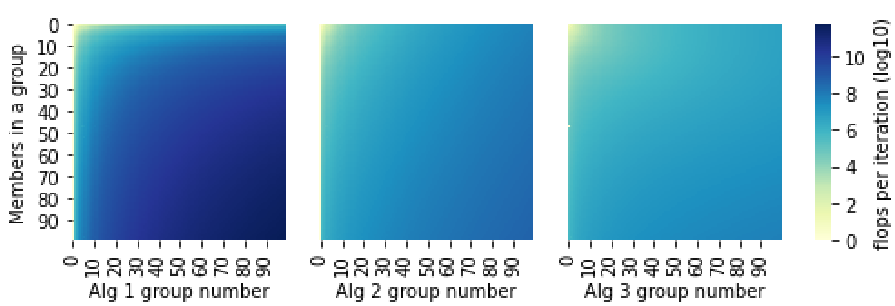


Figure A.1: Complexity per iteration for linear best response.

A.7 Reverse Engineer Multi-scale Structure

A question that naturally arises is whether sparsity in the network can be exploited when the multi-scale structure is not readily available. The utility function in Eqn (2.6) suggests that such reverse engineering is possible if the game satisfies:

1. An agent is either connected to all agents in another group or not connected to any agent in that group; If so, we can create a set of possible group partitions.

2. Based on the partition in the previous step, agents in one group have the same dependency on an agent in another group.

3. Based on the partition, we can represent the groups' aggregate actions from their members' actions using some aggregate functions.

4. Based on the partition, the original utility function of each agent can be separated to components on different levels, each component only based on the actions and dependencies on the corresponding level.

An example of the first condition is shown in Figs. A.2 and A.3. For the other conditions, the "flattened" utility functions used in Appendix A.5 are good examples.

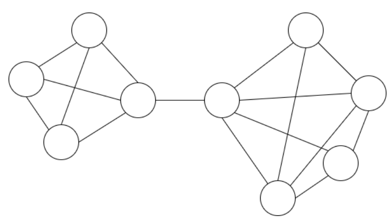


Figure A.2: Ungrouped.

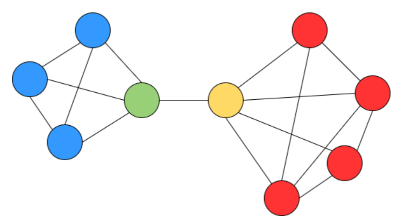


Figure A.3: Grouped.

A.8 Flow Charts of the Algorithms

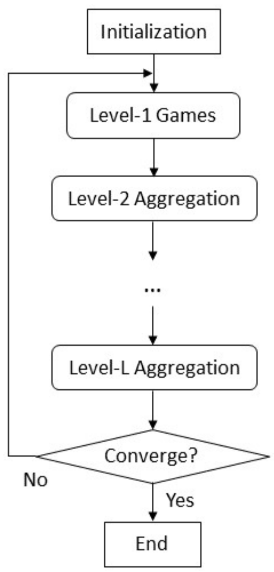


Figure A.4: MS-BRD

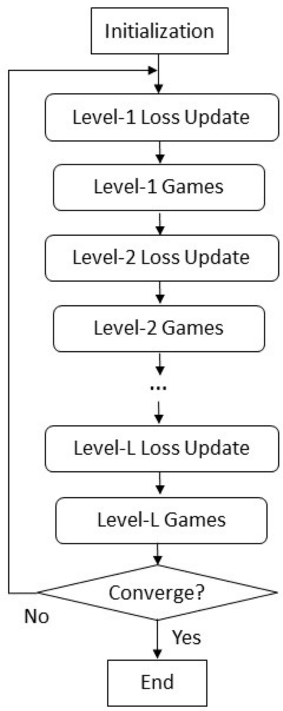


Figure A.5: SH-BRD

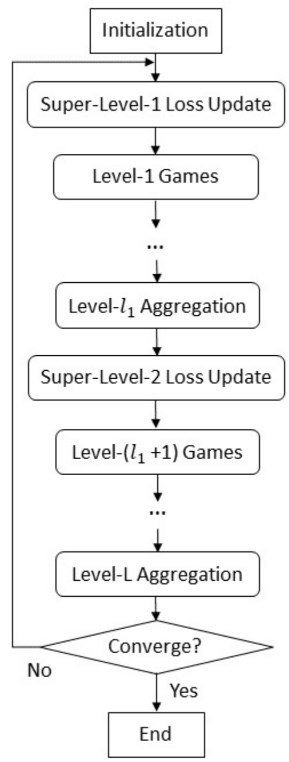


Figure A.6: HH-BRD

APPENDIX B

Chapter 3 Appendix

B.1 Concrete numbers in Figure 1-4

We use Figures 1-4 to illustrate the high level idea of the partitioning on structured multi-relational networks. In Figure 1 and 2, using the linear quadratic utility functions $u_i = b_i x_i + \sum_{j \neq i} g_{ij} x_i x_j - \frac{1}{2} x_i^2$, the adjacency matrices are

$$G = \begin{bmatrix} 0 & 0.6 & 0.2 \\ 0.6 & 0 & 0.2 \\ 0.2 & 0.2 & 0 \end{bmatrix}, \ G = \begin{bmatrix} 0 & 0 & 0.2 \\ 0 & 0 & 0.2 \\ 0.2 & 0.2 & 0 \end{bmatrix}$$

respectively. In Figure 3 and 4, when using the utility function

$$u_i = b_i x_i + \sum_{j \neq i} g_{ij}^{(1)} x_i^{(1)} x_j^{(1)} + \sum_{j \neq i} g_{ij}^{(2)} x_i^{(2)} x_j^{(2)} - \frac{1}{2}((x_i^{(1)})^2 + (x_i^{(2)})^2 + 0.2 x_i^{(1)} x_j^{(1)}),$$

the adjacency matrices are

$$G^{(1)} = \begin{bmatrix} 0 & 0.6 & 0 \\ 0.6 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \ G^{(2)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0.6 \\ 0 & 0.6 & 0 \end{bmatrix}, \ G = \begin{bmatrix} G^{(1)} & 0 \\ 0 & G^{(2)} \end{bmatrix}.$$

B.2 Proof of Lemma 1

Proof. We will prove that $\|\Upsilon^U_{\mathcal{P}_i, \mathcal{P}_j}\|_2 \geq \beta^S_{ij}$. The proof establishing $\|\Upsilon^U_{\mathcal{P}_i, \mathcal{P}_i}\|_2 \leq \alpha^S_i$ is similar.

From the definition of β^S_{ij} in Eqn (3.7) , we denote the action profile in Q that obtains the value of β^S_{ij} as $\hat{\boldsymbol{x}}(i, j)$, i.e.,

$$\beta^S_{ij} = \sup_{\boldsymbol{x} \in Q} \|\nabla_j F_i(\boldsymbol{x})\|_2 = \|\nabla_j F_i(\hat{\boldsymbol{x}}(i, j))\|_2,$$

where $\nabla_j F_i(\hat{\boldsymbol{x}}(i, j)) \in \mathbb{R}^{N_i \times N_j}$. We note here that since our action space Q is compact and ∇F is assumed to be continuous and differentiable on q, there exist maxima for $\|\nabla_j F_i(\boldsymbol{x})\|$ such that

$\hat{\boldsymbol{x}} \in Q$. Therefore, the supremum equals the maximum and can be achieved. Since we will only focus on a specific pair of (i, j) values where $i \neq j$ in this part, we will simply use $\hat{\boldsymbol{x}}$ to denote this action profile for convenience.

Also, based on the definition of β_{kl}^U, we have

$$\beta_{kl}^U = \sup_{\boldsymbol{x} \in Q} |\nabla_l F_k(\boldsymbol{x})| \geq |\nabla_l F_k(\hat{\boldsymbol{x}})|,$$

where $\nabla_l F_k(\hat{\boldsymbol{x}}) \in \mathbb{R}$.

Next, we perform Singular Value Decomposition on $\nabla_j F_i(\hat{\boldsymbol{x}})$ such that $\nabla_j F_i(\hat{\boldsymbol{x}}) = USV^T$. From this, we can get the left singular vector $\boldsymbol{u} \in \mathbb{R}^{N_i}$ and the right singular vector $\boldsymbol{v} \in \mathbb{R}^{N_j}$ that correspond to the largest singular value β_{ij}^S, where $\|\boldsymbol{u}\| = \|\boldsymbol{v}\| = 1$. Then

$$\boldsymbol{u}^T \nabla_j F_i(\hat{\boldsymbol{x}}) \boldsymbol{v} = \beta_{ij}^S \cdot \|\boldsymbol{u}\| \cdot \|\boldsymbol{v}\| = \beta_{ij}^S.$$

To proceed with the proof, we introduce the following matrix A obtained from matrix $\nabla_j F_i(\hat{\boldsymbol{x}}(i, j))$ by replacing every element with its absolute value, formally,

$$A = (|\nabla_l F_k(\hat{\boldsymbol{x}})|)_{k,l:a_k \in \mathcal{P}_i, a_l \in \mathcal{P}_j}.$$

We also introduce vectors $\boldsymbol{u}^+$ and $\boldsymbol{v}^+$, where

$$\boldsymbol{u}^+ = (|u_k|)_{k=1}^{N_i},\ \boldsymbol{v}^+ = (|v_l|)_{l=1}^{N_j}.$$

Then, $\|\boldsymbol{u}^+\| = \|\boldsymbol{v}^+\| = 1$, and it is easy to see that

$$(\boldsymbol{u}^+)^T A \boldsymbol{v}^+ \geq \boldsymbol{u}^T \nabla_j F_i(\hat{\boldsymbol{x}}) \boldsymbol{v},$$

Next, consider the product $\boldsymbol{u}^T \Upsilon_{\mathcal{P}_i,\mathcal{P}_j}^U \boldsymbol{v}$. We denote the index in $\Upsilon_{\mathcal{P}_i,\mathcal{P}_j}^U$ that corresponds to the element u_p and v_q as k_p and l_q respectively, so that,

$$\boldsymbol{u}^T \Upsilon_{\mathcal{P}_i,\mathcal{P}_j}^U \boldsymbol{v} = \sum_{p=1}^{N_i} \sum_{q=1}^{N_j} \Upsilon_{k_p l_q}^U \cdot u_p \cdot v_q\ .$$

then we have

$$
\begin{aligned}
\|\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_j}\| &= \|-\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_j}\| \cdot \|\boldsymbol{u}^+\| \cdot \|\boldsymbol{v}^+\| \\
&\geq (\boldsymbol{u}^+)^T(-\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_j})\boldsymbol{v}^+ \\
&= -\sum_{p=1}^{N_i}\sum_{q=1}^{N_j}\Upsilon^U_{k_pl_q}\cdot|u_p|\cdot|v_q| \\
&= \sum_{p=1}^{N_i}\sum_{q=1}^{N_j}\beta^U_{k_pl_q}\cdot|u_p|\cdot|v_q| \\
&\geq \sum_{p=1}^{N_i}\sum_{q=1}^{N_j}|\nabla_{l_q}F_{k_p}(\hat{\boldsymbol{x}})|\cdot|u_p|\cdot|v_q| \\
&= (\boldsymbol{u}^+)^T A\boldsymbol{v}^+ \\
&\geq \boldsymbol{u}^T\nabla_j F_i(\hat{\boldsymbol{x}})\boldsymbol{v} \\
&= \beta^S_{ij}.
\end{aligned}
$$

This completes the proof.

□

B.3 Proof of Theorem 5

Proof. Given the community level partition $Q = \prod_{i=1}^{M} Q_i$, we denote $\nabla F_i(\boldsymbol{z}) = ((\nabla_i F_j(\boldsymbol{z}))_{j=1}^M)^T \in \mathbb{R}^{N_i \times N}$.

We use the notation $L(\boldsymbol{x}, \boldsymbol{y})$ to denote the line segment between two points $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\mathbb{R}^N$. Formally,

$$L(\boldsymbol{x}, \boldsymbol{y}) = \{\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y} : 0 \leq \alpha \leq 1\}.$$

Under Assumption 4 in Section II, $F_i : Q \to \mathbb{R}^{N_i}, Q \subseteq \mathbb{R}^N$ is continuously differentiable on Q, and for $\forall \boldsymbol{x}, \boldsymbol{y}$ in Q_i, $L(\boldsymbol{x}, \boldsymbol{y}) \subseteq Q_i$. According to [5, Theorem 12.9] we know that for every vector $\boldsymbol{a}$ in $\mathbb{R}^{N_i}$, there is a point $\boldsymbol{z} \in L(\boldsymbol{x}, \boldsymbol{y})$ such that:

$$\boldsymbol{a} \cdot (F_i(\boldsymbol{x}) - F_i(\boldsymbol{y})) = \boldsymbol{a} \cdot (\nabla F_i(\boldsymbol{z})(\boldsymbol{x} - \boldsymbol{y})). \tag{B.1}$$

Let $\boldsymbol{a}$ in equation (B.1) be $(\boldsymbol{x}_i - \boldsymbol{y}_i)^T$, and denote $\boldsymbol{l} = (l_j)_{j=1}^M$, where $l_j = \|\boldsymbol{x}_j - \boldsymbol{y}_j\|_2, \forall j \in \mathbb{N}[1, M]$, then,

$$\begin{aligned}
& (\boldsymbol{x}_i - \boldsymbol{y}_i)^T (F_i(\boldsymbol{x}) - F_i(\boldsymbol{y})) \\
= \quad & (\boldsymbol{x}_i - \boldsymbol{y}_i)^T (\nabla F_i(\boldsymbol{z})(\boldsymbol{x} - \boldsymbol{y})) \\
= \quad & (\boldsymbol{x}_i - \boldsymbol{y}_i)^T [\sum_{j=1}^{M} \nabla_i F_j(\boldsymbol{z})(\boldsymbol{x}_j - \boldsymbol{y}_j)] \\
\geq \quad & (\boldsymbol{x}_i - \boldsymbol{y}_i)^T \nabla_i F_i(\boldsymbol{z})(\boldsymbol{x}_i - \boldsymbol{y}_i) \\
& - |\sum_{j \neq i} (\boldsymbol{x}_i - \boldsymbol{y}_i)^T \nabla_i F_j(\boldsymbol{z})(\boldsymbol{x}_j - \boldsymbol{y}_j)| \\
\geq \quad & \alpha_i^S (l_i)^2 - \sum_{j \neq i} \beta_{ij}^S \cdot l_i \cdot l_j \\
= \quad & l_i \cdot (\Upsilon^S \boldsymbol{l})_i
\end{aligned} \tag{B.2}$$

By [23, Theorem 3.3.4(b)], a real square matrix $M \in \mathbb{R}^{n \times n}$ is a P-matrix if it satisfies $\forall \boldsymbol{l} \in \mathbb{R}^n$

$$\max_{i \in \mathbb{N}[1,M]} l_i (M\boldsymbol{l})_i > 0$$

Denote $b = \max_{i \in \mathbb{N}[1,M]} \frac{l_i \cdot [\Upsilon^C \boldsymbol{l}]_i}{\|\boldsymbol{l}\|_2^2} > 0$. Then, we have

$$\max_{i \in \mathbb{N}[1,M]} (\boldsymbol{x}_i - \boldsymbol{y}_i)^T (F_i(\boldsymbol{x}) - F_i(\boldsymbol{y})) \geq \max_{i \in \mathbb{N}[1,M]} l_i \cdot [\Upsilon^C \boldsymbol{l}]_i \geq b \cdot \|\boldsymbol{x} - \boldsymbol{y}\|_2^2$$

which, according to Definition 2.(b), shows that F satisfies uniform block P-condition. Therefore, by [79, Proposition 2 part (b)] and [28, Proposition 3.5.10 part (b)], the Nash equilibrium is unique.

Finally, we show that the two conditions

1. Υ^S is a P-matrix,

2. $\Upsilon^U_{\mathcal{P}_i, \mathcal{P}_i}$ are P-matrices,

are sufficient for Υ^U to be a P-matrix. Using Lemma 1,

$$\begin{aligned}
& \Upsilon^S \in \mathbb{R}^{M\times M} \text{ is a P-matrix} \\
\Leftrightarrow\quad & \max_j \; l_j(\Upsilon^S \boldsymbol{l})_j > 0, \forall \boldsymbol{l} \neq \mathbf{0}, \boldsymbol{l} \in \mathbb{R}^M, \text{let } i = \arg\max_j l_j(\Upsilon^C \boldsymbol{l})_j, \\
\Leftrightarrow\quad & \quad \boldsymbol{x}_i(\Upsilon^U_{\mathcal{P}_i,:} \cdot \boldsymbol{x})_{\mathcal{P}_i} \\
& \geq \; \|\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_i}\| \cdot \|\boldsymbol{x}_i\|^2 - \sum_{j\neq i} \|\Upsilon^U_{\mathcal{P}_i,\mathcal{P}_j}\| \cdot \|\boldsymbol{x}_i\| \cdot \|\boldsymbol{x}_j\| \\
& \geq \; l_i \alpha^S_{ii} l_i - \sum_{j\neq i} l_i \beta^C_{ij} l_i \; = \; l_i(\Upsilon^C \boldsymbol{l})_i \; > \; 0, \\
\Rightarrow\quad & \max_{k\in S_i} \; x_k(\Upsilon^U \boldsymbol{x})_k > 0, \\
\Rightarrow\quad & \max_k \; x_k(\Upsilon^U \boldsymbol{x})_k > 0, \;\; \forall \boldsymbol{x} \neq \mathbf{0}, \boldsymbol{x} \in \mathbb{R}^N, \\
\Leftrightarrow\quad & \Upsilon^U \in \mathbb{R}^{N\times N} \text{ is a P-matrix}
\end{aligned}$$

□

B.4 Proof of Corollary 2

Proof. We can see that $I + \Gamma^S \succ \mathbf{0}$, iff $\lambda_1(\Gamma^S) > -1$, and from symmetry, $I + \Gamma^S$ is a P-matrix. The Υ^S matrix can be obtained by scaling the ith row of $I + \Gamma^S$ by α_i^S, which is a positive number. Therefore, the determinant of every principal minor of Υ^S has the same sign as the determinant of the corresponding principal minor of $I + \Gamma^S$, and thus Υ^S is also a P-matrix. By Theorem 5, the Nash equilibrium is unique and the sufficiency holds. □

B.5 Proof of Theorem 7

We prove for the $K = 1$ case and the proof generates to an arbitrary K.

Proof. When we consider borderline inactive and active groups, the set of all their members are a superset of the union of the borderline inactive and active set of agents.

We denote

$$Y = \{a_k | a_k \in \mathcal{P}_i, s.t. \mathcal{P}_i \in A_S(\boldsymbol{x}^*) \cup B_S(\boldsymbol{x}^*)\}$$

as the set of all agents that belong to communities in $A_S(\boldsymbol{x}^*) \cup B_S(\boldsymbol{x}^*)$, then we have $A(\boldsymbol{x}^*) \bigcup B(\boldsymbol{x}^*) \subseteq \mathcal{S}$. Similar to Theorem 6, we denote $\triangledown_{\mathcal{S}} F_{\mathcal{S}}(\boldsymbol{x}^*)$ as a sub-matrix of $\triangledown F(\boldsymbol{x}^*)$ whose columns and rows correspond to the agents in set $\mathcal{S}$.

Our proof proceeds by the following logic:

$$G^S(\boldsymbol{x}^*) \succ \mathbf{0} \Rightarrow \nabla_{\mathcal{S}} F_{\mathcal{S}}(\boldsymbol{x}^*) \succ \mathbf{0} \Rightarrow \nabla_{A,B} F_{A,B}(\boldsymbol{x}^*) \succ \mathbf{0}.$$

For the first part, we adopt techniques similar to those we used in Appendix B.3. For $\forall \boldsymbol{v} \in \mathbb{R}_{\geq 0}^{|S|}$, we denote $\boldsymbol{l} = (l_i)_{i=1}^Z \in \mathbb{R}_{\geq 0}^Z$, where $l_i = \|\boldsymbol{v}_i\|_2, \forall i \in \mathbb{N}[1, Z]$. Then we have

$$\begin{aligned}
& \boldsymbol{v}^T \nabla_Y F_{\mathcal{S}}(\boldsymbol{x}^*) \boldsymbol{v} \\
= \quad & \sum_{i=1}^{Z} \sum_{j=1}^{Z} \boldsymbol{z}_j^T (\nabla_j F_i(\boldsymbol{x}^*)) \boldsymbol{v}_i \\
= \quad & \sum_{i=1}^{Z} \boldsymbol{v}_i^T (\nabla_i F_i(\boldsymbol{x}^*)) \boldsymbol{v}_i + \sum_{i=1}^{Z} \sum_{j=1, j \neq i}^{Z} \boldsymbol{v}_j^T (\nabla_j F_i(\boldsymbol{x}^*)) \boldsymbol{v}_i \\
\geq \quad & \sum_{i=1}^{Z} \theta_i^S \cdot \|\boldsymbol{v}_i\|_2^2 - \sum_{i=1}^{Z} \sum_{j=1, j \neq i}^{Z} \delta_{ij}^S \cdot \|\boldsymbol{v}_i\|_2 \cdot \|\boldsymbol{v}_j\|_2 \\
= \quad & \sum_{i=1}^{Z} \theta_i^S \cdot {l_i}^2 - \sum_{i=1}^{Z} \sum_{j=1, j \neq i}^{Z} \delta_{ij}^S \cdot l_i \cdot l_j \\
= \quad & \boldsymbol{l}^T G^S(\boldsymbol{x}^*) \boldsymbol{l} \ > \ 0
\end{aligned}$$

which shows $\nabla_{\mathcal{S}} F_{\mathcal{S}}(\boldsymbol{x}^*) \succ \mathbf{0}$ and completes the proof of the first part.

For the second part, we know from the Sylvester's criterion that an $N \times N$ Hermitian matrix (for real valued matrices, it is symmetric) is positive definite if and only if every leading principal component of it (the top left $k \times k$ sub-matrices, for $k = 1, \ldots, N$) has a positive determinant. Since $A(\boldsymbol{x}^*) \bigcup B(\boldsymbol{x}^*) \subseteq Y$, we know that $\nabla_{A,B} F_{A,B}(\boldsymbol{x}^*)$ is a leading principal minor of $\nabla_Y F_Y(\boldsymbol{x}^*)$ and thus $\nabla_{A,B} F_{A,B}(\boldsymbol{x}^*) \succ \mathbf{0}$, which completes the proof. □

B.6 Proof of Theorem 8

Proof. Since $\Upsilon^S \succ \mathbf{0}$, we know from Theorem 5 that the Nash equilibrium is unique. It remains to prove that this NE $\boldsymbol{x}^*$ is stable.

We denote $\Upsilon_{A,B}^S \in \mathbf{Z} \times \mathbf{Z}$ as the principal minor of Υ^S by picking out all the rows and columns corresponding to groups in $A_S(\boldsymbol{x}^*) \bigcup B_S(\boldsymbol{x}^*)$, and thus $\Upsilon_{A,B}^S \succ \mathbf{0}$.

Without loss of generality, suppose $\mathcal{P}_1, \mathcal{P}_2 \in A_S(\boldsymbol{x}^*) \bigcup B_S(\boldsymbol{x}^*)$, then we know their new indices in $G^S(\boldsymbol{x}^*)$ remain unchanged. We know from the definition that $\alpha_1^S \leq \theta_1^S$ and $\beta_{12}^S \geq \delta_{12}^S$ (similar comparison can generalize to all other elements), and thus for $\forall \boldsymbol{v} \in \mathbf{R}^Z$, we have

$$\begin{aligned}
&\boldsymbol{v}^T G^S(\boldsymbol{x}^*)\boldsymbol{v} \\
&= \sum_{i=1}^{Z}\sum_{j=1}^{Z} G^S_{ij}(\boldsymbol{x}^*)\cdot v_i \cdot v_j \\
&= \sum_{i=1}^{Z} \theta_i^S \cdot v_i^2 - \sum_{i=1}^{Z}\sum_{j=1, j\neq i}^{Z} \delta_{ij}\cdot v_i \cdot v_j \\
&\geq \sum_{i=1}^{Z} \alpha_i^S \cdot v_i^2 - \sum_{i=1}^{Z}\sum_{j=1, j\neq i}^{Z} \beta_{ij}\cdot v_i \cdot v_j \\
&= \boldsymbol{v}^T \Upsilon^S_{A,B}\boldsymbol{v} \; > \; 0,
\end{aligned}$$

which shows $G^S(\boldsymbol{x}^*) \succ \mathbf{0}$, and thus $\boldsymbol{x}^*$ is stable. □

B.7 Proof of Theorem 9

Proof. First of all, when $\boldsymbol{e} = \mathbf{1}, \forall t > 0$, $D^{out}_{max}(\mathbf{1}) < 1$ implies

$$\alpha_i^S > \sum_{j\neq i} \beta_{ij}^S, \forall i = 1, \dots, M,$$

and this shows that the Υ^S matrix is diagonally row dominant, and thus is a P-matrix [93] (Generating method 4.1, this matrix is a *positively stable* P-matrix).

For an arbitrary $\boldsymbol{e} \succ \mathbf{0}$, we define the following matrix E^S, where the ith column of E^S is equal to e_i times the ith column of Υ^S. Then since $e_i > 0$, every principal minor's determinant of E^S is equivalent to the corresponding principal minor's determinant of Υ^S and thus E^S is a P-matrix iff Υ^S is a P-matrix. Then, it's easy to see that $D^{out}_{max}(\boldsymbol{e}) < 1$ implies the diagonal row dominance of E^S, which is sufficient to show E^S is a P-matrix. This completes the proof of the out-degree part and the in degree part is similar. Moreover, the stability result follows from Theorem 8 and the in and out degree parts of this theorem.

□

B.8 Complexity of condition verification in Theorem 5

B.9 Supplementary reasoning in Section 3.7

We begin with the claim that the upper bounds are large enough for the study of sufficiency gaps.

Size	Unstructured (sec)	Structured (sec)
10×10	$1.08{\times}10^{35}$	$\mathbf{1.26}{\times}10^{6}$
20×20	$1.39{\times}10^{127}$	$\mathbf{1.69}{\times}10^{10}$
50×50	$4.90{\times}10^{761}$	$\mathbf{6.34}{\times}10^{20}$

Table B.1: Verification complexity in CPU times of conditions in Theorem 5 over number of agents.

When the parameters that control the external impact are all chosen at their normalized upper bound, the expected sum of the off-diagonal elements will be twice as high as the mean value of the internal impact. For our choices of $S^{in}_{low}, S^{in}_{high}, S^{out}_{low}, S^{out}_{high}$, the Υ^U will not be a P-matrix with probability 1 and thus Υ^S will not be a P-matrix. For example, we assume that Υ^U is symmetric, and for a matrix with positive diagonal elements and non-positive off-diagonal elements, as long as the sum of the absolute values of the off-diagonal elements is greater than the sum of the diagonal elements (the sum of all its elements is negative), then we know $\mathbf{1}^T \Upsilon^U \mathbf{1} < 0$ and it is not positive definite and thus not a P-matrix. Increasing such upper bounds will only increase the upper right area above the curve formed by dark cells.

For the parameters that control the internal impact variances, the argument is similar, when the values of $c^{high}_{\mathcal{P}_i} - c^{low}_{\mathcal{P}_i}$ are $c^{high} - c^{low}$ close to 1, then the agent with the lowest internal impact will have an internal impact close to 0 and Υ^U will not be a P-matrix. Figures B.3 and B.4 shows the comparison after we double such normalized upper bounds, and support our argument above.

Next, we elaborate more on the “critical values” in Figures 3.11 and 3.12.

If agents have homogeneous internal impact, i.e., α^U_k are constant, then when $\mathbf{E}[\sum_{k \neq l} \beta^U_{kl}]$ is about 10% higher than $\sum_{k=1}^N \alpha^U_k$, the games are most likely to have sufficiency gaps. We discussed above that when Υ^U is symmetric, and $\sum_{k \neq l} \beta^U_{kl} > \sum_{k=1}^N \alpha^U_k$, then the unstructured condition is not satisfied, which is sufficient to conclude that the structured condition does not hold either. But when it’s $\mathbf{E}[\sum_{k \neq l} \beta^U_{kl}] > \sum_{k=1}^N \alpha^U_k$, there are realizations (sample games under our choices of $S^{in}_{low}, S^{in}_{high}, S^{out}_{low}, S^{out}_{high}$) that have $\sum_{k \neq l} \beta^U_{kl} < \sum_{k=1}^N \alpha^U_k$ and thus are possible candidates for causing sufficiency gaps between the two sets of conditions. However, if $\mathbf{E}[\sum_{k \neq l} \beta^U_{kl}]$ is sufficiently larger (30% or more) $\sum_{k=1}^N \alpha^U_k$, all sample games will have neither set of conditions satisfied, which corresponds to the top right corner of the figures.

B.10 Sufficiency gap heatmaps for multipartite graphs

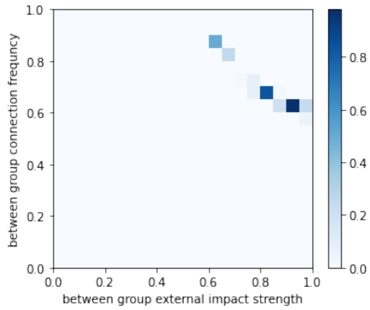


Figure B.1: Sufficiency gap frequency over the between group external impact, frequency 0.83%.

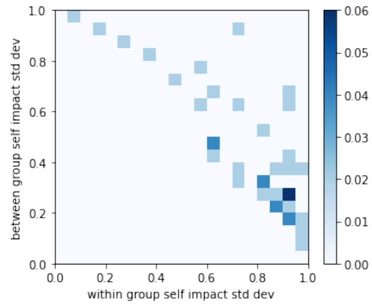


Figure B.2: Sufficiency gap frequency over the internal impact variances, with weak external impact, frequency 0.18%.

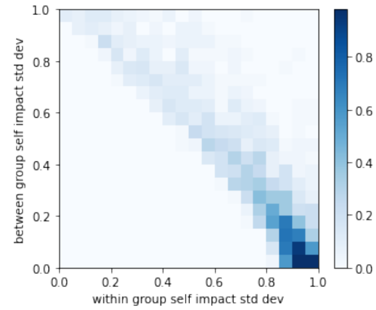


Figure B.3: Sufficiency gap frequency over the internal impact variances, with medium external impact, frequency 6.31%.

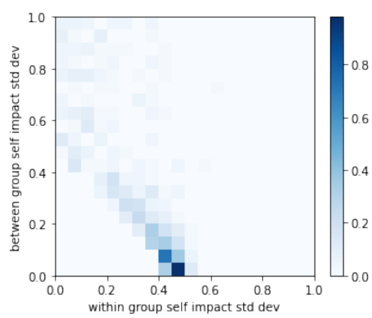


Figure B.4: Sufficiency gap frequency over the internal impact variances, with medium external impact, frequency 2.09%, doubled upper bounds.

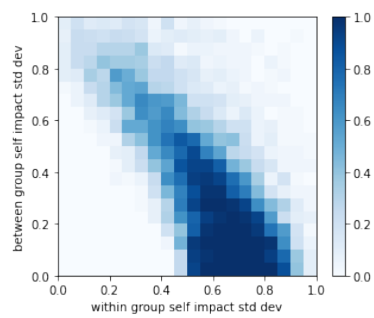


Figure B.5: Sufficiency gap frequency over the internal impact variances, with strong external impact, frequency 26.93%.

APPENDIX C

Chapter 4 Appendix

C.1 Derivation of Eqn (4.8)

$$\begin{aligned}
U_k(\boldsymbol{x}^*, \boldsymbol{y}_{S_k}) &= (\boldsymbol{x}^*_{S_k})^T(-\frac{1}{2}I + G_{S_k,S_k})\boldsymbol{x}^*_{S_k} + \sum_{l\neq k}(\boldsymbol{x}^*_{S_k})^T G_{S_k,S_l}\boldsymbol{x}^*_{S_l} + (\boldsymbol{b}_{S_k} + \boldsymbol{y}_{S_k})^T\boldsymbol{x}^*_{S_k} \\
&= (\boldsymbol{x}^*_{S_k})^T(-I + G_{S_k,S_k})\boldsymbol{x}^*_{S_k} + \frac{1}{2}(\boldsymbol{x}^*_{S_k})^T\boldsymbol{x}^*_{S_k} + \sum_{l\neq k}(\boldsymbol{x}^*_{S_k})^T G_{S_k,S_l}\boldsymbol{x}^*_{S_l} + (\boldsymbol{b}_{S_k} + \boldsymbol{y}_{S_k})^T\boldsymbol{x}^*_{S_k} \\
&= \frac{1}{2}(\boldsymbol{x}^*_{S_k})^T\boldsymbol{x}^*_{S_k}.
\end{aligned}$$

C.2 Proof of Theorem 10 and Theorem 11

Here we introduce some concepts and definitions related to the best responses and the variational inequality(VI) problem.

An agent's best response is the set of strategies that maximizes its utility given the actions taken by all the other agents. Formally, the best response is a set defined by

$$BR_i(\boldsymbol{x}_{-i}, u_i) = \underset{x_i}{\text{argmax}}\ u_i(x_i, \boldsymbol{x}_{-i}). \tag{C.1}$$

Clearly, an NE of a game is a fixed point of this best response correspondence. One important tool that is useful for analyzing the uniqueness of NE is *Variational Inequalities (VI)*. To establish the connection between NE and VI we assume the utility functions $u_i, \forall i = 1, \ldots, N$, for agents $a_1, \ldots, a_n$ are continuously twice differentiable. Let $K = \prod_{i=1}^N K_i$ and define $F : \mathbb{R}^N \to \mathbb{R}^N$ as follows:

$$F(\boldsymbol{x}) := \Big(-\nabla_{x_i} u_i(\boldsymbol{x})\Big)_{i=1}^N. \tag{C.2}$$

Then $\boldsymbol{x}^*$ is said to be a solution to VI(K, F) if and only if

$$(\boldsymbol{x} - \boldsymbol{x}^*)^T F(\boldsymbol{x}^*) \geq 0,\ \forall \boldsymbol{x} \in K\ . \tag{C.3}$$

In other words, the solution set to $\mathrm{VI}(K, F)$ is equivalent to the set of NE of the game. The following condition can guarantee the uniqueness of NE and the convergence of BRD.

Definition 16. ***The*** P_Υ ***condition****: We denote*

$$\alpha_i(F) = \inf_{\boldsymbol{x}\in K} \|\nabla_i F_i\|_2,\ \beta_{i,j}(F) = \sup_{\boldsymbol{x}\in K} \|\nabla_j F_i\|_2, i \neq j.$$

The Υ *matrix generated from* $F : \mathbb{R}^N \to \mathbb{R}^N$ *is given as follows*

$$\Upsilon(F) = \begin{bmatrix} \alpha_1(F) & -\beta_{1,2}(F) & \cdots & -\beta_{1,N}(F) \\ -\beta_{2,1}(F) & \alpha_2(F) & \cdots & -\beta_{2,N}(F) \\ \vdots & \vdots & \ddots & \vdots \\ -\beta_{N,1}(F) & -\beta_{N,2}(F) & \cdots & \alpha_N(F) \end{bmatrix}.$$

If $\Upsilon(F)$ *is a P-matrix, that is, if all of its principal components have a positive determinant, then we say* F *satisfies the* P_Υ *condition.*

In [89], the authors showed that if F *satisfies the* P_Υ *condition,* F *is strongly monotone on* K*, and VI*(K, F) *has a unique solution. Moreover, the BRD (both synchronous and asynchronous) converges to the unique NE.*

We will show the corresponding Υ matrices in (P-C) and (P-NC-alt) are P-matrices.

Proof. We first prove that this is true for the cooperative intervention game. We first fit the planners' game into the VI framework, where Q clearly is the action space, and we can similarly define the operator as

$$F(\boldsymbol{y}) := \Big(- \nabla_{y_{S_k}} W(\boldsymbol{y}) \Big)_{k=1}^{M},$$

Then we can define

$$\alpha_k(F) = \inf_{\boldsymbol{y}\in Q} \|\nabla_k F_k\|_2 = \|A_{S_k,S_k}\|_2,$$

$$\beta_{k,l}(F) = \sup_{\boldsymbol{y}\in Q} \|\nabla_l F_k\|_2 = \|A_{S_k,S_l}\|_2$$

and have the corresponding Υ matrix such that

$$A \succ \mathbf{0} \Leftrightarrow \boldsymbol{l}^T \Upsilon(F) \boldsymbol{l} \geq \boldsymbol{y}^T A \boldsymbol{y} > 0, \forall \boldsymbol{y} \Rightarrow \Upsilon(F) \succ \mathbf{0},$$

where $\boldsymbol{l} \in \mathbf{R}^K, l_k = \|\boldsymbol{y}_k\|_2$.

$\Upsilon \succ \mathbf{0}$ if and only if it's a P-matrix, and thus there is a unique equilibrium in the cooperative planners' intervention game and Algorithm 5 converges to it. $A \succ \mathbf{0}$ and Q is convex and compact also implies the budget tightness.

Similarly, we can fit the non-cooperative planners' intervention game into the variational inequality framework, where Q is the action space and the operator is

$$F(\boldsymbol{y}) := \Big(- \nabla_{y_{S_k}} W_k(\boldsymbol{y}) \Big)_{k=1}^{M},$$

then the Υ matrix is(also shown in proposition 1)

$$\Upsilon(F) = \tilde{A} \succ \mathbf{0},$$

so there is a unique equilibrium in the cooperative planners' intervention game and Algorithm 5 converges to it. $\tilde{A} \succ \mathbf{0}$ and Q is convex and compact also implies the budget tightness.

It remains to prove the $\tilde{A} \succ \mathbf{0}$ part above. We note that $A_{S_k,S_k} \succ \mathbf{0}, \forall k$ since they are principal minors of A and $A \succ \mathbf{0}$. Therefore, for $\forall \boldsymbol{z} \in \mathbf{R}^N$, $\boldsymbol{z} \neq \mathbf{0}$, we have

$$2\boldsymbol{z}^T \tilde{A} \boldsymbol{z} = \boldsymbol{z}^T A \boldsymbol{z} + \sum_{k=1}^{M} \boldsymbol{z}_{S_k}^T A_{S_k,S_k} \boldsymbol{z}_{S_k} > 0 \Leftrightarrow \tilde{A} \succ \mathbf{0}.$$

□

C.3 Proof of Proposition 1

Proof. We can write out the first order derivative in Eqn (4.9) for planner p_k in non-cooperative intervention game,

$$\nabla_{\boldsymbol{y}_{S_k}} W_k(\boldsymbol{y}) = A_{S_k,S_k}(\boldsymbol{y}_{S_k} + \boldsymbol{b}_{S_k}) + \frac{1}{2} \sum_{l \neq k} A_{S_k,S_l}(\boldsymbol{y}_{S_l} + \boldsymbol{b}_{S_l}).$$

Similarly, we can write out the first order derivative in Eqn (4.10) for planner p_k in non-cooperative intervention game,

$$\nabla_{\boldsymbol{y}_{S_k}} W(\boldsymbol{y}) = A_{S_k,S_k}(\boldsymbol{y}_{S_k} + \boldsymbol{b}_{S_k}) + \sum_{l \neq k} A_{S_k,S_l}(\boldsymbol{y}_{S_l} + \boldsymbol{b}_{S_l}).$$

So we can see that if the planners cooperatively play an intervention game such that the A matrix is replaced by the $\tilde{A}$ matrix, we have

$$\begin{aligned}\nabla_{\boldsymbol{y}_{S_k}} \tilde{W}(\boldsymbol{y}) &= \tilde{A}_{S_k,S_k}(\boldsymbol{y}_{S_k} + \boldsymbol{b}_{S_k}) + \sum_{l \neq k} \tilde{A}_{S_k,S_l}(\boldsymbol{y}_{S_l} + \boldsymbol{b}_{S_l}) \\ &= A_{S_k,S_k}(\boldsymbol{y}_{S_k} + \boldsymbol{b}_{S_k}) + \frac{1}{2}\sum_{l \neq k} A_{S_k,S_l}(\boldsymbol{y}_{S_l} + \boldsymbol{b}_{S_l}) \\ &= \nabla_{\boldsymbol{y}_{S_k}} W_k(\boldsymbol{y}).\end{aligned}$$

Since in both Eqn (4.9) and (4.12), every planner has the same gradient $\forall \boldsymbol{y} \in Q$, and the intervention action space are the same, we can conclude that solving the original non-cooperative intervention game and the alternative cooperative intervention game with $\tilde{A}$ are equivalent. In other words, for any arbitrary starting point, the trajectories of BRD will always be the same for the planners, and will converge to the same unique equilibrium in these two problems and thus they are equivalent. □

C.4 Direction of the optimal intervention

We first introduce the cosine similarity that defines the similarity of of two vectors in their directions, formally,

$$\rho(\boldsymbol{r}, \boldsymbol{s}) = \frac{\boldsymbol{r}^T \boldsymbol{s}}{\|\boldsymbol{r}\|_2 \|\boldsymbol{s}\|_2}, \tag{C.4}$$

when $\rho(\boldsymbol{r}, \boldsymbol{s}) = 1$, the two vectors have the same direction and $\boldsymbol{r} = \alpha \boldsymbol{s}$ for some $\alpha > 0$.

For non-cooperative planners, we have the following results on the optimal intervention.

Proposition 3. *For an arbitrary fixed budget constraint allocation, when $\boldsymbol{b} = \boldsymbol{0}$, $\rho(\boldsymbol{y}^*, \boldsymbol{v}_{max}) = 1$, otherwise if $C_k \gg \|\boldsymbol{b}_{S_k}\|_2^2, \forall k$, $\rho(\boldsymbol{y}^*, \boldsymbol{v}_{max}) \approx 1$, where $\boldsymbol{y}^*$ is the non-cooperative optimal intervention to the problem in Eqn (4.4), and $\boldsymbol{v}_{max}$ is the eigenvector that corresponds to the largest eigenvalue of $\tilde{B}$ matrix, which is defined as*

$$\tilde{B} = \begin{bmatrix} B_{S_1,S_1} & \frac{1}{2}B_{S_1,S_2} & \cdots & \frac{1}{2}B_{S_1,S_M} \\ \frac{1}{2}B_{S_2,S_1} & B_{S_2,S_2} & \cdots & \frac{1}{2}B_{S_2,S_M} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2}B_{S_M,S_1} & \frac{1}{2}B_{S_M,S_2} & \cdots & B_{S_M,S_M} \end{bmatrix},$$

where $B = A^{1/2} D A^{1/2}$, $D = diag((\frac{1}{\lambda_k^*}\boldsymbol{1}_{N_k})_{k=1}^M)$.

Proof. We begin with the case where $\boldsymbol{b} = \boldsymbol{0}$, and consider the following non-cooperative objective for p_k,

$$\begin{aligned} &\text{maximize} \;\; \hat{W}_k(\boldsymbol{y}) = \frac{1}{2}\|(B^{1/2}\boldsymbol{y})_{S_k}\|_2^2 = \frac{1}{2\lambda_k^*}\|\boldsymbol{x}_{S_k}^*\|_2^2 \\ &\text{subject to} \;\; \sum_{i:a_i\in S_k}(y_i)^2 \leq C_k, \end{aligned} \tag{C.5}$$

clearly, we have

$$\nabla_{\boldsymbol{y}_{S_k}}\hat{W}_k\big|_{\boldsymbol{y}=\boldsymbol{y}^*} = \frac{1}{\lambda_k^*}\nabla_{\boldsymbol{y}_{S_k}}W_k\big|_{\boldsymbol{y}=\boldsymbol{y}^*}.$$

From the definition of $\boldsymbol{\lambda}^*$ and $\boldsymbol{y}^*$, we have

$$\nabla_{\boldsymbol{y}_{S_k}}W_k\big|_{\boldsymbol{y}=\boldsymbol{y}^*} - 2\lambda_k^*\boldsymbol{y}_{S_k}^* = \boldsymbol{0},$$

and thus

$$\nabla_{\boldsymbol{y}_{S_k}}\hat{W}_k\big|_{\boldsymbol{y}=\boldsymbol{y}^*} - 2\boldsymbol{y}_{S_k}^* = \boldsymbol{0}.$$

We can then define the following problem

$$\begin{aligned} &\text{maximize} \;\; \tilde{W}_B(\boldsymbol{y}) = \frac{1}{2}\boldsymbol{y}^T\tilde{B}\boldsymbol{y} \\ &\text{subject to} \;\; \sum_{i=1}^{N}(y_i)^2 \leq C, \end{aligned}$$

similar to proposition 1, we know that

$$\nabla_{\boldsymbol{y}_{S_k}}\hat{W}_k = \nabla_{\boldsymbol{y}_{S_k}}\tilde{W}_B(\boldsymbol{y}),$$

and we know from the KKT conditions that since $\|\boldsymbol{y}^*\|_2^2 = C$, we have $\boldsymbol{y} = \boldsymbol{y}^*, \mu^* = 1$ as the primal and dual optimal values to the Lagrangian

$$L_B(\boldsymbol{y},\mu) = \tilde{W}_B(\boldsymbol{y}) - \mu(C - \|\boldsymbol{y}^*\|_2^2),$$

and thus since $B = A^{1/2}DA^{1/2}, A \succ \boldsymbol{0}, D \succ \boldsymbol{0}$, we know $B \succ \boldsymbol{0}$ and then similar to the proof of proposition 1, $\tilde{B} \succ \boldsymbol{0}$. Since $\boldsymbol{y}^*$ is the primal optimal, we know that $\rho(\boldsymbol{y}^*, \boldsymbol{v}_{max}) = 1$, where $\boldsymbol{v}_{max}$ is the eigenvector of $\tilde{B}$'s largest eigenvalue.

When $\boldsymbol{b} \neq \boldsymbol{0}$ but $C_k >> \|\boldsymbol{b}_{S_k}\|_2^2$, it's not hard to show that

$$(\boldsymbol{y}+\boldsymbol{b})^T A(\boldsymbol{y}+\boldsymbol{b}) \to \boldsymbol{y}^T A\boldsymbol{y}, \text{ as } \|\boldsymbol{y}_{S_k}\|/\|\boldsymbol{b}_{S_k}\| \to \infty, \forall k,$$

an thus the above results still hold. □

In [30], the single planner's problem can be thought of multiple cooperative planners with transferable budget, then the shadow prices for every planner is the same in the optimal intervention, and thus $\rho(\boldsymbol{y}^*, \boldsymbol{v}_{max}) = 1$, where $\boldsymbol{v}_{max}$ is the eigenvector of A's largest eigenvalue. Moreover, in that case, when G is an all positive(negative) matrix, $\boldsymbol{v}_{max}$ is the eigenvector of G's largest eigenvalue($-G$'s smallest eigenvalue).

C.5 Proof of Theorem 12

Proof. We will provide the derivations for the numerator and denominator separately. And we begin with the case where $\boldsymbol{b} = \boldsymbol{0}$.

The denominator corresponds to the cooperative planners' intervention game, we can write out the Lagrangian function as follows

$$L(\boldsymbol{y}, \boldsymbol{\lambda}) = W(\boldsymbol{y}) + \sum_{k=1}^{M} \lambda_k (C_k - \|\boldsymbol{y}_{S_k}\|^2).$$

As mentioned earlier in Section 4.4, the strong duality holds, and from the KKT conditions, we have at the optimum that

$$\nabla_{\boldsymbol{y}_{S_k}} W\big|_{\boldsymbol{y}=\overline{\boldsymbol{y}}} - 2\overline{\lambda}_k \overline{\boldsymbol{y}}_{S_k} = \boldsymbol{0} \;\Leftrightarrow\; A_{S_k,S_k}\overline{\boldsymbol{y}}_{S_k} + \sum_{l\neq k} A_{S_k,S_l}\overline{\boldsymbol{y}}_{S_l} = 2\overline{\lambda}_k \overline{\boldsymbol{y}}_{S_k}.$$

So we can rewrite the social welfare as

$$\begin{aligned} W(\overline{\boldsymbol{y}}) &= \frac{1}{2}\overline{\boldsymbol{y}}^T A \overline{\boldsymbol{y}} \\ &= \frac{1}{2}\sum_{k=1}^{M} \overline{\boldsymbol{y}}_{S_k}^T \left(\sum_{l=1}^{M} A_{S_k,S_l}\overline{\boldsymbol{y}}_{S_l} \right) \\ &= \frac{1}{2}\sum_{k=1}^{M} \overline{\boldsymbol{y}}_{S_k}^T (2\overline{\lambda}_k \overline{\boldsymbol{y}}_{S_k}) \\ &= \sum_{k=1}^{M} \overline{\lambda}_k \|\overline{\boldsymbol{y}}\|_2^2 \\ &= \sum_{k=1}^{M} \overline{\lambda}_k C_k. \end{aligned}$$

For the numerator that corresponds to the non-cooperative planners' intervention game, we can write out the Lagrangian function for p_k as follows

$$L_k(\boldsymbol{y}_{S_k}, \boldsymbol{y}_{-S_k}, \lambda_k) = W_k(\boldsymbol{y}_{S_k}, \boldsymbol{y}_{-S_k}) + \lambda_k(C_k - \|\boldsymbol{y}_{S_k}\|^2).$$

From the KKT conditions, we have at the equilibrium that

$$\begin{aligned} &\nabla_{\boldsymbol{y}_{S_k}} W_k\big|_{\boldsymbol{y}=\boldsymbol{y}^*} - 2\lambda_k^* \boldsymbol{y}_{S_k}^* = \mathbf{0} \\ \Leftrightarrow\ &A_{S_k,S_k}\boldsymbol{y}_{S_k}^* + \frac{1}{2}\sum_{l\neq k} A_{S_k,S_l}\boldsymbol{y}_{S_l}^* = 2\lambda_k^*\boldsymbol{y}_{S_k}^*. \end{aligned}$$

So we can rewrite the social welfare as

$$\begin{aligned} W(\boldsymbol{y}^*) &= \frac{1}{2}(\boldsymbol{y}^*)^T A \boldsymbol{y}^* \\ &= \frac{1}{2}\sum_{k=1}^{M}(\boldsymbol{y}_{S_k}^*)^T\left(\sum_{l=1}^{M} A_{S_k,S_l}\boldsymbol{y}_{S_l}^*\right) \\ &= \frac{1}{2}\sum_{k=1}^{M}(\boldsymbol{y}_{S_k}^*)^T(4\lambda_k^*\boldsymbol{y}_{S_k}^* - A_{S_k,S_k}\boldsymbol{y}_{S_k}^*) \\ &\geq \sum_{k=1}^{M}(2\lambda_k^* - \frac{1}{2}\rho_k)\|\boldsymbol{y}_{S_k}^*\|_2^2 \\ &= \sum_{k=1}^{M}(2\lambda_k^* - \frac{1}{2}\rho_k)C_k, \end{aligned}$$

and since $e_{L2} = \frac{W(\boldsymbol{y}^*)}{W(\overline{\boldsymbol{y}})}$, we know the lower bound holds.

When $\boldsymbol{b} \neq \mathbf{0}$ but $C_k >> \|\boldsymbol{b}_{S_k}\|_2^2$, it's not hard to show that

$$(\boldsymbol{y}+\boldsymbol{b})^T A(\boldsymbol{y}+\boldsymbol{b}) \to \boldsymbol{y}^T A\boldsymbol{y}, \text{ as } \|\boldsymbol{y}_{S_k}\|/\|\boldsymbol{b}_{S_k}\| \to \infty, \forall k,$$

an thus the above results still hold. □

C.6 Proof of Proposition 2

Proof. Suppose now group k transfers an infinitesimal amount of budget to group l, then after the transfer, the new intervention profile of group k and l becomes

$$\boldsymbol{y}'_{S_k} = (1-\delta_k)\boldsymbol{y}_k,\ \ \boldsymbol{y}'_{S_l} = (1+\delta_l)\boldsymbol{y}_l,\ \ \frac{\delta_l}{\delta_k} = \frac{\|\boldsymbol{y}_k\|^2}{\|\boldsymbol{y}_l\|^2} = \frac{C_k}{C_l}.$$

Then we look at group k's welfare after the transfer, if it increases, group k has an incentive to do the transfer

$$\begin{aligned}
W_k(\boldsymbol{y}'_{S_k}, \boldsymbol{y}'_{S_l}, \boldsymbol{y}_{-(S_k,S_l)}) - W_k(\boldsymbol{y}) &= \frac{1}{2}\big([(1+\delta_l)(1-\delta_k)-1]\boldsymbol{y}_{S_k}^T A_{S_k,S_l}\boldsymbol{y}_{S_l} \\
&\quad + [(1-\delta_k)^2-1]\boldsymbol{y}_{S_k}^T A_{S_k,S_k}\boldsymbol{y}_{S_k} \\
&\quad + 2[(1-\delta_k)-1]\boldsymbol{y}_{S_k}^T (A_{S_k,:})\boldsymbol{b}\big) \geq 0 \\
&\Leftrightarrow (\delta_l - \delta_k)(\nabla_{\boldsymbol{y}_{S_l}} W_k)^T \boldsymbol{y}_{S_l} \geq \delta_k (\nabla_{\boldsymbol{y}_{S_k}} W_k)^T \boldsymbol{y}_{S_k} \\
&\Leftrightarrow (C_k - C_l)(\nabla_{\boldsymbol{y}_{S_l}} W_k)^T \boldsymbol{y}_{S_l} \geq C_l (\nabla_{\boldsymbol{y}_{S_k}} W_k)^T \boldsymbol{y}_{S_k}.
\end{aligned}$$

since δ_l and δ_k are all infinitesimal, we can ignore their second order products in the above derivations. □

APPENDIX D

Chapter 5 Appendix

D.1 Supplementary Material for Section 5.2

D.1.1 Discussion on Remark 2

In this part, we discuss the case where game parameters like $\boldsymbol{\theta}$ and P are unknown and the decision maker need to be learn them. Unlike the single round, two stage game in the main article, the learning process requires online learning with multiple rounds, each containing two stages.

We note that the quality coefficients $\boldsymbol{\theta}$ can be learned in one round by setting $f = 0$ and then we have $(\boldsymbol{z}, y') = (\boldsymbol{x}, y)$, and running any suitable learning algorithm can get an estimate of $\boldsymbol{\theta}$.

However, P can not always be learned in the conventional learning problem. We can use an example can from the impossibility conditions in Theorem 13, given those conditions, only the columns whose index has substitutability 1 can be learned, the other columns are always unknown. Below we show how the discount mechanism help with learning the the projection matrix P.

In the regression problem with L1 cost, we can use the following procedures to learn the projection matrices,

- Choose f such that $\boldsymbol{w} > 0$ (without loss of generality, assume that $\boldsymbol{w} > 0 \Rightarrow P^T\boldsymbol{w} > 0$, otherwise some action dimensions are meaningless)
- For each time step $t = 1, \ldots, M$, get a sufficiently large sample of agents with their observable features $\boldsymbol{z}$
- At $t = 0$, $G_d = 0$, let $\overline{\boldsymbol{z}}_0 = \mathbb{E}[\boldsymbol{z}]$
- At $t = 1, \ldots, M$, let G_d induce the best response along action dimension t by lowering the cost to $\tilde{c}_t$, and let $\overline{\boldsymbol{z}}_t = \mathbb{E}[\boldsymbol{z}]$
- Compute $\boldsymbol{\nu}_t = (\overline{\boldsymbol{z}}_t - \overline{\boldsymbol{z}}_0)\tilde{c}_t/B$, which is an estimate of $P\boldsymbol{e}_t = \boldsymbol{p}_t$, i.e., the t-th column of P.

Discount mechanisms can enable best responses to in action dimensions that are impossible to be incentivized with the decision rule itself, and this is true for both classification and regression, both L1 cost and other types of costs like L2 or squared.

D.1.2 Discussion on Remark 3

We will use the L2 cost $h(\boldsymbol{a}) = \|\boldsymbol{a}\|_2$ for demonstration purpose, and we note that higher orders of cost functions $h(\boldsymbol{a}) = \frac{1}{2}\|\boldsymbol{a}\|_2^2$ are very similar in classification but different in regression. In regression, higher order costs are convex and the marginal cost grows, and thus there is no need to be a budget constraint $B \geq h(\boldsymbol{a})$, other than that, $h(\boldsymbol{a}) = \|\boldsymbol{a}\|_2$ is very representative.

For all other cost functions, we can equivalently have a set of "equal cost contour" i.e., $\{\boldsymbol{a}|h(\boldsymbol{a}) = C\}$ for some constant C is a contour. Most cost functions used in economic and computer science literature have contours with different sizes but a constant "shape" (the surface of norm balls, since the cost functions are norm based), like the L1 cost, L2 cost, tilted L2 cost $h(\boldsymbol{a}) = \sqrt{\boldsymbol{a}^T C \boldsymbol{a}}$ and squared cost $h(\boldsymbol{a}) = \frac{1}{2}\|\boldsymbol{a}\|_2^2$. The constant shape of contours made it possible to have a concise (closed-form in most cases) representation of the best responses' directional and magnitude properties.

For example, when $h(\boldsymbol{a}) = \|\boldsymbol{a}\|_2$, the best responses satisfy $\rho(\boldsymbol{a}_t^*, P^T\boldsymbol{w}) = 1$ where $\rho(\boldsymbol{v}_1, \boldsymbol{v}_2) = \frac{\boldsymbol{v}_1^T\boldsymbol{v}_2}{\|\boldsymbol{v}_1\|_2\|\boldsymbol{v}_2\|_2}$ is the cosine similarity. We still have properties in Lemma 2 and in classification and regression, the best responses are

$$\boldsymbol{a}_C^*(\boldsymbol{x}) = \frac{\tau - \boldsymbol{w}^T\boldsymbol{x}}{\|P^T\boldsymbol{w}\|_2^2} P^T\boldsymbol{w}, \quad \boldsymbol{a}_C^*(\boldsymbol{x}) = \frac{B}{\|P^T\boldsymbol{w}\|_2} P^T\boldsymbol{w},$$

and we can similarly write out the expressions of the AS best responses for other cost functions.

For L2 cost $h(\boldsymbol{a}) = \|\boldsymbol{a}\|_2$, we can think of discounts with minimum effective discount value as giving certain action directions a fixed discount rate or incentivizing agents to play a different action and pay the cost differences.

Therefore, the implementer will try to incentivize some of the agents to take an AS best response that also reach the boundary, this can be done by making the discount amount equal the cost difference between the AS and the CS/LS best response. The implementer wants to maximize the subsidy surplus on a given agent, which is the quality gain $l(\boldsymbol{a}) - l(\boldsymbol{a}_C^*(\boldsymbol{x}))$ minus the subsidy cost $\|\boldsymbol{a}\|_2 - \|\boldsymbol{a}_C^*(\boldsymbol{x})\|_2$ and thus $\boldsymbol{a}_A^*(\boldsymbol{x})$ is the solution to the optimization problem

$$\begin{aligned} \text{minimize}_{\boldsymbol{a}} \;\; & \|\boldsymbol{a}\|_2 - l(\boldsymbol{\theta}^T(\boldsymbol{x} + \hat{P}\boldsymbol{a})) \\ \text{subject to} \;\; & \boldsymbol{w}^T P\boldsymbol{a} = \tau - \boldsymbol{w}^T\boldsymbol{x} \end{aligned} \tag{D.1}$$

However, the above problem is in general not convex and can be NP hard to find the optimal solution. But the below assumption guarantees a solution.

Assumption 6. $\boldsymbol{w} = \boldsymbol{\theta}$*, and the implementer limit the AS best response to be* gaming free*, i.e.,* $[\boldsymbol{a}_A^*(\boldsymbol{x})]_j = \mathbf{1}\{j \leq M_i\} \Leftrightarrow P\boldsymbol{a}_A^*(\boldsymbol{x}) = \hat{P}\boldsymbol{a}_A^*(\boldsymbol{x})$,

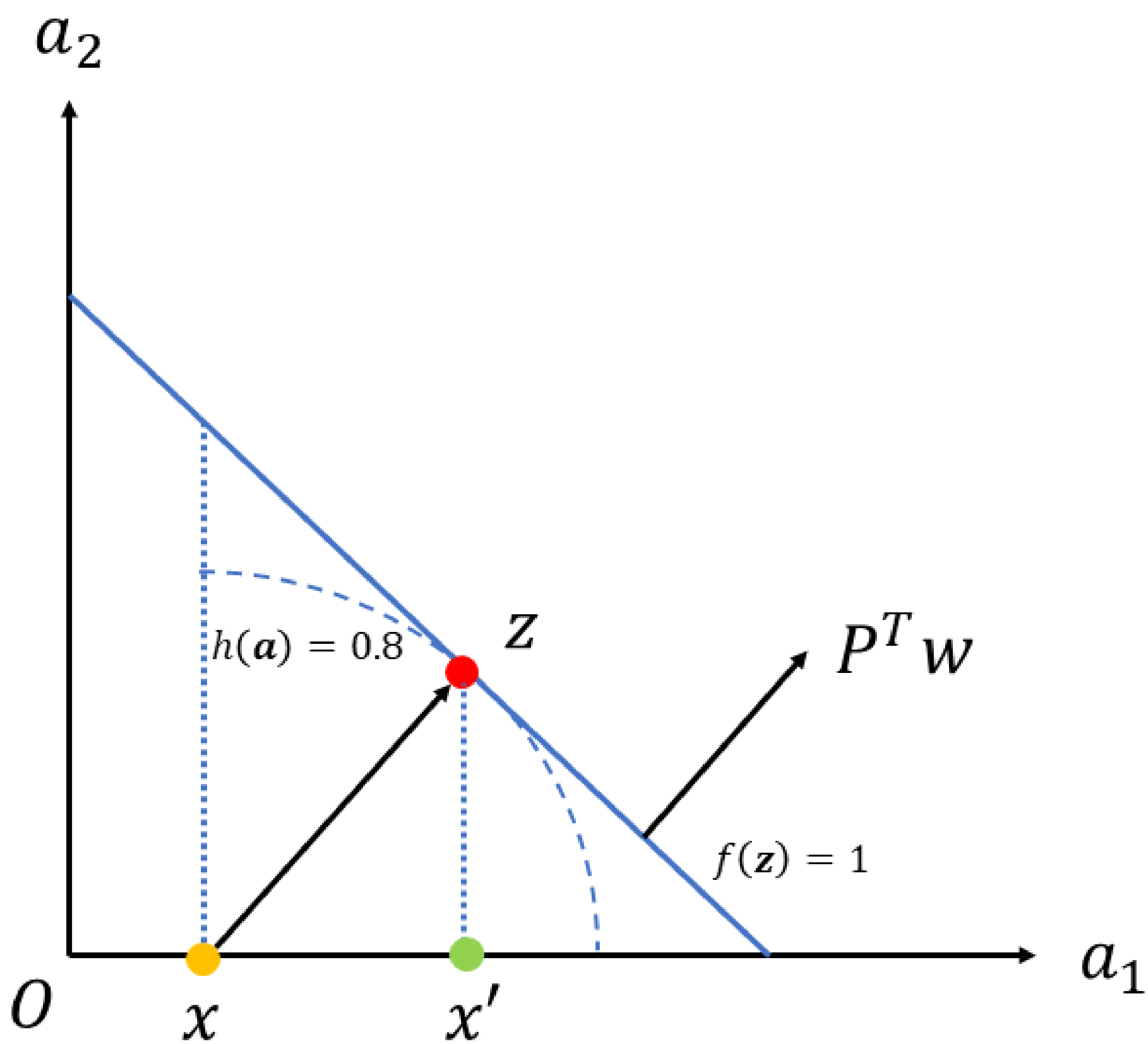


Figure D.1: An illustration of a CS best response in classification with L2 cost, where the blue dashed curve (quarter circle) is an equal cost contour, $P = [1, 1]$, $\boldsymbol{w} = (1, 1)$, a_1 is improvement and a_2 is gaming.

Under Assumption 6, the problem becomes convex since $l(\boldsymbol{\theta}^T(\boldsymbol{x} + \hat{P}\boldsymbol{a})) = l(\tau)$ is constant

$$\begin{aligned} \text{minimize}_{\boldsymbol{a}} \quad & \|\boldsymbol{a}\|_2 \\ \text{subject to} \quad & \boldsymbol{\theta}^T\hat{P}\boldsymbol{a} = \tau - \boldsymbol{\theta}^T\boldsymbol{x} \end{aligned} \tag{D.2}$$

and the solution (AS best response to incentivize) is

$$\boldsymbol{a}_A^*(\boldsymbol{x}) = (\tau - \boldsymbol{\theta}^T\boldsymbol{x})\frac{\hat{P}^T\boldsymbol{\theta}}{\|\hat{P}^T\boldsymbol{\theta}\|_2^2} \tag{D.3}$$

We can then similarly define the individual subsidy surplus in the L2 case and find sufficient conditions that guarantees an IC, IR and BB mechanism $G \neq 0$ or even find the optimal solutions with the same assumptions made in the Theorems 15 and 16.

One interesting difference in the L2 cost case is that the decision rule can incentivize *partial improvement*, which can also be called *partial gaming*, which means $\boldsymbol{\theta}^T P\boldsymbol{a} > \boldsymbol{\theta}^T\hat{P}\boldsymbol{a} > 0$, and the corresponding theorems in L1 case still applies when f incentivizes pure gaming $\boldsymbol{\theta}^T P\boldsymbol{a} > \boldsymbol{\theta}^T\hat{P}\boldsymbol{a} = 0$. An example of pure gaming happens when for every improvement action j, there is a corresponding gaming action k with the an exaggerated effect $\boldsymbol{p}_k = \alpha_j\boldsymbol{p}_j, \alpha_j > 1$, which can model problems like multi-subject exams where an agent has an improvement and gaming action for each of the subject and cheating is always more cost efficient than working hard without an incentivize mechanism.

D.1.3 An Alternative Incentive Mechanism

An alternative mechanism to consider, the *transfer mechanisms* is based on *monetary transfer*, where the mechanism designer provides reimbursement or bonus payment when the agent meets certain feature criteria, e.g., rewards for high scores. We use G_t to denote the transfer mechanism, where the designer chooses a bonus amount $b(\boldsymbol{z}), b : \mathbb{R}^N \mapsto \mathbb{R}$, effectively revising the agent's utility to

$$u_A(\boldsymbol{x}, \boldsymbol{a}) = f(\boldsymbol{x} + P\boldsymbol{a}) - h(\boldsymbol{a}) + b(\boldsymbol{x} + P\boldsymbol{a}). \tag{D.4}$$

In transfer mechanisms, knowing the actual $\boldsymbol{x}$ seems to help the designer reduce the subsidy cost on agents with high endowment and low improvement, but we will show below that this extended version with bonus amount $\tilde{b}(\tilde{\boldsymbol{x}}, \boldsymbol{z})$ is equivalent as the bonus $b(\boldsymbol{z})$ that only uses features as input, where $\tilde{x}$ is the reported pre-response attribute. This is because $\tilde{b}(\tilde{\boldsymbol{x}}, \boldsymbol{z})$ either can not incentivize agents to truthfully report $\tilde{\boldsymbol{x}} = \boldsymbol{x}$, or it can not generate more benefit for the mechanism designer.

With the alternative version of the monetary transfer mechanism, the agent's utility now becomes

$$\tilde{u}_A(\boldsymbol{x}, \boldsymbol{a}, G_t) = f(\boldsymbol{x} + P\boldsymbol{a}) - h(\boldsymbol{a}) + \max_{\tilde{\boldsymbol{x}}} \tilde{b}(\tilde{\boldsymbol{x}}, \boldsymbol{x} + P\boldsymbol{a}),$$

and we can find the corresponding $\boldsymbol{a}_A^*(\boldsymbol{x})$, and only if

$$\boldsymbol{x} \in \arg\max_{\tilde{\boldsymbol{x}}} \tilde{b}(\tilde{\boldsymbol{x}}, \boldsymbol{x} + P\boldsymbol{a}_A^*(\boldsymbol{x})),$$

truth-reporting is incentivized. If truth reporting is not incentivized, $\tilde{b}(\tilde{\boldsymbol{x}}, \boldsymbol{z})$ and $b(\boldsymbol{z}) = \max_{\tilde{\boldsymbol{x}}} \tilde{b}(\tilde{\boldsymbol{x}}, \boldsymbol{z})$ are equivalent for both the agents and the mechanism designer. Meanwhile, for $\forall \boldsymbol{x}_1 \neq \boldsymbol{x}_2$, truth telling requires either

$$\boldsymbol{x}_1 + P\boldsymbol{a}_A^*(\boldsymbol{x}_1) \neq \boldsymbol{x}_2 + P\boldsymbol{a}_A^*(\boldsymbol{x}_2),$$

indicating that backward induction from $\boldsymbol{x} + P\boldsymbol{a}_A^*(\boldsymbol{x})$ to $\boldsymbol{x}$ is achievable, or

$$\boldsymbol{x}_1 + P\boldsymbol{a}_A^*(\boldsymbol{x}_1) = \boldsymbol{x}_2 + P\boldsymbol{a}_A^*(\boldsymbol{x}_2), \text{ and } \boldsymbol{x}_1, \boldsymbol{x}_2 \in \arg\max_{\tilde{\boldsymbol{x}}} \tilde{b}(\tilde{\boldsymbol{x}}, \boldsymbol{x} + P\boldsymbol{a}_A^*(\boldsymbol{x})).$$

In either case, $b(\boldsymbol{z})$ is sufficient.

However, the computational complexity is very high in the backward induction step for a general $b(\boldsymbol{z})$ bonus function. Recall that the AS utility of an agent is

$$u_A(\boldsymbol{x}, \boldsymbol{a}) = f(\boldsymbol{x} + P\boldsymbol{a}) - h(\boldsymbol{a}) + b(\boldsymbol{x} + P\boldsymbol{a}),$$

and thus computing $\boldsymbol{a}_A^*(\boldsymbol{x}) = \arg\max_{\boldsymbol{a}} u_A(\boldsymbol{x}, \boldsymbol{a})$ is non-convex for a non-concave $b(r)$ bonus function.

On one hand, we can't guarantee concave $b(r)$ is the optimal solution. On the other hand, for a concave $b(\boldsymbol{z})$, the computation of $\boldsymbol{a}_A^*(\boldsymbol{x}) = \arg\max_{\boldsymbol{a}} u_A(\boldsymbol{x}, \boldsymbol{a})$ is convex and but the individual subsidy surplus

$$s(\boldsymbol{x}, f, G_t) = l(\boldsymbol{\theta}^T(\boldsymbol{x} + \hat{P}\boldsymbol{a}_A^*(\boldsymbol{x}))) - l(\boldsymbol{\theta}^T(\boldsymbol{x} + \hat{P}\boldsymbol{a}_C^*(\boldsymbol{x}))) - b(\boldsymbol{x} + P\boldsymbol{a}_A^*(\boldsymbol{x}))$$

on the agents are not concave unless l is convex (we are supposing $\boldsymbol{x} \in \mathcal{M}(f)$ here, otherwise more non-convexity is introduced). Moreover, the overall objective depends on the integration on a subset of $\hat{\mathcal{X}} \subseteq \mathcal{X}$

$$S(f, G_t) = \int_{\hat{\mathcal{X}}} s(\boldsymbol{x}, f, G_t) p(\boldsymbol{x}) d\boldsymbol{x},$$

and a general probability density function p, and the convexity of set $\hat{\mathcal{X}}$ can make the mechanism designer's objective non-convex even if l is convex.

We also note that when changing the value $b(\boldsymbol{z})$ for a certain $\boldsymbol{z}$, the AS best response for all agents with pre-response attribute $\boldsymbol{x}$ in the cone $\boldsymbol{x} - \boldsymbol{z} \leq 0$ (element wise non-positive) might change, and this also makes the analysis hard.

D.2 Supplementary Material for Section 5.3

D.2.1 Characterization of the optimal LS decision rule

Lemma 6. *The LS optimal decision rule is* $f_L^*(\boldsymbol{z}) = \mathbf{1}\{\boldsymbol{\theta}^T \boldsymbol{z} \geq \tau_L\},\ \tau_L = \arg\min_\tau l(\tau) \geq 0.5.$

Proof. This is because it is optimal for the decision maker to accept every agent with $l(\boldsymbol{\theta}^T \boldsymbol{x}) \geq 0.5$, since rejecting this agent results in a decrease in the expected individual prediction outcome $1 - l(\boldsymbol{\theta}^T \boldsymbol{x}) \leq l(\boldsymbol{\theta}^T \boldsymbol{x})$. Similarly, the decision maker wants to reject every agent with $l(\boldsymbol{\theta}^T \boldsymbol{x}) < 0.5$. □

D.2.2 Proof of Theorem 13

Proof. The proofs of the claims

1. If $\kappa_j = 1$, then there exists a $\boldsymbol{w}$ in f that can incentivize action dimension j, and the $\boldsymbol{w}$ can be found in polynomial time;

2. if $\kappa_j < 1$, meaning there always are linear combinations of gaming actions weakly dominate every action j, then there is no f that can incentivize best response on action j.

are covered in [54, 50]. Intuitively, if $\kappa_j < 1, \forall j \leq M_+$, the corresponding $\boldsymbol{a}$ is the combination that strictly dominates $\boldsymbol{e}_j$ for any f and thus there is no f that can incentivize improvement.

We will proceed to show the decision maker's CS optimal strategy satisfy $\boldsymbol{w} = \boldsymbol{\theta}$. The main idea is that when f always incentivizes gaming, then the CS decision outcomes with $f_C(\boldsymbol{z}) = \mathbf{1}\{\boldsymbol{w}_C^T \boldsymbol{z} \geq \tau_C\}$ always have an equivalent LS decision outcomes with $f_L(\boldsymbol{z}) = \mathbf{1}\{\boldsymbol{w}_L^T \boldsymbol{z} \geq \tau_L\}$, where the $\boldsymbol{w}_C = \boldsymbol{w}_L$, and τ_C, τ_L satisfy

$$\tau_L = \min\left\{0, \tau_C - \frac{(P^T \boldsymbol{w})_k}{c_k}\right\}.$$

In other words, we can show that $\forall \boldsymbol{x}, f_L(\boldsymbol{x}) = f_C(\boldsymbol{x} + P\boldsymbol{a}^*)$, and thus is equivalent for the decision maker to find an optimal f_L which guarantees $\boldsymbol{w}_L = \boldsymbol{\theta}$ as the Lemma 6 suggests. □

D.2.3 Proof of Theorem 14

Proof. We will first show the problem is non-convex when discount is placed on multiple actions, then show even the discount is only on one action, the problem is still non-convex.

When the discount is on multiple actions, providing the optimal tie breaking strategy for an agent with $\boldsymbol{x}$ requires solving

$$\text{maximize}_{\boldsymbol{a}} \; l(\boldsymbol{\theta}^T(\boldsymbol{x}+\hat{P}\boldsymbol{a})) - \triangle\boldsymbol{c}^T\boldsymbol{a},$$

which is non-convex for a general l function. This is for individual subsidy surplus for a fixed $\triangle\boldsymbol{c}$, and it has to be integrated over $\mathcal{X}$ to compute the overall subsidy surplus $S(f, G)$. So finding the optimal mechanism will only have higher computational complexity when the decision maker has to optimize over $\triangle\boldsymbol{c}, \underline{c}, \overline{c}$, and take into account the influence of $p(\boldsymbol{x})$.

When the discount is only on one action, from Lemma 4, the mechanism designer need to choose $\triangle\boldsymbol{c}$ such that

$$\triangle\boldsymbol{c}_j \geq \triangle c_j^* = c_j - \frac{(P^T\boldsymbol{w})_j}{(P^T\boldsymbol{w})_{i_C}} c_{i_C},$$

for some improvement action dimension $j \leq M_+$ that it wants to incentivize the agents.

Then for the decision maker, maximizing its AS utility is equivalent as maximizing the subsidy surplus, so the decision maker solves

$$\begin{aligned} \text{maximize}_{j,\triangle\boldsymbol{c}_j,\underline{c},\overline{c}} \quad & \int_{\mathcal{X}} [Pr(f(\boldsymbol{x}+P\boldsymbol{a}_A^*(\boldsymbol{x})) = y_A') - \mathbf{1}\{\triangle\boldsymbol{c}^T\boldsymbol{a}_A^*(\boldsymbol{x}) \in [\underline{c}, \overline{c}]\}] p(\boldsymbol{x}) d\boldsymbol{x} \\ \text{subject to} \quad & \triangle\boldsymbol{c} \in [\triangle\boldsymbol{c}^*, c_j), j \leq M_+ \end{aligned}$$

where the problem can be non-convex and not monotone for general p and l. Specifically, when j has the highest return of investment after the discount, the backward induction that anticipates the agent's AS best response is,

$$\boldsymbol{a}_A^*(\boldsymbol{x}) = \begin{cases} \frac{\tau - \boldsymbol{w}^T\boldsymbol{x}}{(P^T\boldsymbol{w})_j}\boldsymbol{e}_j, \text{ if } \frac{\triangle c_j(\tau-\boldsymbol{w}^T\boldsymbol{x})}{(P^T\boldsymbol{w})_j} \in [\underline{c}, \overline{c}], \\ \frac{\tau-\boldsymbol{w}^T\boldsymbol{x}}{(P^T\boldsymbol{w})_{i_C}}\boldsymbol{e}_{i_C}, \text{ o.w.} \end{cases}$$

This indicates that agents with $\boldsymbol{x}$ in a belt shape subset of $\mathcal{X}$ will be incentivized to improve, but the overall subsidy surplus is in general not convex, not concave and not monotone in either the upper bound (determined by f and $\overline{c}$) or the lower bound (determined by f and $\underline{c}$) of the belt even when the other is fixed. Moreover, the minimum effective discount value $\triangle c_j^*$ is not always the optimal solution, adding more complexity to the problem. This is because sometimes the decision maker wants to put more discount on the action dimension and incentivize some agents outside of the manipulation margin to improve and accept them rather than reject them. For example, if 80 percent agent has attribute that makes their likelihood 0.49, the minimum effective discount value still makes them rejected and take 0 AS best response, but a slightly higher discount can incentivize them all to improve to the threshold value, say 0.7, the $0.7 - (1 - 0.49) \cdot 0.8 = 0.152$ amount of improvement may largely outweigh the extra subsidy cost.

Overall speaking, the difference between $\boldsymbol{w}$ and $\boldsymbol{\theta}$ in f, the global properties of p, l and their local properties influenced by τ all makes the problem hard to solve. □

D.2.4 Proof of Theorem 15

Proof. We will show that any $G \neq 0$ returned by Algorithm 6 is IC, IR and satisfies $S(f, G) \geq 0$.

The IC part follows that the participants act in self-interest. Also, as previously discussed, the minimum effective discounted value $\triangle c_j = \triangle c_j^* = c_j - \frac{(P^T\boldsymbol{w})_j}{(P^T\boldsymbol{w})_{i_C}} c_{i_C}$ makes sure the agents are weakly better off in the AS game than the CS game (given the same f).

We note that for all f that incentivizes gaming, the decision maker would prefer $\boldsymbol{w} = \boldsymbol{\theta}$ and we can use Theorem 16 to find G, so below we have $i_C \leq M_+$.

The basic logic of ensuring $S(f, G) \geq 0$ is that the algorithm finds a specific agent that is incentivized, and if this specific agent has a non-negative individual subsidy surplus, it is sufficient that all the other incentivized agents also have non-negative individual subsidy surplus and thus $S(f, G) \geq 0$.

In Algorithm 6, the designer finds (a convex problem and easy to solve)

$$\underline{\boldsymbol{x}} = \arg \min_{\boldsymbol{x}:\boldsymbol{w}^T\boldsymbol{x} = \tau - \delta_j (P^T\boldsymbol{w})_j} \boldsymbol{\theta}^T\boldsymbol{x},$$

which is the attribute of the specific agent. From the upper bound set on δ_j in the algorithm, we assume the specific agent is in $\mathcal{M}(f)$, and then uses

$$\underline{s} = l_+ - \delta_j \triangle c_j = l(\boldsymbol{\theta}^T(\underline{\boldsymbol{x}} + \delta_j \hat{P}\boldsymbol{e}_j)) - l(\boldsymbol{\theta}^T(\underline{\boldsymbol{x}} + \delta_{i_C}\hat{P}\boldsymbol{e}_{i_C})) - \delta_j \triangle c_j,$$

as a benchmark, where δ_j is the δ in the algorithm and $\delta_{i_C} = \frac{(P^T\boldsymbol{w})_j}{(P^T\boldsymbol{w})_{i_C}}\delta_j$. $\delta_j \boldsymbol{e}_j$ and $\delta_{i_C}\boldsymbol{e}_{i_c}$ help the agent achieve the same $\boldsymbol{w}^T\boldsymbol{z}$, $\underline{c} = 0, \overline{c} = \delta_j \triangle c_j$ here.

Then $\underline{s}$ is the specific agent's individual subsidy surplus, i.e.,

$$s(\underline{\boldsymbol{x}}, f, G) = l(\boldsymbol{\theta}^T(\underline{\boldsymbol{x}} + \hat{P}\boldsymbol{a}_A^*(\underline{\boldsymbol{x}}))) - l(\boldsymbol{\theta}^T(\underline{\boldsymbol{x}} + \hat{P}\boldsymbol{a}_C^*(\underline{\boldsymbol{x}}))) - \mathbf{1}\{\triangle \boldsymbol{c}^T\boldsymbol{a}_A^*(\boldsymbol{x}) \in [\underline{c}, \overline{c}]\} = \underline{s}.$$

We start with agents with CS best response $\boldsymbol{a}_C^*(\boldsymbol{x}) = \delta_{i_C}\boldsymbol{e}_{i_C}$, i.e., $\boldsymbol{w}^T\boldsymbol{x} = \boldsymbol{w}^T\underline{\boldsymbol{x}}$. For them, the AS best response is $\boldsymbol{a}_A^*(\boldsymbol{x}) = \delta_j\boldsymbol{e}_j$, the individual subsidy surplus is then

$$s(\boldsymbol{x}, f, G) = l(\boldsymbol{\theta}^T(\boldsymbol{x} + \delta_j\hat{P}\boldsymbol{e}_j)) - l(\boldsymbol{\theta}^T(\boldsymbol{x} + \delta_{i_C}\hat{P}\boldsymbol{e}_{i_C})) - \delta_j \triangle c_j,$$

since (1) $\boldsymbol{\theta}^T\hat{P}(\delta_j\boldsymbol{e}_j - \delta_{i_C}\boldsymbol{e}_{i_C})$ is constant, (2) $\boldsymbol{\theta}^T\boldsymbol{x} \geq \boldsymbol{\theta}^T\underline{\boldsymbol{x}}$ and (3) l is convex on this range, we have $s(\boldsymbol{x}, f, G) \geq \underline{s} \geq 0$.

For agents with "higher endowment" $\boldsymbol{w}^T\boldsymbol{x} > \boldsymbol{w}^T\underline{\boldsymbol{x}}$, i.e., with CS best response $\boldsymbol{a}_C^*(\boldsymbol{x}) = \alpha_{i_C}\boldsymbol{e}_{i_C}$, $\alpha_{i_C} < \delta_{i_C}$, we denote $\alpha_j = \alpha_{i_C}(P^T\boldsymbol{w})_{i_C}/(P^T\boldsymbol{w})_j$, then the (sub-optimal) AS best response is $\boldsymbol{a}_A^*(\boldsymbol{x}) = \alpha_j\boldsymbol{e}_j$, and the individual subsidy surplus is

$$\begin{aligned} s(\boldsymbol{x}, f, G) &= l(\boldsymbol{\theta}^T(\boldsymbol{x} + \alpha_j\hat{P}\boldsymbol{e}_j)) - l(\boldsymbol{\theta}^T(\boldsymbol{x} + \alpha_{i_C}\hat{P}\boldsymbol{e}_{i_C})) - \alpha_{i_C}\overline{c}/\delta_{i_C} \\ &\geq \frac{\alpha_{i_C}}{\delta_{i_C}}[l(\boldsymbol{\theta}^T(\boldsymbol{x} + \delta_j\hat{P}\boldsymbol{e}_j)) - l(\boldsymbol{\theta}^T(\boldsymbol{x} + \delta_{i_C}\hat{P}\boldsymbol{e}_{i_C})) - \overline{c}] \\ &\geq \frac{\alpha_{i_C}}{\delta_{i_C}}\underline{s} \geq 0, \end{aligned}$$

where the second inequality comes from the convexity of l.

For agents with "lower endowment" i.e., with CS best response $\boldsymbol{a}_C^*(\boldsymbol{x}) = \beta_{i_C}\boldsymbol{e}_{i_C}$, $\beta_{i_C} < \delta i_C$, the mechanism designer suggest that they break tie choosing $\boldsymbol{a}_A^*(\boldsymbol{x}) = \beta_{i_C}\boldsymbol{e}_{i_C}$ as the AS best response and thus the individual subsidy surplus is 0. For $\beta_j = \beta_{i_C}(P^T\boldsymbol{w})_{i_C}/(P^T\boldsymbol{w})_j$, we note that $\boldsymbol{a}_A^*(\boldsymbol{x}) = \beta_j\boldsymbol{e}_j$ is a dominated strategy since $\triangle\boldsymbol{c}^T\boldsymbol{a}_A^*(\boldsymbol{x}) > \overline{c}$.

□

Algorithm 10: Extended Grid Search an IC, IR and BB Discount Mechanism for Classification

Choose $\epsilon > 0$, set $\overline{c}_{max} \leftarrow 0$, $ans \leftarrow (\mathbf{0}, 0)$;
Define $a(r, j) = (\tau - r)\boldsymbol{e}_j/(\hat{P}^T\boldsymbol{w})_j$;
Define $r(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x}$;
Define $s(\boldsymbol{x}, j, \triangle\boldsymbol{c}) = l(\boldsymbol{\theta}^T(\boldsymbol{x} + \hat{P}a(r, j))) - l(\boldsymbol{\theta}^T(\boldsymbol{x} + \hat{P}a(r, i_C))) - (\tau - r)\triangle c_j/(\hat{P}^T\boldsymbol{w})_j$;
for $j = 1 : M_+$ **do**
  $\triangle\boldsymbol{c} \leftarrow \mathbf{0}$; $\overline{c} \leftarrow 0$;
  $S \leftarrow 0$;
  $\triangle c_j \leftarrow c_j - \frac{(P^T\boldsymbol{w})_j}{(P^T\boldsymbol{w})_{i_C}}c_{i_C}$;
  while $S \geq 0$ *and* $\delta_j \leq \underline{\boldsymbol{x}}_j$ **do**
    $\overline{c} \leftarrow \overline{c} + \epsilon$;
    $\underline{r} \leftarrow \tau - \delta_j(\hat{P}^T\boldsymbol{w})_j$;
    $S \leftarrow \int_{\{\boldsymbol{x}:r(\boldsymbol{x})\in[\underline{r},\tau]\}} s(\boldsymbol{x}, j, \triangle\boldsymbol{c})p(\boldsymbol{x})d\boldsymbol{x}$;
    if $S > S_{max}$ **then**
      $S_{max} \leftarrow S$; $ans \leftarrow (\triangle\boldsymbol{c}, \overline{c})$;
Return ans.

We see from the proof that when l is convex, agents with "high endowment" will have "high return on investment" for the mechanism designer when utilizing the discount. On the other hand, if l is concave on $[0, \max_{\boldsymbol{x}:\boldsymbol{w}^T\boldsymbol{x}=\tau} l(\boldsymbol{x})]$, we can infer that the agents with "low endowment" will have "high return on investment" when utilizing the discount. So then finding a suitable $\underline{c}$ becomes important, finding the approximate optimal mechanism can follow similar steps in Algorithm 10.

In general, real world data like FICO shows that the likelihood function has an S-shape and is concave on higher score range, and choosing a threshold too high hurts the decision maker.

We also note that the minimum effective discount value is used because it is also "sufficient". For convex l, if an agent cannot guarantee a non-negative individual subsidy surplus under the minimum effective discount value, it can not have a non-negative individual subsidy surplus for any other effective discount value. Not only because the individual subsidy cost goes up, but also because the marginal quality improvement is lower for agents farther away from the boundary while the marginal cost is constant.

D.2.5 Proof of Theorem 16

Proof. When $\boldsymbol{w} = \boldsymbol{\theta}$, the mechanism designer is indifferent about AS best responses along any improvement action dimension.

The mechanism designer find the "cheapest to incentivize" target action dimension

$$i_A = \arg\max_{j \leq M_+} (P^T\boldsymbol{\theta})_j / c_j \;\Leftrightarrow\; i_A = \arg\min \frac{\triangle c_j^*(\tau - \boldsymbol{\theta}^T\boldsymbol{x})}{(P^T\boldsymbol{\theta})_j}$$

and set $\triangle \boldsymbol{c}$ so that $\triangle c_{i_A} \geq \triangle c_{i_A}^* = c_{i_A} - \frac{(P^T\boldsymbol{\theta})_{i_A}}{(P^T\boldsymbol{\theta})_{i_C}} c_{i_C}$.

The choice of $\overline{c}$ depends on the individual subsidy surplus, which is the quality improvement of an incentivized agent minus the subsidy cost, denote $r_{\boldsymbol{x}} = \boldsymbol{\theta}^T\boldsymbol{x}$, then [1]

$$s(r_{\boldsymbol{x}}, f, G_d) := l(\tau) - l(r_{\boldsymbol{x}})\mathbf{1}\{\boldsymbol{x} \in \mathcal{M}(f)\} - [1 - l(r_{\boldsymbol{x}})]\mathbf{1}\{\boldsymbol{x} \notin \mathcal{M}(f)\} - \frac{(\tau - r_{\boldsymbol{x}})\triangle c_{i_A}^*}{(P^T\boldsymbol{\theta})_{i_A}}, \quad \text{(D.5)}$$

which is because when agents break tie choosing the action with the largest improvement, we have

$$\boldsymbol{\theta}(\boldsymbol{x} + \hat{P}\boldsymbol{a}_A^*(\boldsymbol{x})) = \tau.$$

When the minimum effective discount value is chosen, and the condition

$$l(\tau) - l(\underline{r}_f) \leq \frac{(\tau - \underline{r}_f)\triangle c_{i_A}^*}{(P^T\boldsymbol{\theta})_{i_A}} = (\tau - \underline{r}_f)\left[\frac{c_{i_A}}{(P^T\boldsymbol{w})_{i_A}} - \frac{c_{i_C}}{(P^T\boldsymbol{w})_{i_C}}\right] \quad \text{(D.6)}$$

[1] If $\boldsymbol{x} \in \mathcal{M}(f)$, incentivizing this agent will result in the same decision outcome and an improvement equilibrium qualification status and thus the subsidy benefit is $l(\tau) - l(r_{\boldsymbol{x}})$; if $\boldsymbol{x} \notin \mathcal{M}(f)$, subsidizing this agent will change the decision outcome from 0 to 1, and the subsidy benefit is $l(\tau) - [1 - l(r_{\boldsymbol{x}})]$. When applying the minimum effective discount value, the agent's equilibrium action cost is the same in AS and CS outcomes, and thus $\boldsymbol{x} \in \mathcal{M}(f)$ are incentivized to improve.

holds, all incentivized agents satisfy $\boldsymbol{x} \in \mathcal{M}(f)$ and $s(r_{\boldsymbol{x}}, f, G_d) = l(\tau) - l(r_{\boldsymbol{x}}) - \frac{(\tau - r_{\boldsymbol{x}})\triangle c^*_{i_A}}{(P^T\boldsymbol{\theta})_{i_A}}$, which is concave in $r, \forall r \leq \tau$ since l is convex on $[0, \tau]$. A rational decision maker will make sure that an agent with $r_{\boldsymbol{x}} \geq 0.5$ is in $\mathcal{M}(f)$, and $r_{\boldsymbol{x}} < 0.5$ is not. And similar to the case in Theorem 15, agents that fully spends $\bar{c}$ but still need $(\boldsymbol{a}_A)_{i_C} > 0$ are suggested to stick with their CS best responses.

The decision maker chooses $\bar{c}$ by

$$\bar{c} = (\tau - \underline{r})\triangle c^*_{i_A}/(P^T\boldsymbol{\theta})_{i_A}, \text{ where } \underline{r} = \arg\min_r \; s.t. \; s(r, f, G) \geq 0,$$

intuitively, it incentivizes every agent with non-negative individual subsidy surplus.

Here we highlight some of the key reasons why the mechanism is still IC, IR and satisfies $S(f, G)$ if the condition in (D.6) does not hold.

In fact, when $\boldsymbol{w} = \boldsymbol{\theta}$ in f, we can assume that a rational decision maker makes sure if $\boldsymbol{x} \notin \mathcal{M}(f)$, then $l(r_{\boldsymbol{x}}) < 0.5 \Leftrightarrow 1 - l(r_{\boldsymbol{x}}) > l(r_{\boldsymbol{x}})$. As a result, we know that

$$\bar{s}(r, f, G) = l(\tau) - l(r) - \frac{(\tau - r)\triangle c_{i_A}}{(P^T\boldsymbol{\theta})_{i_A}} \geq s(r, f, G),$$

is concave in r and

$$\underline{s}(r, f, G) = l(\tau) + l(r) - 1 - \frac{(\tau - r)\triangle c_{i_A}}{(P^T\boldsymbol{\theta})_{i_A}} \leq s(r, f, G),$$

is increasing in $r, \forall r \; s.t. \; l(r) < 0.5$. Therefore, if the condition in (D.6) does not hold, we have $l(\underline{r}) < 0.5$, where $\underline{r} = \arg\min_r \; s.t. \; s(r, f, G) \geq 0$ we can also conclude that $r_{\boldsymbol{x}} in [\underline{r}, \tau]$ satisfies $s(r_{\boldsymbol{x}}, f, G) \geq 0$, i.e., every agent incentivized has non-negative individual subsidy surplus. □

When (D.6) does not hold or f incentivizes improvement, the mechanism designer can approximate the optimal $\triangle c_{i_A}$ by doing a grid search on the value of $1/(c_{i_A} - \triangle c_{i_A})$ with step size ϵ to find the corresponding $\triangle c_{i_A}$ and the related optimal $\bar{c}$ like in Theorem 16, which guarantees $\max_G S(f, G) - \max S_{grid}(f, G_{grid})$ is $O(\epsilon)$ if $\max_v \int_v^{v+\epsilon} p_R(r)dr$ is $O(\epsilon)$.

This is because if the optimal discounted value is $\tilde{\triangle c}_{i_A}$, there is one scanned value $1/(c_{i_A} - \triangle c_{i_A})$ that is at most $\epsilon/2$ away from $1/(c_{i_A} - \tilde{\triangle c}_{i_A})$, meaning that

1. the optimal scanned value at worst failed to incentivize $O(\epsilon)$ of agents to improve with a highest subsidy benefit of $O(\epsilon)$ and possibly an infinitesimal extra amount of subsidy cost;

2. the optimal scanned value at worst paid an extra subsidy cost of $O(\epsilon)$ ($O(\epsilon)$ probability mass of agents with individual payment no more than 1) and has no improved a highest subsidy benefit.

D.3 Supplementary Material for Section 5.4

D.3.1 Proof of Theorem 17

Proof. Recall that the AS utility of the decision maker is

$$U_A^{(reg)}(f) = \int_{\mathcal{X}} \mathbb{E}_\sigma\big[- \big(f(\boldsymbol{x} + P\boldsymbol{a}_A^*(\boldsymbol{x})) - y_A'\big)^2\big]p(\boldsymbol{x})d\boldsymbol{x} - H(G),$$

which if we rewrite the equilibrium individual error as

$$\mathcal{E}(f, \boldsymbol{a}, \boldsymbol{x}) = [\boldsymbol{w}^T(\boldsymbol{x} + P\boldsymbol{a}) - \boldsymbol{\theta}^T(\boldsymbol{x} + \hat{P}\boldsymbol{a})]^2 + err(\sigma),$$

the objective becomes

$$U_A^{(reg)}(f) = \int_{\mathcal{X}} -\mathcal{E}(f, \boldsymbol{a}, \boldsymbol{x})p(\boldsymbol{x})d\boldsymbol{x} - H(G).$$

The integral part is non-concave for general $p(\boldsymbol{x})$.

On the other hand, for a target AS best response where $\alpha < 1$, $\boldsymbol{a}_A = \alpha\frac{B}{c_j - \triangle c_j^*}\boldsymbol{e}_j + (1-\alpha)\frac{B}{c_{i_C}}\boldsymbol{e}_{i_C}$, we have $H(G) = \bar{c} = \frac{\alpha B \triangle c_j^*}{c_j - \triangle c_j^*}$, and the where $H(G)$ is linear in $\boldsymbol{a}$. for a target AS best response where $\alpha > 1$, $\boldsymbol{a}_A = \alpha\frac{B}{c_j - \triangle c_j^*}\boldsymbol{e}_j$, we have

$$\frac{\alpha}{c_j - \triangle c_j^*} = \frac{1}{c_j - \triangle c_j} \Leftrightarrow \triangle c_j = \frac{(\alpha - 1)c_j + \triangle c_j^*}{\alpha},$$

and

$$H(G) = \frac{B\triangle c_j}{c_j - \triangle c_j} = \frac{B((\alpha - 1)c_j + \triangle c_j^*)}{c_j - \triangle c_j^*}.$$

We can similarly show that $H(G)$ is piece-wise affine in $\boldsymbol{a}$ and thus the entire objective is non-concave and the problem is non-convex. □

D.3.2 Proof of Theorem 18

Proof. This algorithm has two loops, making it finish in polynomial time.

The outer loop enumerates through all improvement action dimensions and chooses the minimum effective discount amount to incentivize the agents to take an AS best response $\boldsymbol{a}_A = \alpha\frac{B}{c_j - \triangle c_j^*}\boldsymbol{e}_j + (1 - \alpha)\frac{B}{c_{i_C}}\boldsymbol{e}_{i_C}$, where $\alpha < 1$. The inner loop grid searches the α values for each j to see if an IC and IR and $S(f, G) > 0$, computes the corresponding $\bar{c}$ and keeps track of the G that generates the largest $S(f, G)$. □

D.3.3 Proof of Theorem 19

Proof. In the special case, if improvement is incentivized by the mechanism, it is the dominant strategy to use the minimum effective discount amount, since a higher discount achieves the same error reduction but a higher subsidy cost.

For an AS best response $\boldsymbol{a}_A = \alpha \frac{B}{c_j - \triangle c_j^*} \boldsymbol{e}_j + (1-\alpha) \frac{B}{c_{i_C}} \boldsymbol{e}_{i_C}$, where $\alpha < 1$, the alternative form of individual subsidy benefit is the reduction in the expected prediction error

$$(\boldsymbol{\theta}^T P \boldsymbol{a}_C^*)^2 - (1-\alpha)^2 (\boldsymbol{\theta}^T P \boldsymbol{a}_C^*)^2,$$

the subsidy cost is $H(G) = \bar{c} = \frac{\alpha B \triangle c_j^*}{c_j - \triangle c_j^*}$, and thus we have the alternative individual subsidy urplus

$$s(\alpha) = (2\alpha - \alpha^2)(\boldsymbol{\theta}^T P \boldsymbol{a}_C^*)^2 - \alpha B \triangle c_{i_A} (c_{i_A} - \triangle c_{i_A})^{-1}.$$

□

D.4 Supplementary Material for Section 5.5

D.4.1 Proof of Theorem 20

Proof. The DP gap is only related to $f(\boldsymbol{z})$ but not y or y', when the two groups have the same action cost but group 2 is disadvantaged in attribute, the implicit threshold (the lower side boundary of $\mathcal{M}^d(f)$, $\hat{\tau}_L$) is the same for both groups and from the definition of attribute disadvantage, $PR^{(1)} = 1 - F^{(1)}(\hat{\tau}_L) > 1 - F^{(2)}(\hat{\tau}_L) = PR^{(2)}$, and we know that the DP gap exists.

When f incentivizes gaming, the reason of an EO gap is similar as above $TPR^{(1)} = 1 - F_+^{(1)}(\hat{\tau}_L) > 1 - F_+^{(2)}(\hat{\tau}_L) = TPR^{(2)}$. If f incentivizes improvement, then the EO gap depends on both the CS TPR in both groups, the CS PR in both groups, and the AS quality improvement in both groups. For example, if G only not incentivize agents in the manipulation margins, then

$$TPR_A = 1 - FNR_A = 1 - FNR_C \cdot \frac{PR_C}{PR_A} = 1 - \frac{(1 - TPR_C) \cdot PR_C}{\triangle Q_A + PR_C},$$

and we know that the AS EO gap depends on $\triangle Q_A^{(1)}, \triangle Q_A^{(2)}$ which is based on $p^{(1)}(\boldsymbol{x})$ and $p^{(2)}(\boldsymbol{x})$ and we can not easily conclude the EO gap changes. □

D.4.2 Proof of Theorem 21

Proof. Part (1) is obvious since gaming results in no quality gain, and $\mathcal{M}^{(1)}(f) \supseteq \mathcal{M}^{(2)}(f)$ results in the quality gain gap if f incentivizes improvement and the DP gap no matter f incentivizes

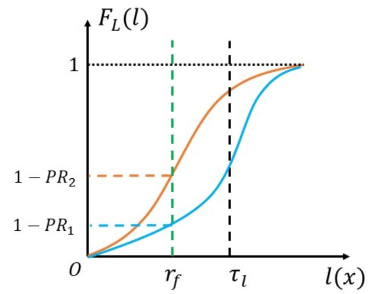


Figure D.2: An illustration of the CS DP gap when group 2 is disadvantaged in attributes.

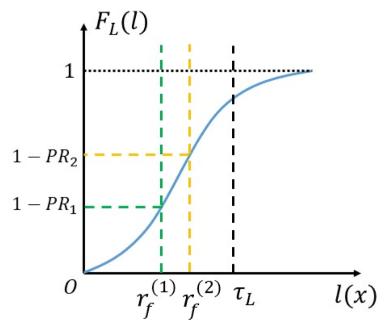


Figure D.3: An illustration of the CS DP gap when group 2 is disadvantaged in cost.

gaming or improvement.

If f incentivizes gaming, the reason of the EO gap is similar to that of the DP gap.

If f incentivizes improvement, again we can look at the formula for AS TPR

$$TPR_A = 1 - FNR_A = 1 - FNR_C \cdot \frac{PR_C}{PR_A} = 1 - \frac{(1 - TPR_C) \cdot PR_C}{\triangle Q_A + PR_C},$$

group 1 has a higher TPR_C, and a higher $\triangle Q_A / PR_C$, and thus a higher TPR_A and the EO gap always exists. □

D.4.3 Proof and Discussion on Theorem 22

Proof. If group 2 is disadvantaged in cost, then it is cheaper to incentivize a group 1 agent than a group 2 agent to get the same qualification status improvement, and thus the decision maker subsidizes more group 1 agents and creates a quality gain gap. □

For DP gap, if G only incentivizes agents in the manipulation margins, the agents' CS or AS equilibrium decision remains the same.

For EO gap, we note that with G, the AS true positive increases in both groups and how the EO gap in classification changes depends on both the CS positive decision rate and the qualification status improvement and we do not have certain conclusions. For example, if G only not incentivize agents in the manipulation margins, then

$$TPR_A = 1 - FNR_A = 1 - FNR_C \cdot \frac{PR_C}{PR_A} = 1 - \frac{(1 - TPR_C) \cdot PR_C}{\triangle Q_A + PR_C},$$

we have $PR_C^{(1)} > PR_C^{(2)}$, $\triangle Q_A^{(1)} > \triangle Q_A^{(2)}$ and we can not easily conclude the $\texttt{EO}$ gap changes. $FNR_A = 1 - FNR_C \cdot \frac{PR_C}{PR_A}$ because all false negative agents in CS remain to be false negatives in AS (the positive individuals with lower attribute than the lower side boundary of manipulation margins).

D.5 Supplementary Material for Section 5.6

D.5.1 Proof of Theorem 23

Proof. We still need G to be IR for the decision maker, where the maximum tax a rational decision maker accepts is the subsidy benefit $\mathcal{T}(G) \leq S(f, G) + H(G)$, and the BB condition requires $S(f, G) + H(G) \geq \mathcal{T}(G) \geq H(G)$. So, as long as $S(f, G) \geq 0$, there is an IC, IR, and BB third party mechanism. Therefore, finding the optimal IC, IR, and BB third party mechanism is the same as

$$\text{maximize}_G \, W(f, G), \;\; \text{subject to } S(f, G) \geq 0,$$

and if $S(f, G) > 0$ the mechanism can further improve its objective by setting the surplus at 0. □

D.5.2 Proof of Theorem 25

Proof. For Part (1), the fairness oriented third party can implement the ideal mechanism on group 2 and even further subsidize other group 2 agents to reduce the gap while avoiding subsidizing more group 1 agents to enlarge the fairness gaps.

For Part (2), any ideal mechanism makes sure the efficiency oriented third party has "remaining budget" to incentivize more agents to improve compared to AS-dm outcome and thus has the strictly highest equilibrium social quality improvement. □

D.5.3 Proof of Theorem 26

Proof. If $h_A^{(1)}(\boldsymbol{a}) = h_A^{(2)}(\boldsymbol{a}), \forall \boldsymbol{a}$, then the equilibrium feature and attribute distribution are the same for both groups, and thus there is no fairness gap. Meanwhile, the subsidy benefit are the same in both groups, so the overall benefit is $S(f, G^{(1)}) + H(G^{(1)})$, and the overall subsidy cost is $p^{(1)} H(G^{(1)}) + p^{(2)} H(G^{(2)})$. □

Bibliography

[1] Daron Acemoglu, Vasco M Carvalho, Asuman Ozdaglar, and Alireza Tahbaz-Salehi. The network origins of aggregate fluctuations. *Econometrica*, 80(5):1977–2016, 2012.

[2] Daron Acemoglu, Asuman Ozdaglar, and Alireza Tahbaz-Salehi. Networks, shocks, and systemic risk. Technical report, National Bureau of Economic Research, 2015.

[3] Nizar Allouch. On the private provision of public goods on networks. *Journal of Economic Theory*, 157:527–552, 2015.

[4] Hamidreza Alvari, Alireza Hajibagheri, and Gita Sukthankar. Community detection in dynamic social networks: A game-theoretic approach. 08 2014.

[5] Tom M Apostol. *Mathematical analysis; 2nd ed.* Addison-Wesley Series in Mathematics. Addison-Wesley, Reading, MA, 1974.

[6] Adarsh Barik and Jean Honorio. Provable computational and statistical guarantees for efficient learning of continuous-action graphical games. *CoRR*, abs/1911.04225, 2019.

[7] Adarsh Barik and Jean Honorio. Provable sample complexity guarantees for learning of continuous-action graphical games with nonparametric utilities. *CoRR*, abs/2004.01022, 2020.

[8] Abdellah Bnouhachem, Hafida Benazza, and Mohamed Khalfaoui. An inexact alternating direction method for solving a class of structured variational inequalities. *Applied Mathematics and Computation*, 219:7837–7846, 03 2013.

[9] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[10] Yann Bramoullé, Rachel Kranton, and Martin D'amours. Strategic interaction and networks. *American Economic Review*, 104(3):898–930, 2014.

[11] Yann Bramoullé and Rachel Kranton. Games Played on Networks. AMSE Working Papers 1530, Aix-Marseille School of Economics, France, March 2015.

[12] Mark Braverman and Sumegha Garg. The role of randomness and noise in strategic classification. In *1st Symposium on Foundations of Responsible Computing*, 2020.

[13] Piotr Bródka, Krzysztof Skibicki, Przemysław Kazienko, and Katarzyna Musiał. A degree centrality in multi-layered social network. In *2011 International Conference on Computational Aspects of Social Networks (CASoN)*, pages 237–242. IEEE, 2011.

[14] Gavin Brown, Shlomi Hod, and Iden Kalemaj. Performative prediction in a stateful world. *CoRR*, abs/2011.03885, 2020.

[15] Michael Brückner and Tobias Scheffer. Stackelberg games for adversarial prediction problems. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, page 547–555, New York, NY, USA, 2011. Association for Computing Machinery.

[16] Michael Brückner, Christian Kanzow, and Tobias Scheffer. Static prediction games for adversarial learning problems. *The Journal of Machine Learning Research*, 13:2617–2654, 09 2012.

[17] Edward Buckley and Rachel Croson. Income and wealth heterogeneity in the voluntary provision of linear public goods. *Journal of Public Economics*, 90(4-5):935–955, 2006.

[18] Ozan Candogan, Kostas Bimpikis, and Asuman Ozdaglar. Optimal pricing in networks with externalities. *Computing Research Repository - CORR*, 60, 01 2011.

[19] Ozan Candogan, Kostas Bimpikis, and Asuman Ozdaglar. Optimal pricing in networks with externalities. *Operations Research*, 60(4):883–905, 2012.

[20] Yatong Chen, Jialu Wang, and Yang Liu. Strategic recourse in linear classification. *arXiv preprint arXiv:2011.00355*, 2020.

[21] George Christodoulou and Elias Koutsoupias. The price of anarchy of finite congestion games. In *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing*, STOC '05, page 67–73, New York, NY, USA, 2005. Association for Computing Machinery.

[22] Aaron Clauset, Cristopher Moor, and M.E.J. Newman. Hierarchical structure and the prediction of missing links in networks. *Nature*, 453:98–101, 2008.

[23] Richard Cottle, Jong-Shi Pang, and Richard E Stone. The linear complementarity problem / richard w. cottle, jong-shi pang, richard e. stone. *SERBIULA (sistema Librum 2.0)*, 03 2019.

[24] Greg Coxson. The p-matrix problem is co-np-complete. *Math. Program.*, 64:173–178, 03 1994.

[25] Erik Demaine, Mohammadtaghi Hajiaghayi, Hamid Mahini, and Morteza Zadimoghaddam. The price of anarchy in network creation games. *ACM Transactions on Algorithms - TALG*, 8:1–13, 04 2012.

[26] Jinshuo Dong, Aaron Roth, Zachary Schutzman, Bo Waggoner, and Zhiwei Steven Wu. Strategic classification from revealed preferences. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 55–70, 2018.

[27] Dmitriy Drusvyatskiy and Lin Xiao. Stochastic optimization with decision-dependent distributions. *Mathematics of Operations Research*, 2022.

[28] F Facchinei and J S. Pang. Finite-dimensional variational inequalities and complementarity problems-volume i. *Springer Series in Operations Research*, 1, 01 2003.

[29] Daniel Gabay and Bertrand Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2:17–40, 12 1976.

[30] Andrea Galeotti, Benjamin Golub, and Sanjeev Goyal. Targeting interventions in networks. *Econometrica*, 88:2445–2471, 01 2020.

[31] James Gentle. Matrix analysis and applied linear algebra by carl d. meyer; numerical linear algebra by lloyd trefethen; david bau; applied numerical linear algebra by james w. demmel. *Journal of the American Statistical Association*, 96:1136–1137, 09 2001.

[32] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proc. Natl. Acad. Sci.*, 99:7821–7826, 2002.

[33] Roland Glowinski and J. Oden. Numerical methods for nonlinear variational problems. *Journal of Applied Mechanics*, 52:739–, 01 1985.

[34] Nika Haghtalab, Nicole Immorlica, Brendan Lucier, and Jack Z. Wang. Maximizing welfare with incentive-aware evaluation mechanisms. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 160–166. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track.

[35] Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. Strategic classification. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, ITCS '16, page 111–122, New York, NY, USA, 2016. Association for Computing Machinery.

[36] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, 2016.

[37] Keegan Harris, Hoda Heidari, and Steven Z. Wu. Stateful strategic regression. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 28728–28741. Curran Associates, Inc., 2021.

[38] Bing-Sheng He. Parallel Splitting Augmented Lagrangian Methods for Monotone Structured Variational Inequalities. *Computational Optimization and Applications*, 42:195–212, 02 2009.

[39] Bing-Sheng He, Haiqin Yang, and S. Wang. Alternating direction method with self-adaptive penalty parameters for monotone variational inequalities. *Journal of Optimization Theory and Applications*, 106:337–356, 08 2000.

[40] Bing-Sheng He and Xiaoming Yuan. On the O(1/t) Convergence Rate of Alternating Direction Method. *Optimization Online*, 05 2012.

[41] Ashish R Hota and Shreyas Sundaram. Interdependent security games on networks under behavioral probability weighting. *IEEE Transactions on Control of Network Systems*, 5(1):262–273, 2018.

[42] Lily Hu, Nicole Immorlica, and Jennifer Wortman Vaughan. The disparate effects of strategic manipulation. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, page 259–268, New York, NY, USA, 2019. Association for Computing Machinery.

[43] Zachary Izzo, Lexing Ying, and James Zou. How to learn when data reacts to your model: Performative gradient descent. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 4641–4650. PMLR, 18–24 Jul 2021.

[44] Zachary Izzo, James Zou, and Lexing Ying. How to learn when data gradually reacts to your model. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 3998–4035. PMLR, 28–30 Mar 2022.

[45] Matthew O Jackson and Yves Zenou. Games on networks. In *Handbook of game theory with economic applications*, volume 4, pages 95–163. Elsevier, 2015.

[46] Libin Jiang, Venkat Anantharam, and Jean Walrand. How bad are selfish investments in network security? *Networking, IEEE/ACM Transactions on*, 19:549 – 560, 05 2011.

[47] Kun Jin, Mohammad Mahdi Khalili, and Mingyan Liu. Games on networks with community structure: Existence, uniqueness and stability of equilibria. In *2020 American Control Conference (ACC)*, pages 4664–4670, 2020.

[48] Kun Jin and Mingyan Liu. Multi-planner intervention in network games with community structures. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 2374–2379, 2021.

[49] Kun Jin, Yevgeniy Vorobeychik, and Mingyan Liu. Multi-scale games: Representing and solving games on networks with group structure. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 5497–5505. AAAI Press, 2021.

[50] Kun Jin, Tongxin Yin, Charles A. Kamhoua, and Mingyan Liu. Network games with strategic machine learning. In Branislav Bošanský, Cleotilde Gonzalez, Stefan Rass, and Arunesh Sinha, editors, *Decision and Game Theory for Security*, pages 118–137, Cham, 2021. Springer International Publishing.

[51] Kun Jin, Xueru Zhang, Mohammad Mahdi Khalili, Parinaz Naghizadeh, and Mingyan Liu. Incentive mechanisms for strategic classification and regression problems. In David M. Pennock, Ilya Segal, and Sven Seuken, editors, *EC '22: The 23rd ACM Conference on Economics and Computation, Boulder, CO, USA, July 11 - 15, 2022*, pages 760–790. ACM, 2022.

[52] Michael J. Kearns, Michael L. Littman, and Satinder P. Singh. Graphical models for game theory. In *Conference in Uncertainty in Artificial Intelligence*, page 253–260, 2001.

[53] Mohammad Mahdi Khalili, Xueru Zhang, and Mingyan Liu. Public good provision games on networks with resource pooling. In *Network Games, Control, and Optimization*, pages 271–287. Springer, 2019.

[54] Jon Kleinberg and Manish Raghavan. How do classifiers induce agents to invest effort strategically? *ACM Transactions on Economics and Computation*, 8:1–23, 11 2020.

[55] Jerzy Kyparisis. Uniqueness and differentiability of solutions of parametric nonlinear complementarity problems. *Mathematical Programming*, 36(1):105–113, 1986.

[56] Richard J La. Interdependent security with strategic agents and cascades of infection. *IEEE/ACM Transactions on Networking*, 24(3):1378–1391, 2016.

[57] Yan Leng, Xiaowen Dong, and Alex Pentland. Learning quadratic games on networks. *CoRR*, abs/1811.08790, 2018.

[58] Jure Leskovec, Jon M. Kleinberg, and Christos Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Trans. Knowl. Discov. Data*, 1:2, 2007.

[59] Qiang Li and Hoi-To Wai. State dependent performative prediction with stochastic approximation. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 3164–3186. PMLR, 28–30 Mar 2022.

[60] Qiang LI, Chung-Yiu Yau, and Hoi-To Wai. Multi-agent performative prediction with greedy deployment and consensus seeking agents. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 38449–38460. Curran Associates, Inc., 2022.

[61] P. Lions and Bertrand Mercier. Mercier, b.: Splitting algorithms for the sum of two nonlinear operators. siam j. numer. anal. 16(6), 964-979. *Siam Journal on Numerical Analysis - SIAM J NUMER ANAL*, 16:964–979, 12 1979.

[62] Patrick Mcsweeney, Kishan Mehrotra, and Jae Oh. *Game-Theoretic Framework for Community Detection*, pages 1–16. 01 2017.

[63] Emerson Melo. A variational approach to network games. *SSRN Electronic Journal*, 05 2017.

[64] Celestine Mendler-Dünner, Juan Perdomo, Tijana Zrnic, and Moritz Hardt. Stochastic optimization for performative prediction. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 4929–4939. Curran Associates, Inc., 2020.

[65] John Miller, Smitha Milli, and Moritz Hardt. Strategic classification is causal modeling in disguise. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6917–6926. PMLR, 13–18 Jul 2020.

[66] John P Miller, Juan C Perdomo, and Tijana Zrnic. Outside the echo chamber: Optimizing the performative risk. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 7710–7720. PMLR, 18–24 Jul 2021.

[67] Smitha Milli, John Miller, Anca D. Dragan, and Moritz Hardt. The social cost of strategic classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, page 230–239, New York, NY, USA, 2019. Association for Computing Machinery.

[68] R. A. Miura-Ko, B. Yolken, J. Mitchell, and N. Bambos. Security decision-making among interdependent organizations. In *2008 21st IEEE Computer Security Foundations Symposium*, pages 66–80, June 2008.

[69] Katta G Murty and Feng-Tien Yu. *Linear complementarity, linear and nonlinear programming*, volume 3. Citeseer, 1988.

[70] Parinaz Naghizadeh and Mingyan Liu. Budget balance or voluntary participation? incentivizing investments in interdependent security games. In *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1102–1109, 2014.

[71] Parinaz Naghizadeh and Mingyan Liu. Exit equilibrium: Towards understanding voluntary participation in security games. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9, 2016.

[72] Parinaz Naghizadeh and Mingyan Liu. Opting out of incentive mechanisms: A study of security as a non-excludable public good. *IEEE Transactions on Information Forensics and Security*, 11:2790–2803, 12 2016.

[73] Parinaz Naghizadeh and Mingyan Liu. On the uniqueness and stability of equilibria of network games. In *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 280–286. IEEE, 2017.

[74] Parinaz Naghizadeh and Mingyan Liu. Provision of public goods on networks: on existence, uniqueness, and centralities. *IEEE Transactions on Network Science and Engineering*, 5(3):225–236, 2018.

[75] Adhyyan Narang, Evan Faulkner, Dmitriy Drusvyatskiy, Maryam Fazel, and Lillian J. Ratliff. Multiplayer performative prediction: Learning in decision-dependent games. *CoRR*, abs/2201.03398, 2022.

[76] M Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 74:036104, 10 2006.

[77] Mark Newman. Detecting community structure in networks. *Eur Phys J*, 38, 03 2004.

[78] Tore Opsahl, Filip Agneessens, and John Skvoretz. Node centrality in weighted networks: Generalizing degree and shortest paths. *Social networks*, 32(3):245–251, 2010.

[79] Francesca Parise and Asuman Ozdaglar. A variational inequality framework for network games: Existence, uniqueness, convergence and sensitivity analysis. *Games and Economic Behavior*, 2019.

[80] Jaeok Park and Mihaela van der Schaar. Intervention mechanism design for networks with selfish users. *CoRR*, abs/1008.0147, 2010.

[81] Juan Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. Performative prediction. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7599–7609. PMLR, 13–18 Jul 2020.

[82] Reilly Raab and Yang Liu. Unintended selection: Persistent qualification rate disparities and interventions. *Advances in Neural Information Processing Systems*, 34:26053–26065, 2021.

[83] Mitas Ray, Lillian J. Ratliff, Dmitriy Drusvyatskiy, and Maryam Fazel. Decision-dependent risk minimization in geometrically decaying dynamic environments. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022*, pages 8081–8088. AAAI Press, 2022.

[84] Yann Rébillé and Lionel Richefort. Equilibrium Uniqueness in Network Games with Strategic Substitutes. working paper or preprint, February 2012.

[85] US Federal Reserve. Report to the congress on credit scoring and its effects on the availability and affordability of credit. *Board of Governors of the Federal Reserve System*, 2007.

[86] Tim Roughgarden. The price of anarchy is independent of the network topology. *Journal of Computer and System Sciences*, 67, 02 2003.

[87] Tim Roughgarden. Intrinsic robustness of the price of anarchy. *J. ACM*, 62(5), nov 2015.

[88] Gesualdo Scutari, Francisco Facchinei, Jong-Shi Pang, and Daniel P Palomar. Real and complex monotone communication games. *IEEE Transactions on Information Theory*, 60(7):4197–4231, 2014.

[89] Gesualdo Scutari, Francisco Facchinei, Jong-Shi Pang, and Daniel P Palomar. Real and complex monotone communication games. *IEEE Transactions on Information Theory*, 60(7):4197–4231, 2014.

[90] Gesualdo Scutari, Daniel P Palomar, and Sergio Barbarossa. Asynchronous iterative water-filling for gaussian frequency-selective interference channels. *IEEE Transactions on Information Theory*, 54(7):2868–2878, 2008.

[91] Yonadav Shavit, Benjamin Edelman, and Brian Axelrod. Causal strategic linear regression. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8676–8686. PMLR, 13–18 Jul 2020.

[92] Marilyn Strathern. 'Improving Ratings': Audit in the British University System. *European review*, 5(3):305–321, 1997.

[93] Michael Tsatsomeros. Generating and detecting matrices with positive principal minors. *Asian-Inf.-Sci.-Life*, 1, 01 2002.

[94] Paul Tseng. Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. *Mathematical Programming*, 48:249–263, 03 1990.

[95] Yevgeniy Vorobeychik and Joshua Letchford. Securing interdependent assets. *Journal of Autonomous Agents and Multiagent Systems*, 29:305–333, 2015.

[96] Killian Wood, Gianluca Bianchin, and Emiliano Dall'Anese. Online projected gradient descent for stochastic optimization with decision-dependent distributions. *IEEE Control Systems Letters*, 6:1646–1651, 2022.

[97] Sixie Yu, Kai Zhou, P. Jeffrey Brantingham, and Yevgeniy Vorobeychik. Computing equilibria in binary networked public goods games. In *AAAI Conference on Artificial Intelligence*, pages 2310–2317, 2020.

[98] Xiao-Ming Yuan and Min Li. An LQP-Based Decomposition Method for Solving a Class of Variational Inequalities. *SIAM Journal on Optimization*, 21:1309–1318, 10 2011.

[99] Hanrui Zhang and Vincent Conitzer. Incentive-aware pac learning. In *AAAI Conference on Artificial Intelligence*, 2021.

[100] Zhengyuan Zhou, Benjamin Yolken, Reiko Ann Miura-Ko, and Nicholas Bambos. A game-theoretical formulation of influence networks. In *2016 American Control Conference (ACC)*, pages 3802–3807. IEEE, 2016.

[101] Tijana Zrnic, Eric Mazumdar, Shankar Sastry, and Michael Jordan. Who leads and who follows in strategic classification? In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 15257–15269. Curran Associates, Inc., 2021.