

# Inverse Game Theory: An Incenter-Based Approach

## Abstract

Estimating player utilities from observed equilibria is crucial for many applications. Existing approaches to tackle this problem are either limited to specific games or do not scale well with the number of players. Our work addresses these issues by proposing a novel utility estimation method for general multi-player non-cooperative games. Our main idea consists in reformulating the inverse game problem as an inverse variational inequality problem and in selecting among all utility parameters consistent with the data, the so-called *incenter*. We show that the choice of the incenter can produce parameters that are most robust to the observed equilibrium behaviors. However, its computation is challenging, as the number of constraints in the corresponding optimization problem increases with the number of players and the behavior space size. To tackle this challenge, we propose a loss function-based algorithm, making our method scalable to games with many players or a continuous action space. Furthermore, we show that our method can be extended to incorporate prior knowledge of player utilities, and that it can handle inconsistent data, i.e., data where players do not play exact equilibria. Numerical experiments on three game applications demonstrate that our methods outperform the state of the art.

## 1 Introduction

Game theory analyzes the strategic behaviors of players using equilibrium concepts, while inverse game theory focuses on the inverse problem: *given the observed equilibrium behaviors of players, what are the possible player utilities leading to these behaviors?* Inverse game theory has garnered increasing attention [Waugh *et al.*, 2011; Kuleshov and Schrijvers, 2015; Ling *et al.*, 2019; Noti, 2021; Wu *et al.*, 2022], because understanding player utilities can lead to better-designed mechanisms and economic policies.

When estimating player utilities, the existing approaches have two main limitations: (i) most methods are tailored to *specific games*, such as matrix games [Noti, 2021; Yu *et al.*, 2022] and attacker-defender security games [Blum *et al.*,

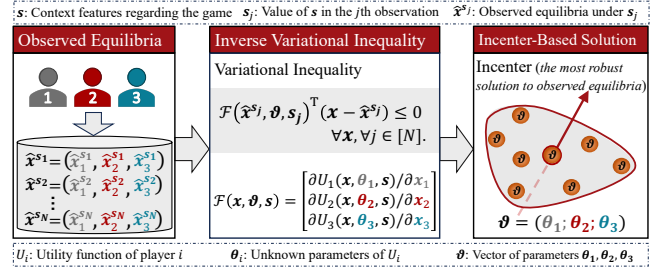


Figure 1: Our approach to utility estimation from equilibria.

2014]; (ii) many existing methods *do not scale well* with the number of players, often focusing on two-player games [Bertsimas *et al.*, 2015; Tsai *et al.*, 2016; Ling *et al.*, 2018; Ling *et al.*, 2019; Wu *et al.*, 2022].

In this work, we study general multi-player non-cooperative games, and propose utility estimation methods applicable to a broad range of games and practical applications. Figure 1 illustrates our main ideas. The player utility functions are parameterized, and we aim to estimate the vector  $\vartheta$  of these utility parameters under which the observed player behaviors constitute a Nash equilibrium. Our idea is to reformulate this as an inverse variational inequality problem: *given the observed player behaviors, what vector  $\vartheta$  ensures that these behaviors satisfy the variational inequalities?*

The crucial challenge is that even with many observations of player behaviors, there may be multiple parameter vectors  $\vartheta$  solving the inverse variational inequality problem. This raises a fundamental question: *how should we select a single  $\vartheta$  from the set of  $\vartheta$  consistent with the observed equilibria?* Existing approaches, such as [Bertsimas *et al.*, 2015], do not address this question but simply single out a  $\vartheta$  without following a guiding principle. To the contrary, we attempt to solve this problem using a more principled approach.

Our key idea is to select the *incenter* among all consistent  $\vartheta$ . This concept introduced in inverse optimization [Besbes *et al.*, 2023; Scroccaro *et al.*, 2023] describes a point that lies within a given set and is located furthest away from its boundary. We select the incenter for the following three reasons.

First, due to its geometric properties, selecting the incenter as the estimated  $\vartheta$  reduces the error in estimating the true parameter vector  $\vartheta_{\text{true}}$ . Consider a toy example where the set

of consistent  $\vartheta$  forms an equilateral triangle. When the true parameter vector  $\vartheta_{\text{true}}$  is uniformly distributed within this triangle, the incenter minimizes the expected distance to  $\vartheta_{\text{true}}$ , compared to other points in the triangle. This property generalizes to some other regular shapes, such as squares. When the set of consistent  $\vartheta$  forms an irregular shape, our experiments demonstrate that the incenter consistently outperforms the  $\vartheta$  estimated by other approaches.

Second, we can efficiently compute the incenter of the set of all consistent  $\vartheta$  using convex optimization. When the action space is continuous, the optimization problem of finding the exact incenter is challenging to solve, since it may involve infinitely many constraints. We convert it to an unconstrained loss minimization problem by defining a novel loss function. Then, we propose a first-order algorithm to minimize the loss, and thus estimate the parameter vector  $\vartheta$ .

Third, our loss function-based approach of approximating the incenter has the advantage of handling settings where players do not play exact equilibria. When players are bounded-rational, it is possible that no  $\vartheta$  satisfies all variational inequalities. In this case, our loss function can still guide the search for  $\vartheta$  towards the vector that minimizes the degree of violating the constraints defined based on variational inequalities.

We further explore a class of games where the player utility functions exhibit a specific structural property (e.g., monotonicity of gradients). We extend our parameter estimation method to incorporate this property by leveraging techniques from *semidefinite programming*.

We summarize our contributions as follows.

- We propose a novel framework for solving inverse equilibrium problems, through integrating techniques from variational inequalities and inverse optimization. We further extend our framework to account for specific structural property of player utility functions.
- Our framework can be broadly applied to many multi-player non-cooperative games and traffic routing games in the transportation field. It imposes no restrictions on the number of players in these game scenarios.
- We conduct extensive numerical experiments on three game applications: Bertrand-Nash price competition [Narahari *et al.*, 2009], aggregative games [Jensen, 2018], and network traffic games [Patriksson, 2015]. Evaluation using different metrics demonstrates that our methods outperform the state of the art.

## 2 Related Work

**Inverse Game Theory** [Kuleshov and Schrijvers, 2015] is one of the pioneering studies in this field, and investigated the tractability of estimating player utilities in succinct games. Most subsequent studies addressed the inverse game problem within specific games, such as normal-form games [Ling *et al.*, 2018; Noti, 2021; Yu *et al.*, 2022], two-player zero-sum games [Ling *et al.*, 2019], and attacker-defender security games [Blum *et al.*, 2014; Haghtalab *et al.*, 2016; Wu *et al.*, 2022]. [Ling *et al.*, 2018] proposed an end-to-end parameter estimation framework that applies to two-player normal and extensive form games. Different from these stud-

ies, our methods are applicable to a broad range of non-cooperative games and applications, and do not impose restrictions on the number of players.

Our work is related to [Bertsimas *et al.*, 2015], which utilized inverse optimization to estimate utility parameters in Bertrand competition. That approach can handle only a small number of players and behavior observations (e.g., it requires dealing with over 120,000 constraints when there are 4 players and 500 observations). As will be shown in experiments, our incenter-based method is more accurate and scalable.

**Inverse Optimization** The goal of inverse optimization [Heuberger, 2004; Chan *et al.*, 2023] is to identify the parameters of an optimization model that render observed decisions approximately or exactly optimal. To estimate the parameters, the literature in this field has proposed various loss functions, such as Predictability Loss [Aswani *et al.*, 2018], Suboptimality Loss [Mohajerin Esfahani *et al.*, 2018; Besbes *et al.*, 2023], Augmented Suboptimality Loss [Scroccaro *et al.*, 2023], and Variational Inequality Loss [Bertsimas *et al.*, 2015]. Our work is inspired by [Scroccaro *et al.*, 2023]. The main difference is that we focus on game scenarios with multiple decision makers, rather than optimization problems involving a single decision maker. We seek parameters that make the observed data equilibria, rather than the optimal solutions to an optimization problem.

## 3 Problem Formulation

We represent a multi-player non-cooperative game using a tuple  $\mathcal{G} := \{\mathcal{I}, \{\mathcal{X}_i\}_{i \in \mathcal{I}}, \{U_i\}_{i \in \mathcal{I}}\}$ . Here,  $\mathcal{I} := \{1, \dots, p\}$  denotes the set of players. Each player  $i \in \mathcal{I}$  chooses an action  $\mathbf{x}_i$  from its feasible action set  $\mathcal{X}_i \subseteq \mathbb{R}^{m_i}$ , where  $m_i \in \mathbb{Z}^+$  denotes the dimension of the action. Let  $\mathbf{x} := (\mathbf{x}_1, \dots, \mathbf{x}_p) \in \mathcal{X}$  be the action profile, where  $\mathcal{X} = \prod_{i \in \mathcal{I}} \mathcal{X}_i$  is the global action set. We consider the setting where the game is played in different *contexts*, representing exogenous (but observable) conditions, e.g., variations in GDP and seasonality in Bertrand-Nash games. Given a contextual feature  $\mathbf{s} \in \mathcal{S}$ , we assume each player  $i$  seeks to maximize its own utility function  $U_i(\mathbf{x}_i, \mathbf{x}_{-i}, \mathbf{s}) : \mathcal{X} \times \mathcal{S} \rightarrow \mathbb{R}$ . The utility depends on its own action  $\mathbf{x}_i \in \mathcal{X}_i$ , other players' actions  $\mathbf{x}_{-i} \in \mathcal{X}_{-i}$ , and the contextual feature  $\mathbf{s}$ . Here,  $\mathcal{X}_{-i} := \prod_{j \in \mathcal{I} \setminus \{i\}} \mathcal{X}_j$ .

A Nash equilibrium  $\mathbf{x}^* := (\mathbf{x}_1^*, \dots, \mathbf{x}_p^*) \in \mathcal{X}$  of the game is defined as a point at which no player can unilaterally increase its utility [Nash, 1950], i.e.,  $U_i(\mathbf{x}_i^*, \mathbf{x}_{-i}^*, \mathbf{s}) \geq U_i(\mathbf{x}_i, \mathbf{x}_{-i}^*, \mathbf{s}), \forall \mathbf{x}_i \in \mathcal{X}_i, i \in \mathcal{I}$ .

We make the following standard assumptions: (i) The feasible set  $\mathcal{X}_i$  is a nonempty, closed, and convex subset of  $\mathbb{R}^{m_i}$  for all  $i \in \mathcal{I}$ ; (ii) Utility function  $U_i$  is continuously differentiable and pseudo-concave with respect to  $\mathbf{x}_i$  for all  $i \in \mathcal{I}$ ; (iii) The contextual feature  $\mathbf{s}$  is publicly known to all players. Under these assumptions, a Nash equilibrium  $\mathbf{x}^* \in \mathcal{X}$  exists in the game  $\mathcal{G}$  [Nash, 1950].

Given a contextual feature  $\hat{\mathbf{s}}^j$  of game  $\mathcal{G}$ , there exists a Nash equilibrium  $\hat{\mathbf{x}}^j$ . Let  $\hat{\mathcal{D}} = \{(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)\}_{j=1}^N$  denote the observable dataset of all equilibrium-context pairs. Our target is to estimate the utility functions of all players from  $\hat{\mathcal{D}}$ .<sup>1</sup>

<sup>1</sup>For example, in a price competition, each firm's utility is known

185 We consider the case where  $U_i(\mathbf{x}_i, \mathbf{x}_{-i}, \mathbf{s}; \boldsymbol{\theta}_i)$  can be pa-  
 186 rameterized by  $\boldsymbol{\theta}_i \in \Theta_i$  for each  $i$ . In practice, the form of  
 187  $U_i(\mathbf{x}_i, \mathbf{x}_{-i}, \mathbf{s}; \boldsymbol{\theta}_i)$  is often known, but  $\boldsymbol{\theta}_i \in \Theta_i$  needs to be  
 188 estimated from data. Let  $\boldsymbol{\vartheta} = (\boldsymbol{\theta}_1; \dots; \boldsymbol{\theta}_p)$  denote the pa-  
 189 rameter vector, i.e., the vertical concatenation of all  $\boldsymbol{\theta}_i$ . Our  
 190 data-driven inverse game problem is as follows.

191 **Problem 1.** Given dataset  $\widehat{\mathcal{D}} = \{(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)\}_{j=1}^N$ , estimate pa-  
 192 rameters  $\boldsymbol{\vartheta}$  that reproduce equilibrium  $\hat{\mathbf{x}}^j$  in context  $\hat{\mathbf{s}}^j$ ,  $\forall j$ .

## 193 4 Incenter-Based Solution

### 194 4.1 Problem Reformulation

195 We begin by exploiting the fact that, under the assumption  
 196 considered,  $\mathbf{x}^* \in \mathcal{X}$  is a Nash equilibrium if and only if it  
 197 solves the *variational inequality* [Harker and Pang, 1990]:

$$\sum_{i=1}^p (-\nabla_{\mathbf{x}_i} U_i(\mathbf{x}^*, \mathbf{s}))^\top (\mathbf{x}_i - \mathbf{x}_i^*) \geq 0, \forall \mathbf{x} \in \mathcal{X}. \quad (1)$$

198 Hence, Problem 1 can be equivalently reformulated as  
 199 seeking a  $\boldsymbol{\vartheta}$  that satisfies the following inequalities for all  
 200  $j \in [N]$  ( $j$  indexes the context-equilibrium pairs):

$$\sum_{i=1}^p (\nabla_{\mathbf{x}_i} U_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j; \boldsymbol{\theta}_i))^\top (\mathbf{x}_i^j - \hat{\mathbf{x}}_i^j) \leq 0, \forall \mathbf{x}^j \in \mathcal{X}^j. \quad (2)$$

201 In the following, we consider a general setting where the  
 202 player action set  $\mathcal{X}^j$  can also change with the contextual fea-  
 203 ture  $\hat{\mathbf{s}}^j$ . Following the existing literature, e.g., [Bertsimas *et*  
 204 *al.*, 2015; Peters *et al.*, 2023; Maddux *et al.*, 2023], our work  
 205 focuses on games where  $U_i(\mathbf{x}, \mathbf{s}; \boldsymbol{\theta}_i)$  can be expressed as a  
 206 linear combination of parameters  $\boldsymbol{\theta}_i$  and functions  $\varphi_i(\mathbf{x}, \mathbf{s})$ ,  
 207 i.e.,  $U_i(\mathbf{x}, \mathbf{s}; \boldsymbol{\theta}_i) = \langle \boldsymbol{\theta}_i, \boldsymbol{\varphi}_i(\mathbf{x}, \mathbf{s}) \rangle$ . As will be shown in  
 208 Section 5.1, many classic games fall into this category, in-  
 209 cluding Bertrand-Nash price competition, aggregative games,  
 210 and traffic games. For ease of presentation, let  $\mathbf{x}_i \in \mathbb{R}$  and  
 211  $\nabla_{\mathbf{x}_i} \varphi_i(\mathbf{x}, \mathbf{s}) = \phi_i(\mathbf{x}, \mathbf{s})$ . Then, we can rewrite (2) as

$$\sum_{i=1}^p (\boldsymbol{\theta}_i^\top \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)) (\mathbf{x}_i^j - \hat{\mathbf{x}}_i^j) \leq 0, \forall \mathbf{x}^j \in \mathcal{X}^j. \quad (3)$$

212 While there may exist (infinitely) many  $\boldsymbol{\vartheta}$  satisfying (3),  
 213 motivated by [Scroccaro *et al.*, 2023], we propose to seek  
 214 the *incenter* of all  $\boldsymbol{\vartheta}$  satisfying (3). As will be explained in  
 215 Section 4.2, this solution is most robust to perturbations of  
 216 the observable dataset  $\widehat{\mathcal{D}}$ .

217 We first formally define the incenter in Section 4.2. We  
 218 then introduce a novel loss function approach to estimate it  
 219 and thus learn  $\boldsymbol{\vartheta}$  in Sections 4.3 and 4.4. In Section 4.5, we  
 220 extend our solution to incorporate prior knowledge about  $\boldsymbol{\vartheta}$ .

### 221 4.2 Incenter Parameter Vector

222 We define the set of parameter vectors consistent with the  
 223 observed dataset  $\widehat{\mathcal{D}}$  as follows.

only to itself, and depends on the prices of all firms and the economic condition. By observing the equilibrium prices under different economic conditions, we estimate the utility functions of all firms.

**Definition 1** (Consistent Parameter Vectors). Given feasible  
 224 action sets  $\{\mathcal{X}^j\}_{j=1}^N$ , the dataset  $\widehat{\mathcal{D}} = \{(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)\}_{j=1}^N$ , and  
 225 functions  $\{\phi_i(\mathbf{x}, \mathbf{s})\}_{i \in \mathcal{I}}$ , the set of consistent parameter vec-  
 226 tors is defined as  
 227

$$\mathbb{C} := \left\{ \boldsymbol{\vartheta} : \langle \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \leq 0, \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N] \right\}. \quad (4)$$

Here,  $\boldsymbol{\Phi}_{\boldsymbol{\vartheta}}^j = (\boldsymbol{\theta}_1^\top \phi_1(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j), \dots, \boldsymbol{\theta}_p^\top \phi_p(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j))^\top \in \mathbb{R}^p$ . 228

In other words,  $\mathbb{C}$  is the set of  $\boldsymbol{\vartheta}$  satisfying (3). We focus on  
 229 searching for the incenter of  $\mathbb{C}$ , which is defined as follows. 230

**Definition 2** (Incenter of  $\mathbb{C}$ ). Given a nonempty set  $\mathbb{C}$ , its  
 231 incenter  $\boldsymbol{\vartheta}^{\text{in}}$  is defined as  
 232

$$\boldsymbol{\vartheta}^{\text{in}} \in \arg \max_{\boldsymbol{\vartheta} \in \mathbb{C}} \min_{\tilde{\boldsymbol{\vartheta}} \in \overline{\text{int}(\mathbb{C})}} a(\boldsymbol{\vartheta}, \tilde{\boldsymbol{\vartheta}}). \quad (5)$$

Here,  $\overline{\text{int}(\mathbb{C})}$  is the region excluding the interior of  $\mathbb{C}$ , and  
 233  $a(\boldsymbol{\vartheta}, \tilde{\boldsymbol{\vartheta}})$  is the angle between  $\boldsymbol{\vartheta}$  and  $\tilde{\boldsymbol{\vartheta}}$ , i.e., 234

$$a(\boldsymbol{\vartheta}, \tilde{\boldsymbol{\vartheta}}) = \arccos \left( \frac{\langle \boldsymbol{\vartheta}, \tilde{\boldsymbol{\vartheta}} \rangle}{\|\boldsymbol{\vartheta}\|_2 \|\tilde{\boldsymbol{\vartheta}}\|_2} \right).$$

Geometrically, an incenter  $\boldsymbol{\vartheta}^{\text{in}}$  of  $\mathbb{C}$  can be viewed as a vec-  
 235 tor furthest away from the boundary of  $\mathbb{C}$ , as measured by the  
 236 angle. Since each facet of  $\mathbb{C}$  is determined by a  $(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  pair  
 237 in dataset  $\widehat{\mathcal{D}}$ , incenter  $\boldsymbol{\vartheta}^{\text{in}}$  can be informally interpreted as  
 238 the parameter vector that is most robust to perturbations of  
 239 data  $\widehat{\mathcal{D}}$  (i.e., perturbations of the facets of  $\mathbb{C}$ ). For a formal  
 240 description of this property, see the Supplementary Material.

For simplicity, we define a column vector  $\boldsymbol{\phi}^j :=$   
 241  $(\phi_1(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j); \dots; \phi_p(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j))$ . In the following theorem,  
 242 we formulate the problem of finding an incenter  $\boldsymbol{\vartheta}^{\text{in}}$ . 243

**Theorem 1** (Incenter Computation). When  $\mathbb{C}$  is defined as  
 244 (4) and its interior is nonempty, finding  $\boldsymbol{\vartheta}^{\text{in}}$  that satisfies (5)  
 245 is equivalent to solving the following problem: 246

$$\begin{aligned} \min_{\boldsymbol{\vartheta}} \quad & \|\boldsymbol{\vartheta}\|_2 \\ \text{s.t.} \quad & \langle \boldsymbol{\Phi}_{\boldsymbol{\vartheta}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + \left\| \boldsymbol{\phi}^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2 \leq 0, \\ & \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N]. \end{aligned} \quad (6)$$

In (6),  $\odot$  is the element-wise multiplication,  $\text{vec}(\cdot)$  means  
 247 stacking all elements into a column vector, and  $\otimes$  denotes the  
 248 Kronecker product.  $\mathbb{1}_n$  is an  $n$ -dimensional all-one vector,  
 249 where  $n$  is the dimension of  $\phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$ . Proofs are provided  
 250 in the Supplementary Material. 251

**Remark:** The formulation in (6) converts the problem of  
 252 finding  $\boldsymbol{\vartheta}^{\text{in}}$  that satisfies (5) to a convex optimization problem.  
 253 Each  $\mathbf{x}^j \in \mathcal{X}^j$  introduces a constraint to (6). In many games,  
 254 the cardinality of  $\mathcal{X}^j$  is very large or infinite. In this case, it is  
 255 intractable to enumerate all elements of  $\mathcal{X}^j$  for all  $j \in [N]$  to  
 256 check if  $\boldsymbol{\vartheta}$  meets the inequality constraints. 257

### 258 4.3 Loss Function Design

We propose a loss function-based method to tackle the case  
 259 where  $|\mathcal{X}^j|$  is large or infinite. Interestingly, we notice that  
 260 261

262 this approach also applies when there may be no  $\vartheta$  consis-  
 263 tent with  $\widehat{\mathcal{D}}$ , i.e.,  $\mathbb{C}/\{\mathbf{0}\} = \emptyset$ . This scenario is common, and  
 264 arises if players are bounded-rational and may only choose  
 265 actions close to the optimum, leading to an  $\epsilon$ -Nash equilib-  
 266 rium. Towards obtaining a loss function, we first relax the  
 267 constraints of (6) by introducing *slack variables*  $\beta_1, \dots, \beta_N$ :

$$\begin{aligned} \min_{\vartheta, \beta} \quad & \frac{1}{N} \sum_{j=1}^N \beta_j + \|\vartheta\|_2 \\ \text{s.t.} \quad & \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + \left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2 \leq \beta_j, \\ & \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N]. \end{aligned} \quad (7)$$

268 Then, we define the following loss function.

269 **Definition 3** (Loss Function). *Given a context-action pair*  
 270 *( $\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j$ ), the loss  $\ell_{\vartheta}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  of a parameter vector  $\vartheta$  is de-*  
 271 *fined as*

$$\max_{\mathbf{x}^j \in \mathcal{X}^j} \left\{ \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + \left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2 \right\}.$$

272 Using the loss function, we reformulate problem (7) to the  
 273 following regularized empirical loss minimization problem:

$$\min_{\vartheta} \frac{1}{N} \sum_{j=1}^N \ell_{\vartheta}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) + \alpha \mathcal{R}(\vartheta). \quad (8)$$

274 Here, we replace  $\|\vartheta\|_2$  in the objective with a general regular-  
 275 ization function  $\mathcal{R}(\cdot)$  and use  $\alpha \geq 0$  as a hyperparameter.

276 **Remark 1:** In (8), we use the loss function to capture  
 277 the hard inequality constraints in (7). Since (8) is an un-  
 278 constrained optimization, we can solve it using a first-order  
 279 algorithm in Section 4.4 even when  $|\mathcal{X}_j|$  is infinite.

280 **Remark 2:** If no  $\vartheta$  is consistent with  $\widehat{\mathcal{D}}$ , problem (6) has  
 281 no feasible solution satisfying all the hard constraints. In this  
 282 case, our loss function can still guide the search towards a  
 283 vector that minimizes the degree of constraint violation.

#### 284 4.4 Estimation Algorithm

285 In this section, we introduce a first-order algorithm for solv-  
 286 ing (8). We apply the *mirror descent method* [Beck and  
 287 Teboulle, 2003], which enables the learning of  $\vartheta$  in both Eu-  
 288 clidean and non-Euclidean geometries. Specifically, at each  
 289 iteration  $t$ , the mirror descent update is give by

$$\vartheta_{t+1} = \arg \min_{\vartheta} \{ \eta_t \langle \mathbf{g}_t(\vartheta_t), \vartheta \rangle + \mathcal{B}_{\omega}(\vartheta, \vartheta_t) \}. \quad (9)$$

290 Here,  $\eta_t > 0$  is the step-size,  $\mathbf{g}_t(\vartheta_t)$  is the subgradient of the  
 291 complete loss function in (8), and function  $\mathcal{B}_{\omega}$  is the Bregman  
 292 divergence w.r.t.  $\omega : \Theta \rightarrow \mathbb{R}$  [Bubeck and others, 2015], i.e.,

$$\mathcal{B}_{\omega}(\vartheta, \vartheta_t) = \omega(\vartheta) - \omega(\vartheta_t) - \langle \nabla \omega(\vartheta_t), \vartheta - \vartheta_t \rangle.$$

293 In particular, when  $\omega(\vartheta) = \frac{1}{2} \|\vartheta\|_2^2$ , (9) reduces to  $\vartheta_{t+1} =$   
 294  $\vartheta_t - \eta_t \mathbf{g}_t(\vartheta_t)$ , i.e., the subgradient descent update.

295 To implement the mirror descent method, we first de-  
 296 rive subgradient  $\mathbf{g}_t(\vartheta_t)$  for the loss function in (8). Let  
 297  $h(\mathbf{x}^j, \hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) := \left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2$ . Us-  
 298 ing Danskin's Theorem [Bertsekas, 2016], we compute  
 299  $\partial \ell_{\vartheta}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  (the subdifferential of  $\ell_{\vartheta}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  w.r.t.  $\vartheta$ ) as

$$\text{conv} \left\{ \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \mid \mathbf{x}^j \in \mathcal{X}^j(\vartheta) \right\}, \quad (10)$$

---

#### Algorithm 1 Mirror Descent for Problem (8)

---

```

1: function MIRROR DESCENT( $\widehat{\mathcal{D}}, \omega, \mathcal{R}, \{\phi_i\}, \alpha, \{\eta_t\}, \vartheta_0$ )
2:   for  $t \in \{0, \dots, T-1\}$  do ▷ Iteration of Estimation
3:     for  $j \in \{1, \dots, N\}$  do ▷ Loop over  $\widehat{\mathcal{D}}$ 
4:       Compute  $\tilde{\mathbf{x}}_t^j$  according to (11);
5:       Compute  $\partial \ell_{\vartheta_t}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  according to (10);
6:     end for
7:     Compute subgradient  $\mathbf{g}_t(\vartheta_t)$  according to (12);
8:     Apply mirror descent updates according to (9);
9:   end for
10:  return  $\{\vartheta_t\}_{t=1}^T$ .
11: end function

```

---

Here,  $\text{conv}\{\cdot\}$  represents the convex hull of the given set, and  
 set  $\mathcal{X}^j(\vartheta) = \{\tilde{\mathbf{x}}^j\}$ , where

$$\tilde{\mathbf{x}}^j = \arg \max_{\mathbf{x} \in \mathcal{X}^j} \left\{ \langle \Phi_{\vartheta}^j, \mathbf{x} - \hat{\mathbf{x}}^j \rangle + h(\mathbf{x}, \hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) \right\}. \quad (11)$$

Based on (10), the subgradient  $\mathbf{g}_t(\vartheta_t)$  is derived as

$$\mathbf{g}_t(\vartheta_t) = \frac{1}{N} \sum_{j=1}^N \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\tilde{\mathbf{x}}_t^j - \hat{\mathbf{x}}^j)) + \alpha \nabla \mathcal{R}(\vartheta_t). \quad (12)$$

Algorithm 1 presents the pseudocode of the mirror descent  
 algorithm for solving (8). In line 4 of Algorithm 1, we com-  
 pute  $\tilde{\mathbf{x}}_t^j$  for each data point  $(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  by solving a maximiza-  
 tion problem. In line 5, we use the obtained  $\tilde{\mathbf{x}}_t^j$  to compute  
 the subdifferential of  $\ell_{\vartheta_t}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  based on (10). In lines 7  
 to 8, we utilize the entire dataset to compute the subgradient  
 $\mathbf{g}_t(\vartheta_t)$  of the objective function in (8), and then update cur-  
 rent parameters to obtain  $\vartheta_{t+1}$  by solving (9). Due to space  
 limitations, we provide a detailed convergence analysis of Al-  
 gorithm 1 in our Supplementary Material.

#### 4.5 Incorporation of Priors

In this subsection, we extend our estimation method to in-  
 corporate a specific type of prior knowledge about  $\vartheta$ . We  
 define  $F_i(\mathbf{x}; \boldsymbol{\theta}_i) := -\partial U_i(\mathbf{x}, \mathbf{s}; \boldsymbol{\theta}_i) / \partial x_i$  and let  $\mathbf{F}(\mathbf{x}; \boldsymbol{\theta}) :=$   
 $(F_1(\mathbf{x}; \boldsymbol{\theta}_1); \dots; F_p(\mathbf{x}; \boldsymbol{\theta}_p))$ . In many games [LeBlanc *et al.*,  
 1975; Jensen, 2018],  $\mathbf{F}(\mathbf{x}; \boldsymbol{\theta})$  is known to be monotone, i.e.,  
 to satisfy

$$(\mathbf{x} - \mathbf{x}')^\top (\mathbf{F}(\mathbf{x}; \boldsymbol{\theta}) - \mathbf{F}(\mathbf{x}'; \boldsymbol{\theta})) \geq 0, \forall \mathbf{x}, \mathbf{x}' \in \mathcal{X}. \quad (13)$$

Such monotonicity property is closely related to the equilib-  
 rium solution in (1). For instance, if  $\mathbf{F}(\mathbf{x}; \boldsymbol{\theta})$  is strictly mono-  
 tone, there exists at most one Nash equilibrium [Harker and  
 Pang, 1990]. Moreover, the projection gradient methods for  
 solving (1) exhibit global convergence under this condition  
 [Fukushima, 1992; Korpelevich, 1976; Xiu and Zhang, 2003;  
 Bnouhachem *et al.*, 2015].

We then aim at estimating  $\vartheta$  while ensuring that  $\mathbf{F}(\mathbf{x}; \boldsymbol{\theta})$   
 remains monotone. Towards this goal, we build upon the  
 fact that  $\mathbf{F}(\mathbf{x}; \boldsymbol{\theta})$  is monotone if and only if its Jacobian ma-  
 trix is positive-semidefinite [Facchinei and Pang, 2003], i.e.,  
 $\mathbf{d}^\top \nabla \mathbf{F}(\mathbf{x}) \mathbf{d} \geq 0, \forall \mathbf{x} \in \mathcal{X}, \mathbf{d} \in \mathbb{R}^p$ . Based on this observa-  
 tion, we incorporate this condition in (8) by using *semidefinite*  
*program* [Vandenberghe and Boyd, 1996].

334 The specific form of  $F(\mathbf{x}; \boldsymbol{\theta})$  can vary across differ-  
 335 ent game applications, making a unified formulation of  
 336 the semidefinite program challenging. Hence, we take the  
 337 Bertrand-Nash competition as an example to introduce our  
 338 solution. It is a classic game widely studied in literature  
 339 [Berry, 1994; Berry *et al.*, 1995; Bertsimas *et al.*, 2015]. We  
 340 include the formulations of Cournot and traffic games in Sup-  
 341plementary Material.

342 For ease of presentation, we consider a two-firm Bertrand  
 343 competition (i.e.,  $p = 2$ ). The analysis of more than two firms  
 344 is provided in Supplementary Material. Each firm  $i \in \{1, 2\}$   
 345 chooses price  $x_i$  to maximize revenue [Bertsimas *et al.*, 2015;  
 346 Maddux *et al.*, 2023]:

$$U_i(\mathbf{x}, s; \boldsymbol{\theta}_i) = x_i(\theta_{i,1}x_1 + \theta_{i,2}x_2 + \theta_{i,3}s + \theta_{i,4}), \quad (14)$$

347 where the terms in the parentheses correspond to the demand  
 348 of firm  $i$ . In this case,  $\boldsymbol{\vartheta} = (\boldsymbol{\theta}_1; \boldsymbol{\theta}_2) \in \mathbb{R}^8$ .

349 In addition to the monotonicity property, there are some  
 350 other common assumptions on  $\boldsymbol{\vartheta}$  in this competition, e.g.,  
 351  $U_i$  is concave w.r.t.  $x_i$ , and  $\theta_{21}, \theta_{12}$  can be normalized to  
 352 1 without loss of generality [Maddux *et al.*, 2023]. Next, we  
 353 reformulate (8) to consider all these priors and assumptions.

354 **Lemma 1.** Let  $\boldsymbol{\sqsupset} := -\begin{bmatrix} \theta_{11} & \theta_{21}/2 \\ \theta_{12}/2 & \theta_{22} \end{bmatrix}$ ,  $\tilde{\boldsymbol{\sqsupset}} := \begin{bmatrix} \theta_{13} & \theta_{23} \\ \theta_{14} & \theta_{24} \end{bmatrix}$ .  
 355 For a two-firm Bertrand competition, formulation (8) can be  
 356 extended to the following problem:

$$\begin{aligned} \min_{\boldsymbol{\sqsupset}, \tilde{\boldsymbol{\sqsupset}}} & \frac{1}{N} \sum_{j=1}^N \ell_{\boldsymbol{\sqsupset}, \tilde{\boldsymbol{\sqsupset}}}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) + \alpha \mathcal{R}(\boldsymbol{\sqsupset}, \tilde{\boldsymbol{\sqsupset}}) \\ \text{s.t.} & \text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \boldsymbol{\sqsupset}) \geq 0, \forall i = 1, 2, \\ & \text{Tr}(\mathbf{A} \boldsymbol{\sqsupset}) = -1, \boldsymbol{\sqsupset} \geq 0, \tilde{\boldsymbol{\sqsupset}} \geq 0. \end{aligned} \quad (15)$$

357 Here,  $\mathbf{A}$  equals  $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ , and the vector  $\mathbf{e}_i$  has a one in the  
 358  $i$ -th position and zeros elsewhere.  $\text{Tr}(\cdot)$  is the trace operator.  
 359 Constraint  $\text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \boldsymbol{\sqsupset}) \geq 0, \forall i = 1, 2$  implies  $\theta_{11}, \theta_{22} \leq$   
 360 0, ensuring the concavity of each  $U_i$ .  $\text{Tr}(\mathbf{A} \boldsymbol{\sqsupset}) = -1$  indi-  
 361 cates  $\theta_{21} = \theta_{12} = 1$ .  $\boldsymbol{\sqsupset}, \tilde{\boldsymbol{\sqsupset}} \geq 0$  means that both  $\boldsymbol{\sqsupset}$  and  $\tilde{\boldsymbol{\sqsupset}}$  are  
 362 positive semidefinite. Loss function  $\ell_{\boldsymbol{\sqsupset}, \tilde{\boldsymbol{\sqsupset}}}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  is

$$\max_{\mathbf{x}^j \in \mathcal{X}^j} \left\{ -\text{Tr}(\Psi_s^j \boldsymbol{\sqsupset}) + \text{Tr}(\tilde{\Psi}_s^j \tilde{\boldsymbol{\sqsupset}}) + h(\mathbf{x}^j, \hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) \right\}, \quad (16)$$

363 where  $\Psi_s^j := \begin{bmatrix} v_1^j & v_2^j + v_5^j \\ v_2^j + v_5^j & v_6^j \end{bmatrix}$ ,  $\tilde{\Psi}_s^j := \begin{bmatrix} v_3^j & v_4^j + v_7^j \\ v_4^j + v_7^j & v_8^j \end{bmatrix}$ ,  
 364 and  $\mathbf{v}^j := \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \in \mathbb{R}^8$ .

365 To solve the convex problem (15), we apply the *primal-*  
 366 *dual interior-point method* [Boyd and Vandenberghe, 2004].  
 367 The primal barrier method augments the primal objective  
 368 with a barrier function to handle inequality constraints, pen-  
 369 nalizing solutions near the boundary of the feasible set. The  
 370 primal-dual interior-point method extends this concept to  
 371 both the primal and dual problems, solving them simultane-  
 372 ously. It is often more efficient than the barrier method.

373 To use the primal-dual interior-point method, we derive the  
 374 perturbed KKT conditions based on the logarithmic barrier  
 375 [Nesterov and Nemirovskii, 1994] as follows.

---

## Algorithm 2 Primal-Dual Interior-Point for Problem (15)

---

```

1: function PD-IP( $\hat{\mathcal{D}}, \alpha, \epsilon, \boldsymbol{\sqsupset}^0, \tilde{\boldsymbol{\sqsupset}}^0, \Xi^0, \tilde{\Xi}^0, \boldsymbol{\lambda}^0, \nu^0$ ).
2:    $\frac{1}{\mu^0} \leftarrow \frac{\text{Tr}(\boldsymbol{\sqsupset}^0 \Xi^0) + \text{Tr}(\tilde{\boldsymbol{\sqsupset}}^0 \tilde{\Xi}^0)}{4p} + \frac{\sum_{i=1}^p \lambda_i^0 \text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \boldsymbol{\sqsupset}^0)}{4}$ ;
3:    $k \leftarrow 0$ .
4:   while  $\frac{1}{\mu^k} > \epsilon$  do
5:     Compute  $\Psi_s^j(\bar{\mathbf{x}}_k^j), \tilde{\Psi}_s^j(\bar{\mathbf{x}}_k^j)$  by  $\hat{\mathcal{D}}, \boldsymbol{\sqsupset}^k$  and  $\tilde{\boldsymbol{\sqsupset}}^k$ ;
6:     Compute  $(\Delta \boldsymbol{\sqsupset}^k, \Delta \tilde{\boldsymbol{\sqsupset}}^k, \Delta \Xi^k, \Delta \tilde{\Xi}^k, \Delta \boldsymbol{\lambda}^k, \Delta \nu^k)$ ;
7:     Backtracking line search for step-sizes  $\eta_p^k, \eta_d^k$ ;
8:     Compute  $(\boldsymbol{\sqsupset}^{k+1}, \tilde{\boldsymbol{\sqsupset}}^{k+1}, \Xi^{k+1}, \tilde{\Xi}^{k+1}, \boldsymbol{\lambda}^{k+1}, \nu^{k+1})$ ;
9:     Compute  $\frac{1}{\mu^{k+1}}$ ;
10:     $k \leftarrow k + 1$ ;
11:  end while
12:  return  $(\boldsymbol{\sqsupset}^k, \tilde{\boldsymbol{\sqsupset}}^k, \Xi^k, \tilde{\Xi}^k, \boldsymbol{\lambda}^k, \nu^k)$ .
13: end function

```

---

**Proposition 1.** Let  $(\boldsymbol{\sqsupset}^*, \tilde{\boldsymbol{\sqsupset}}^*)$  and  $(\Xi^*, \tilde{\Xi}^*, \boldsymbol{\lambda}^*, \nu^*)$  be the op- 376  
 timal primal and dual solutions, respectively, and  $\mu$  be the 377  
 barrier parameter. Then, the optimality conditions for the 378  
 logarithmic barrier centering problem are 379

$$\begin{aligned} \boldsymbol{\sqsupset}^*, \tilde{\boldsymbol{\sqsupset}}^* &\geq 0; \quad \Xi^*, \tilde{\Xi}^* \geq 0; \quad \lambda_i^* \geq 0, \quad \text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \boldsymbol{\sqsupset}^*) \geq 0, \forall i; \\ \Xi^* \boldsymbol{\sqsupset}^* &= \frac{1}{\mu} \mathbf{I}; \quad \tilde{\Xi}^* \tilde{\boldsymbol{\sqsupset}}^* = \frac{1}{\mu} \mathbf{I}; \quad \lambda_i^* \text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \boldsymbol{\sqsupset}^*) - \frac{1}{\mu} = 0, \forall i; \\ \text{Tr}(\mathbf{A} \boldsymbol{\sqsupset}^*) + 1 &= 0; \quad \alpha \tilde{\boldsymbol{\sqsupset}}^* - \tilde{\Xi}^* + \frac{1}{N} \sum_{j=1}^N \tilde{\Psi}_s^j(\bar{\mathbf{x}}^j) = \mathbf{0}; \\ \alpha \boldsymbol{\sqsupset}^* - \sum_{i=1}^2 \lambda_i^* (\mathbf{e}_i \mathbf{e}_i^\top) - \Xi^* + \nu^* \mathbf{A} - \frac{1}{N} \sum_{j=1}^N \Psi_s^j(\bar{\mathbf{x}}^j) &= \mathbf{0}, \end{aligned}$$

380 where  $\mathbf{I}$  is the identity matrix, and  $\bar{\mathbf{x}}^j$  is defined as 381  
 $\arg \max_{\mathbf{x}^j \in \mathcal{X}^j} \left\{ -\text{Tr}(\Psi_s^j \boldsymbol{\sqsupset}^*) + \text{Tr}(\tilde{\Psi}_s^j \tilde{\boldsymbol{\sqsupset}}^*) + h(\mathbf{x}^j, \hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) \right\}$ .

382 We use Newton's method [Alizadeh *et al.*, 1998] to solve 383  
 the equations in Proposition 1. Due to space limit, we leave 384  
 all details to our Supplementary Material.

385 Algorithm 2 shows the primal-dual interior-point method 386  
 for solving (15). In line 5, we use current parameters  $\boldsymbol{\sqsupset}^k$  and 387  
 $\tilde{\boldsymbol{\sqsupset}}^k$  to compute  $\bar{\mathbf{x}}_k^j$  for each data point  $(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$ , and then ob- 388  
 tain  $\Psi_s^j(\bar{\mathbf{x}}_k^j)$  and  $\tilde{\Psi}_s^j(\bar{\mathbf{x}}_k^j)$ . In line 6, we compute the Newton 389  
 updates  $(\Delta \boldsymbol{\sqsupset}^k, \Delta \tilde{\boldsymbol{\sqsupset}}^k, \Delta \Xi^k, \Delta \tilde{\Xi}^k, \Delta \boldsymbol{\lambda}^k, \Delta \nu^k)$ . In lines 7, we utilize 390  
 the backtracking line search to compute current step sizes  $\eta_p^k$  391  
 and  $\eta_d^k$  for the primal and dual variables, respectively. In lines 392  
 8 to 9, we update the current primal and dual variables, and 393  
 compute barrier parameter  $\mu^{k+1}$  in the next iteration.

## 5 Numerical Experiments 394

### 5.1 Games Applications 395

**Demand Estimation in Bertrand Competition** The two- 396  
 firm Bertrand competition was described in Section 4.5. 397  
 The demand function estimation problem is: *Given data* 398  
 $\{(\hat{\mathbf{x}}_1^j, \hat{\mathbf{x}}_2^j, \hat{\mathbf{s}}^j)\}_{j=1}^N$ , we seek to estimate  $\boldsymbol{\vartheta} = (\boldsymbol{\theta}_1; \boldsymbol{\theta}_2) \in \mathbb{R}^8$  399  
 such that each  $(\hat{\mathbf{x}}_1^j, \hat{\mathbf{x}}_2^j)$  is a Nash equilibrium for context  $\hat{\mathbf{s}}^j$ . 400

401 **Profit Estimation in Aggregative Games** In aggregative  
 402 games [Jensen, 2018], each player  $i$ 's utility  $U_i$  depends on  
 403 its own action  $x_i$  and the aggregation of all players' actions,  
 404 i.e.,  $U_i(\mathbf{x}) = \tilde{U}_i(\mathbf{x}_i, \sum_{k=1}^p x_k)$ . We take the Cournot com-  
 405 petition as an example. Each firm  $i$  decides the quantity  $x_i$  of  
 406 a homogeneous product to supply, aiming to maximize:

$$U_i(\mathbf{x}, s; a, b, d, c_i) = x_i \left( b + ds - a \sum_{k=1}^p x_k \right) - c_i x_i. \quad (17)$$

407 The terms in the parentheses represent the inverse demand  
 408 function, and  $c_i$  is the unit cost of production. Following  
 409 [Risanger *et al.*, 2020], we include  $s$  as a demand shock that  
 410 adjusts the demand intercept  $b$ . The problem is as follows:  
 411 Given  $\{(\hat{\mathbf{x}}^j, \hat{s}^j)\}_{j=1}^N$ , we seek  $\vartheta = (a, b, d, c_1, \dots, c_p) \in$   
 412  $\mathbb{R}^{p+3}$  such that each  $\hat{\mathbf{x}}^j$  constitutes a Nash equilibrium.

413 **Cost Estimation in Traffic Game** We represent a road net-  
 414 work by a tuple  $(\mathcal{V}, \mathcal{A}, \mathcal{W}, t(\cdot))$ , where  $\mathcal{V}$  and  $\mathcal{A}$  denote the  
 415 sets of nodes and links, respectively.  $\mathcal{W}$  is the set of all  
 416 Origin-Destination (OD) pairs.  $t(\cdot)$  denotes the cost func-  
 417 tions, and its  $a$ -th element  $t_a(\cdot)$  is the travel latency cost func-  
 418 tion for link  $a \in \mathcal{A}$ . A common choice for  $t_a(\cdot)$  is the U.S.  
 419 Bureau of Public Roads (BPR) [Sheffi, 1985] function:

$$t_a(x_a) = \theta_a^0 + \theta_a^1 \left( \frac{x_a}{C_a} \right)^\gamma. \quad (18)$$

420 Here,  $x_a$  is the overall traffic on link  $a$ ,  $C_a$  is the capacity of  
 421 link  $a$ ,  $\gamma$  is the congestion sensitivity parameter, and  $\theta_a^0, \theta_a^1$   
 422 are cost parameters. In these settings it is standard to assume  
 423  $C_a$  and  $\gamma$  are common knowledge, while only  $\theta_a^0, \theta_a^1$  need  
 424 to be estimated. The cost estimation problem is: Given link  
 425 flow data  $\{\hat{\mathbf{x}}^j = (\hat{x}_a^j; a \in \mathcal{A})\}_{j=1}^N$ , we seek to estimate cost  
 426 parameters  $\{\theta_a = (\theta_a^0, \theta_a^1), a \in \mathcal{A}\}$ , such that each  $\hat{\mathbf{x}}^j$  is a  
 427 Wardrop equilibrium [Sheffi, 1985; Patriksson, 2015].

## 428 5.2 Experimental Settings

429 **Experimental Data** We describe the dataset as follows.

- 430 • **Bertrand Competition:** Following [Maddux *et al.*, 2023],  
 431 we generate  $\vartheta$  by randomly sampling its elements from  
 432 Gaussian distributions:  $\theta_{11} \sim \mathcal{N}(-1.2, 0.5^2)$ ,  $\theta_{12} \sim$   
 433  $\mathcal{N}(0.5, 0.1^2)$ ,  $\theta_{21} \sim \mathcal{N}(0.3, 0.1^2)$ ,  $\theta_{22} \sim \mathcal{N}(-1, 0.5^2)$ ,  
 434 and  $\theta_{i3}, \theta_{i4} \sim \mathcal{N}(1, 0.5^2)$  for  $i = 1, 2$ . We take  $s$  to be i.i.d.  
 435 samples from  $\mathcal{N}(5, 1.5^2)$ . Given each  $\hat{s}^j$ , we solve for the  
 436 equilibrium prices  $(\hat{x}_1^j, \hat{x}_2^j)$  using first-order methods. To  
 437 evaluate different estimation methods, we generate 50 random  
 438  $\vartheta$ . For each  $\vartheta$ , we create a training dataset  $\hat{\mathcal{D}}_{\text{train}}$  and  
 439 a test dataset  $\hat{\mathcal{D}}_{\text{test}}$ , both with a size of 500.
- 440 • **Cournot Competition:** We consider  $p = 3$  players, and  
 441 generate  $\vartheta$  by randomly sampling its elements as follows:  
 442  $a, d \sim \mathcal{U}(5, 10)$ ,  $b \sim \mathcal{N}(50, 5^2)$ ,  $c_i \sim \mathcal{U}(10, 20)$ . Given  
 443 each context  $\hat{s}^j \sim \mathcal{N}(6, 2^2)$ , we solve for the Nash equilib-  
 444 rium  $\hat{\mathbf{x}}^j$  using first-order methods. We randomly generate  
 445 50 different  $\vartheta$  for evaluation. Each  $\vartheta$  has a corresponding  
 446  $\hat{\mathcal{D}}_{\text{train}}$  and  $\hat{\mathcal{D}}_{\text{test}}$ , both with 500 samples.
- 447 • **Traffic Game:** We consider the Sioux Falls network  
 448 [LeBlanc *et al.*, 1975], which contains 24 nodes and 76  
 449 links. Every two nodes in the network constitute OD pairs.

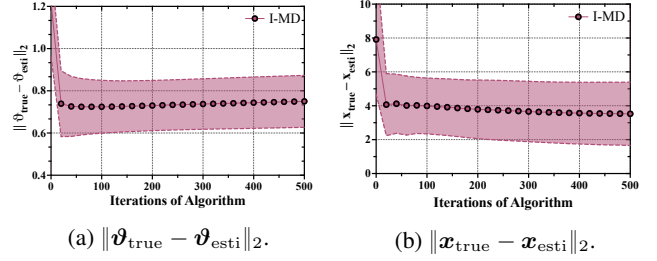


Figure 2: Convergence of **I-MD** on the Bertrand Dataset.

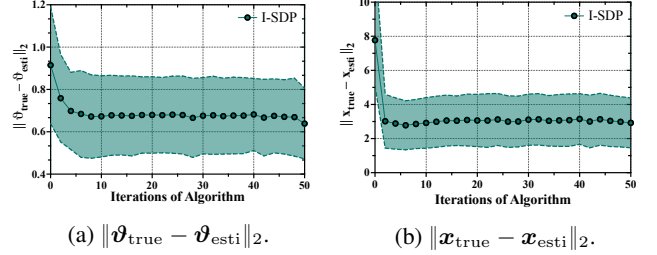


Figure 3: Convergence of **I-SDP** on the Bertrand Dataset.

450 We set  $\vartheta$  according to this real network. To generate multi-  
 451 ple equilibrium data, following [Bertsimas *et al.*, 2015], we  
 452 randomly perturb the OD demands 1000 times, with each  
 453 perturbation drawn from  $\mathcal{U}(0, 0.1)$ . Using Frank-Wolfe al-  
 454 gorithm and true  $\vartheta$ , we then compute  $\hat{x}_a$  for each perturbed  
 455 OD demand.  $\hat{\mathcal{D}}_{\text{train}}$  and  $\hat{\mathcal{D}}_{\text{test}}$  both contain 500 samples.

456 **Comparison Methods** We compare the performance of the  
 457 following five estimation methods:

- 458 • **I-MD (Incenter with Mirror Descent):** We estimate  $\vartheta$  using  
 459 Algorithm 1.
- 460 • **I-SDP (Incenter with SemiDefinite Programming):** We use  
 461 Algorithm 2, which incorporates the monotonicity of  $F$   
 462 (versions of the algorithm tailored for the Cournot compe-  
 463 tition and traffic game are in Supplementary Material).
- 464 • **Bertsimas (Data-Driven Estimation in Equilibrium Using**  
 465 **Inverse Optimization [Bertsimas *et al.*, 2015]):** It estimates  
 466  $\vartheta$  from observed equilibria based on inverse optimization  
 467 without using the concept of incenter.
- 468 • **Feasibility:** It chooses a parameter vector from the set  $\mathcal{C}$   
 469 of consistent parameter vectors. To check the feasibility of  
 470 the parameter vectors w.r.t. (3), it discretizes each  $\mathcal{X}^j$ .
- 471 • **Random:** It randomly chooses a parameter vector.

472 All implementation details, datasets, and codes are in Supple-  
 473 mentary Material, and will be publicly available.

## 474 5.3 Experimental Results

475 We evaluate the estimation errors of the methods using two  
 476 metrics: (i) The  $\ell_2$ -distance between the estimated param-  
 477 eter vector  $\vartheta_{\text{esti}}$  and the true parameter vector  $\vartheta_{\text{true}}$ , i.e.,  
 478  $\|\vartheta_{\text{true}} - \vartheta_{\text{esti}}\|_2$ . To ensure consistent comparisons, all pa-  
 479 rameter vectors are normalized before evaluation; (ii) The  $\ell_2$ -  
 480 distance between the equilibrium  $\mathbf{x}_{\text{esti}}$  computed using  $\vartheta_{\text{esti}}$   
 481 and the equilibrium  $\mathbf{x}_{\text{true}}$  computed using the true parameter  
 482 vector  $\vartheta_{\text{true}}$ , i.e.,  $\|\mathbf{x}_{\text{true}} - \mathbf{x}_{\text{esti}}\|_2$ .

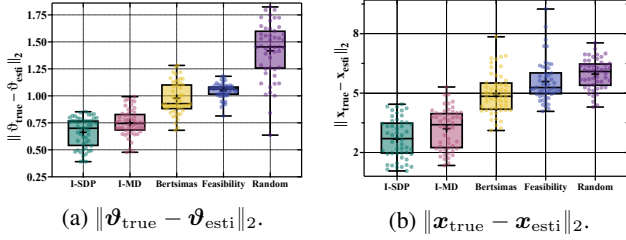


Figure 4: Comparison Results on Bertrand Testing Data.

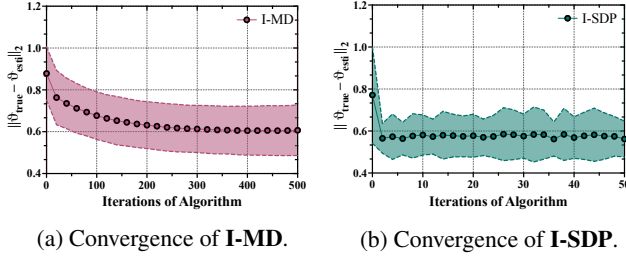


Figure 5: Convergence Results on the Cournot Dataset.

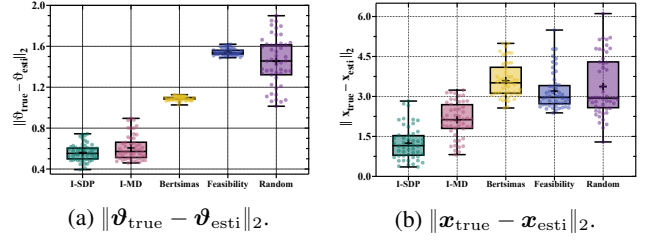


Figure 6: Comparison Results on Cournot Testing Data.

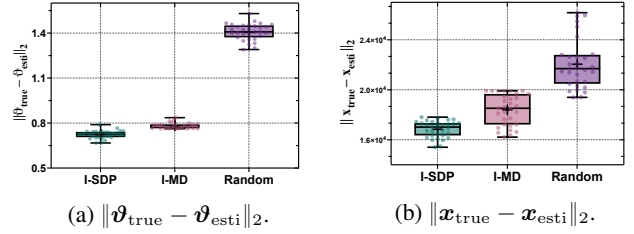


Figure 7: Comparison Results on Traffic Testing Data.

483 **Bertrand Competition** In Figure 2, we depict the conver- 518  
 484 gence of our **I-MD** method on the Bertrand data. Figure 2a 519  
 485 illustrates the average error between  $\vartheta_{\text{esti}}$  and  $\vartheta_{\text{true}}$  over 50 520  
 486 experiments (each with a different  $\vartheta_{\text{true}}$ ). The shaded area 521  
 487 represents the standard deviation of these errors. Figure 2b 522  
 488 shows the average error between the true equilibrium  $x_{\text{true}}$  523  
 489 and the equilibrium  $x_{\text{esti}}$  computed using  $\vartheta_{\text{esti}}$  on the *testing* 524  
 490 *data*. The convergence of our **I-SDP** method is depicted in  
 491 Figure 3. Our **I-SDP** converges to lower errors in both met-  
 492 rics compared to **I-MD**. This is attributed to the incorpora-  
 493 tion of the monotonicity property of  $F$ . It can also be observed  
 494 that **I-SDP** converges much faster, thanks to the efficiency of  
 495 the primal-dual interior-point method.

496 Figure 4 presents the comparative results of different esti-  
 497 mation methods on the testing data, evaluated using two met-  
 498 rics. We use boxplots to illustrate the distribution of two error  
 499 metrics across 50 trials for each method. Each box includes  
 500 all error points, with the central line representing the median  
 501 and the “+” sign denoting the mean error. Our **I-SDP** method  
 502 achieves the lowest median and mean errors with minimal  
 503 variability across both metrics. Specifically, for the met-  
 504 ric  $\|\vartheta_{\text{esti}} - \vartheta_{\text{true}}\|_2$ , **I-SDP** achieves the lowest mean error  
 505 (around 0.663), whereas the comparison methods exhibit  
 506 relatively large errors (with the best among them being 0.979).  
 507 Similarly, for  $\|x_{\text{esti}} - x_{\text{true}}\|_2$ , **I-SDP** again performs best,  
 508 with a mean error around 2.690. Notably, our **I-MD** method  
 509 outperforms all comparison methods across two metrics.

510 **Cournot Competition** In Figure 5, we present the conver-  
 511 gence performance of our two methods, measured by the dif-  
 512 ference between  $\vartheta_{\text{esti}}$  and  $\vartheta_{\text{true}}$ . Due to space limit, we in-  
 513 clude the convergence results about  $\|x_{\text{true}} - x_{\text{esti}}\|_2$  in Sup-  
 514 plementary Material. Comparing Figure 5a and Figure 5b, we  
 515 can see that our **I-SDP** method converges to a lower error at  
 516 a much faster rate compared to **I-MD**.

517 Figure 6 presents the boxplots concerning two error met-

rics across different estimation methods on the testing data. 518  
 It can be observed that our **I-SDP** and **I-MD** methods out- 519  
 perform all the other methods across both metrics. Specifi- 520  
 cally, our **I-SDP** achieves the lowest errors, with a mean er- 521  
 ror of 0.562 for  $\|\vartheta_{\text{esti}} - \vartheta_{\text{true}}\|_2$  and a mean error of 1.245 522  
 for  $\|x_{\text{esti}} - x_{\text{true}}\|_2$ . In contrast, the lowest errors among the 523  
 comparison methods are 1.089 and 3.203 for the two metrics. 524

525 **Traffic Game** Due to space limit, we leave the convergence 525  
 performance of our two methods about  $\|\vartheta_{\text{esti}} - \vartheta_{\text{true}}\|_2$  in our 526  
 Supplementary Material. Figure 7 presents a boxplot compar- 527  
 ison of our two methods against **Random** on the testing data. 528  
**Bertsimas** and **Feasibility** are excluded, because (i) **Bertsi-** 529  
**mas** becomes intractable, as in the traffic game it requires 530  
 solving an optimization problem with infinite constraints; (ii) 531  
**Feasibility** is impractical due to the vast size of the global 532  
 action space (e.g., discretizing each flow region with only 2 533  
 values leads to a size of  $2^{76}$ ). It is evident that both our **I-SDP** 534  
 and **I-MD** significantly outperform the **Random** method. 535

## 6 Conclusion 536

537 In this paper, we introduced a novel framework for estimat- 537  
 ing player utilities from their equilibrium behaviors. Our es- 538  
 timation framework is applicable to a broad range of multi- 539  
 player non-cooperative games and practical applications. We 540  
 also extended it to account for a specific structural property in 541  
 player utility functions. Experimental results on three game 542  
 applications demonstrate the superiority of our methods over 543  
 baselines. The main focus of our work is on efficiently com- 544  
 puting the incenter and evaluating its effectiveness. Future 545  
 directions will include analyzing the sample complexity of 546  
 our utility estimation method and integrating utility estima- 547  
 tion with the prediction of equilibrium behaviors. 548

## References

- [Alizadeh *et al.*, 1998] Farid Alizadeh, Jean-Pierre A Haeberly, and Michael L Overton. Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results. *SIAM Journal on Optimization*, 8(3):746–768, 1998.
- [Aswani *et al.*, 2018] Anil Aswani, Zuo-Jun Shen, and Auyon Siddiq. Inverse optimization with noisy data. *Operations Research*, 66(3):870–892, 2018.
- [Beck and Teboulle, 2003] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.
- [Berry *et al.*, 1995] Steven Berry, James Levinsohn, and Ariel Pakes. Automobile prices in market equilibrium. *Econometrica: Journal of the Econometric Society*, pages 841–890, 1995.
- [Berry, 1994] Steven T Berry. Estimating discrete-choice models of product differentiation. *The RAND Journal of Economics*, pages 242–262, 1994.
- [Bertsekas, 2016] Dimitri Bertsekas. *Nonlinear programming*, volume 4. Athena Scientific, 2016.
- [Bertsimas *et al.*, 2015] Dimitris Bertsimas, Vishal Gupta, and Ioannis Ch Paschalidis. Data-driven estimation in equilibrium using inverse optimization. *Mathematical Programming*, 153:595–633, 2015.
- [Besbes *et al.*, 2023] Omar Besbes, Yuri Fonseca, and Ilan Lobel. Contextual inverse optimization: Offline and online learning. *Operations Research*, 2023.
- [Blum *et al.*, 2014] Avrim Blum, Nika Haghtalab, and Ariel D Procaccia. Learning optimal commitment to overcome insecurity. *Advances in Neural Information Processing Systems*, 27, 2014.
- [Bnouhachem *et al.*, 2015] Abdellah Bnouhachem, Qamrul Hasan Ansari, and Ching-Feng Wen. A projection descent method for solving variational inequalities. *Journal of Inequalities and Applications*, 2015:1–14, 2015.
- [Boyd and Vandenberghe, 2004] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.
- [Bubeck and others, 2015] Sébastien Bubeck *et al.* Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.
- [Chan *et al.*, 2023] Timothy CY Chan, Rafid Mahmood, and Ian Yihang Zhu. Inverse optimization: Theory and applications. *Operations Research*, 2023.
- [Facchinei and Pang, 2003] Francisco Facchinei and Jong-Shi Pang. *Finite-dimensional variational inequalities and complementarity problems*. Springer, 2003.
- [Fukushima, 1992] Masao Fukushima. Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems. *Mathematical programming*, 53:99–110, 1992.
- [Haghtalab *et al.*, 2016] Nika Haghtalab, Fei Fang, Thanh Hong Nguyen, Arunesh Sinha, Ariel D Procaccia, and Milind Tambe. *Three strategies to success: Learning adversary models in security games*. 2016.
- [Harker and Pang, 1990] Patrick T Harker and Jong-Shi Pang. Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications. *Mathematical programming*, 48(1):161–220, 1990.
- [Heuberger, 2004] Clemens Heuberger. Inverse combinatorial optimization: A survey on problems, methods, and results. *Journal of combinatorial optimization*, 8:329–361, 2004.
- [Jensen, 2018] Martin Kaae Jensen. Aggregative games. In *Handbook of Game Theory and Industrial Organization, Volume I*, pages 66–92. Edward Elgar Publishing, 2018.
- [Korpelevich, 1976] Galina M Korpelevich. The extragradient method for finding saddle points and other problems. *Matecon*, 12:747–756, 1976.
- [Kuleshov and Schrijvers, 2015] Volodymyr Kuleshov and Okke Schrijvers. Inverse game theory: Learning utilities in succinct games. In *Web and Internet Economics: 11th International Conference, WINE 2015, Amsterdam, The Netherlands, December 9-12, 2015, Proceedings 11*, pages 413–427. Springer, 2015.
- [LeBlanc *et al.*, 1975] Larry J LeBlanc, Edward K Morlok, and William P Pierskalla. An efficient approach to solving the road network equilibrium traffic assignment problem. *Transportation research*, 9(5):309–318, 1975.
- [Ling *et al.*, 2018] Chun Kai Ling, Fei Fang, and J Zico Kolter. What game are we playing? end-to-end learning in normal and extensive form games. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 396–402, 2018.
- [Ling *et al.*, 2019] Chun Kai Ling, Fei Fang, and J Zico Kolter. Large scale learning of agent rationality in two-player zero-sum games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6104–6111, 2019.
- [Maddux *et al.*, 2023] Anna M Maddux, Nicolò Pagan, Giuseppe Belgioioso, and Florian Dörfler. Data-driven behaviour estimation in parametric games. *IFAC-PapersOnLine*, 56(2):9330–9335, 2023.
- [Mohajerin Esfahani *et al.*, 2018] Peyman Mohajerin Esfahani, Soroosh Shafieezadeh-Abadeh, Grani A Hanasusanto, and Daniel Kuhn. Data-driven inverse optimization with imperfect information. *Mathematical Programming*, 167:191–234, 2018.
- [Narahari *et al.*, 2009] Yadati Narahari, Dinesh Garg, Ramasuri Narayanam, and Hastagiri Prakash. *Game theoretic problems in network economics and mechanism design solutions*. Springer Science & Business Media, 2009.
- [Nash, 1950] John F Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences of the United States of America*, 36(1):48–49, 1950.

- 657 [Nesterov and Nemirovskii, 1994] Yurii Nesterov and  
658 Arkadii Nemirovskii. *Interior-point polynomial algo-*  
659 *rithms in convex programming*. SIAM, 1994.
- 660 [Noti, 2021] Gali Noti. From behavioral theories to econo-  
661 metrics: Inferring preferences of human agents from data  
662 on repeated interactions. In *Proceedings of the AAAI Con-*  
663 *ference on Artificial Intelligence*, volume 35, pages 5637–  
664 5646, 2021.
- 665 [Patriksson, 2015] Michael Patriksson. *The traffic assign-*  
666 *ment problem: models and methods*. Courier Dover Pub-  
667 lications, 2015.
- 668 [Peters *et al.*, 2023] Lasse Peters, Vicenç Rubies-Royo,  
669 Claire J Tomlin, Laura Ferranti, Javier Alonso-Mora,  
670 Cyrill Stachniss, and David Fridovich-Keil. Online and  
671 offline learning of player objectives from partial obser-  
672 vations in dynamic games. *The International Journal of*  
673 *Robotics Research*, 42(10):917–937, 2023.
- 674 [Risanger *et al.*, 2020] Simon Risanger, Stein-Erik Fleten,  
675 and Steven A Gabriel. Inverse equilibrium analysis of  
676 oligopolistic electricity markets. *IEEE Transactions on*  
677 *Power Systems*, 35(6):4159–4166, 2020.
- 678 [Scroccaro *et al.*, 2023] Pedro Zattoni Scroccaro, Bilge Ata-  
679 soy, and Peyman Mohajerin Esfahani. Learning in inverse  
680 optimization: Incenter cost, augmented suboptimality loss,  
681 and algorithms. *arXiv preprint arXiv:2305.07730*, 2023.
- 682 [Sheffi, 1985] Yosef Sheffi. *Urban transportation networks*,  
683 volume 6. Prentice-Hall, Englewood Cliffs, NJ, 1985.
- 684 [Tsai *et al.*, 2016] Dorian Tsai, Timothy L Molloy, and Tris-  
685 tan Perez. Inverse two-player zero-sum dynamic games. In  
686 *2016 Australian Control Conference (AuCC)*, pages 192–  
687 196. IEEE, 2016.
- 688 [Vandenberghe and Boyd, 1996] Lieven Vandenberghe and  
689 Stephen Boyd. Semidefinite programming. *SIAM review*,  
690 38(1):49–95, 1996.
- 691 [Waugh *et al.*, 2011] Kevin Waugh, Brian D Ziebart, and  
692 J Andrew Bagnell. Computational rationalization: the in-  
693 verse equilibrium problem. In *Proceedings of the 28th*  
694 *International Conference on International Conference on*  
695 *Machine Learning*, pages 1169–1176, 2011.
- 696 [Wu *et al.*, 2022] Jibang Wu, Weiran Shen, Fei Fang, and  
697 Haifeng Xu. Inverse game theory for stackelberg games:  
698 the blessing of bounded rationality. *Advances in Neural*  
699 *Information Processing Systems*, 35:32186–32198, 2022.
- 700 [Xiu and Zhang, 2003] Naihua Xiu and Jianzhong Zhang.  
701 Some recent advances in projection-type methods for vari-  
702 ational inequalities. *Journal of Computational and Ap-*  
703 *plied Mathematics*, 152(1-2):559–585, 2003.
- 704 [Yu *et al.*, 2022] Yue Yu, Jonathan Salfity, David Fridovich-  
705 Keil, and Ufuk Topcu. Inverse matrix games with unique  
706 quantal response equilibrium. *IEEE Control Systems Let-*  
707 *ters*, 7:643–648, 2022.

# Inverse Game Theory: An Incenter-Based Approach (Supplementary Material)

## 1 Proof of Theorem 1

*Proof.* Since the angle in the objective is scale-invariant, we first assume that  $\|\boldsymbol{\vartheta}\|_2 = \|\tilde{\boldsymbol{\vartheta}}\|_2 = 1$  for ease of the derivation. We later demonstrate that this constraint can be removed in the convex formulation. Under this assumption, we have

$$\langle \boldsymbol{\vartheta}, \tilde{\boldsymbol{\vartheta}} \rangle = -\frac{1}{2} \|\boldsymbol{\vartheta} - \tilde{\boldsymbol{\vartheta}}\|_2^2 + 1. \quad (1)$$

The problem (5) in Section 4.2 can be rewritten as

$$\begin{aligned} \arg \max_{\|\boldsymbol{\vartheta}\|_2=1} \min_{\substack{\tilde{\boldsymbol{\vartheta}} \in \text{int}(\mathbb{C}) \\ \|\tilde{\boldsymbol{\vartheta}}\|_2=1}} a(\boldsymbol{\vartheta}, \tilde{\boldsymbol{\vartheta}}) &= \arg \max_{\|\boldsymbol{\vartheta}\|_2=1} \min_{\substack{\tilde{\boldsymbol{\vartheta}} \in \text{int}(\mathbb{C}) \\ \|\tilde{\boldsymbol{\vartheta}}\|_2=1}} \arccos \langle \boldsymbol{\vartheta}, \tilde{\boldsymbol{\vartheta}} \rangle \\ &= \arg \max_{\|\boldsymbol{\vartheta}\|_2=1} \min_{\|\tilde{\boldsymbol{\vartheta}}\|_2=1} \arccos \left( -\frac{1}{2} \|\boldsymbol{\vartheta} - \tilde{\boldsymbol{\vartheta}}\|_2^2 + 1 \right) \\ &= \arg \max_{\|\boldsymbol{\vartheta}\|_2=1} \min_{\substack{\tilde{\boldsymbol{\vartheta}} \in \text{int}(\mathbb{C}) \\ \|\tilde{\boldsymbol{\vartheta}}\|_2=1}} \|\boldsymbol{\vartheta} - \tilde{\boldsymbol{\vartheta}}\|_2^2. \end{aligned} \quad (2)$$

The above equation follows from the facts that the range of the function  $\arccos$  is  $[0, \pi]$ , and  $-\cos(\gamma)$  is monotone increasing for  $\gamma \in [0, \pi]$ .

In order to minimize the distance to any  $\boldsymbol{\vartheta}$ , the optimal  $\tilde{\boldsymbol{\vartheta}}$  will always be in the boundary of  $\mathbb{C}$ . Hence, the inner minimization problem in equation (2) becomes

$$\min_{\substack{\tilde{\boldsymbol{\vartheta}} \in \text{int}(\mathbb{C}) \\ \|\tilde{\boldsymbol{\vartheta}}\|_2=1}} \|\boldsymbol{\vartheta} - \tilde{\boldsymbol{\vartheta}}\|_2^2 = \min_{\substack{\boldsymbol{x}^j \in \mathcal{X}^j \\ j \in [N]}} \left\{ \min_{\substack{\langle \Phi_{\tilde{\boldsymbol{\vartheta}}}^j, \boldsymbol{x}^j - \hat{\boldsymbol{x}}^j \rangle = 0 \\ \|\tilde{\boldsymbol{\vartheta}}\|_2=1}} \|\boldsymbol{\vartheta} - \tilde{\boldsymbol{\vartheta}}\|_2^2 \right\}. \quad (3)$$

Next, we first derive a solution to the following minimization problems which is related to the inside curly bracket in (3).

$$\begin{aligned} \min_{\tilde{\boldsymbol{\vartheta}}} \quad & \|\boldsymbol{\vartheta} - \tilde{\boldsymbol{\vartheta}}\|_2^2 \\ \text{s.t.} \quad & \langle \Phi_{\tilde{\boldsymbol{\vartheta}}}^j, \boldsymbol{x}^j - \hat{\boldsymbol{x}}^j \rangle = 0. \end{aligned} \quad (4)$$

The Lagrangian function of Problem (4) is

$$L(\tilde{\boldsymbol{\vartheta}}, v) = \|\boldsymbol{\vartheta} - \tilde{\boldsymbol{\vartheta}}\|_2^2 + v \left( \langle \Phi_{\tilde{\boldsymbol{\vartheta}}}^j, \boldsymbol{x}^j - \hat{\boldsymbol{x}}^j \rangle \right). \quad (5)$$

Correspondingly, we have the Lagrange dual function as

$$g(v) = \inf_{\tilde{\boldsymbol{\vartheta}}} L(\tilde{\boldsymbol{\vartheta}}, v) = \inf_{\tilde{\boldsymbol{\vartheta}}} \|\boldsymbol{\vartheta} - \tilde{\boldsymbol{\vartheta}}\|_2^2 + v \left( \langle \Phi_{\tilde{\boldsymbol{\vartheta}}}^j, \boldsymbol{x}^j - \hat{\boldsymbol{x}}^j \rangle \right). \quad (6)$$

To derive  $g(v)$ , we first derive  $\frac{\partial L(\tilde{\boldsymbol{\vartheta}}, v)}{\partial \tilde{\boldsymbol{\vartheta}}} = 0$  as follows.

$$\begin{aligned} \frac{\partial L(\tilde{\boldsymbol{\vartheta}}, v)}{\partial \tilde{\boldsymbol{\vartheta}}} &= -2(\boldsymbol{\vartheta} - \tilde{\boldsymbol{\vartheta}}) + v \cdot \frac{\partial \langle \Phi_{\tilde{\boldsymbol{\vartheta}}}^j, \boldsymbol{x}^j - \hat{\boldsymbol{x}}^j \rangle}{\partial \tilde{\boldsymbol{\vartheta}}} \\ &= -2(\boldsymbol{\vartheta} - \tilde{\boldsymbol{\vartheta}}) + v \cdot \boldsymbol{\phi}^j \odot \text{vec}(\mathbb{1}_n \otimes (\boldsymbol{x}^j - \hat{\boldsymbol{x}}^j)) = 0. \end{aligned} \quad (7)$$

Here,  $\odot$  denotes Hadamard product. Then, we obtain

$$\tilde{\boldsymbol{\vartheta}} = \boldsymbol{\vartheta} - \frac{v}{2} \boldsymbol{\phi}^j \odot \text{vec}(\mathbb{1}_n \otimes (\boldsymbol{x}^j - \hat{\boldsymbol{x}}^j)), \quad (8)$$

where  $\mathbb{1}_n \in \mathbb{R}^n$ ,  $\otimes$  is Kronecker product, and  $\text{vec}$  means the vectorization of a matrix. Plugging (8) into (6) gives  $g(v)$  as

$$\left\| \frac{v}{2} \boldsymbol{\phi}^j \odot \text{vec}(\mathbb{1}_n \otimes (\boldsymbol{x}^j - \hat{\boldsymbol{x}}^j)) \right\|_2^2 + v \left( \langle \tilde{\Phi}_{\tilde{\boldsymbol{\vartheta}}}^j, \boldsymbol{x}^j - \hat{\boldsymbol{x}}^j \rangle \right), \quad (9)$$

where  $\tilde{\Phi}_{\tilde{\boldsymbol{\vartheta}}}^j = \left[ (\boldsymbol{\theta}_i - \frac{v}{2} \boldsymbol{\phi}_i(\hat{\boldsymbol{x}}^j, \hat{\boldsymbol{s}}^j) (\boldsymbol{x}_i^j - \hat{\boldsymbol{x}}_i^j))^\top \boldsymbol{\phi}_i(\hat{\boldsymbol{x}}^j, \hat{\boldsymbol{s}}^j) \right]_{i=1}^p \in \mathbb{R}^p$ . Then, we derive the dual problem  $\max_v g(v)$  of (4) as

$$\max_v \left\| \frac{v}{2} \boldsymbol{\phi}^j \odot \text{vec}(\mathbb{1}_n \otimes (\boldsymbol{x}^j - \hat{\boldsymbol{x}}^j)) \right\|_2^2 + v \left( \langle \tilde{\Phi}_{\tilde{\boldsymbol{\vartheta}}}^j, \boldsymbol{x}^j - \hat{\boldsymbol{x}}^j \rangle \right). \quad (10)$$

Based on (10), we further compute  $\frac{\partial g(v)}{\partial v}$  as follows.

$$\begin{aligned} \frac{\partial g(v)}{\partial v} &= \frac{v}{2} \left\| \boldsymbol{\phi}^j \odot \text{vec}(\mathbb{1}_n \otimes (\boldsymbol{x}^j - \hat{\boldsymbol{x}}^j)) \right\|_2^2 \\ &\quad + \langle \tilde{\Phi}_{\tilde{\boldsymbol{\vartheta}}}^j, \boldsymbol{x}^j - \hat{\boldsymbol{x}}^j \rangle + v \left( \langle \dot{\Phi}_{\tilde{\boldsymbol{\vartheta}}}^j, \boldsymbol{x}^j - \hat{\boldsymbol{x}}^j \rangle \right), \end{aligned} \quad (11)$$

where  $\dot{\Phi}^j = \left[ -\frac{1}{2} \boldsymbol{\phi}_i(\hat{\boldsymbol{x}}^j, \hat{\boldsymbol{s}}^j)^\top (\boldsymbol{x}_i^j - \hat{\boldsymbol{x}}_i^j) \boldsymbol{\phi}_i(\hat{\boldsymbol{x}}^j, \hat{\boldsymbol{s}}^j) \right]_{i=1}^p$ , which belongs to  $\mathbb{R}^p$ .

To further derive (11), we rewrite  $\tilde{\Phi}_{\tilde{\boldsymbol{\vartheta}}}^j$  in (11) as

$$\begin{aligned} & \left[ \boldsymbol{\theta}_i^\top \boldsymbol{\phi}_i(\hat{\boldsymbol{x}}^j, \hat{\boldsymbol{s}}^j) - \frac{v}{2} \boldsymbol{\phi}_i(\hat{\boldsymbol{x}}^j, \hat{\boldsymbol{s}}^j)^\top (\boldsymbol{x}_i^j - \hat{\boldsymbol{x}}_i^j) \boldsymbol{\phi}_i(\hat{\boldsymbol{x}}^j, \hat{\boldsymbol{s}}^j) \right]_i \\ &= \left[ \boldsymbol{\theta}_i^\top \boldsymbol{\phi}_i(\hat{\boldsymbol{x}}^j, \hat{\boldsymbol{s}}^j) \right]_i - \frac{v}{2} \left[ \boldsymbol{\phi}_i(\hat{\boldsymbol{x}}^j, \hat{\boldsymbol{s}}^j)^\top (\boldsymbol{x}_i^j - \hat{\boldsymbol{x}}_i^j) \boldsymbol{\phi}_i(\hat{\boldsymbol{x}}^j, \hat{\boldsymbol{s}}^j) \right]_i \\ &= \Phi_{\tilde{\boldsymbol{\vartheta}}}^j + v \dot{\Phi}^j. \end{aligned} \quad (12)$$

27 Plugging (12) into (11), we can derive  $\frac{\partial g(v)}{\partial v} = 0$  as

$$\begin{aligned} & \frac{v}{2} \left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2^2 + \langle \Phi_{\vartheta}^j + v \dot{\Phi}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \\ & + v \langle \dot{\Phi}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle = \frac{v}{2} \left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2^2 \\ & + \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + 2v \langle \dot{\Phi}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle = 0. \end{aligned}$$

28 Then, we can derive  $v$  as the following equation.

$$v = \frac{-2 \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle}{\left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2^2 + 4 \langle \dot{\Phi}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle}. \quad (13)$$

29 Plugging (13) into (8), we get the form of  $\tilde{\vartheta}$  as follows.

$$\begin{aligned} \tilde{\vartheta} &= \vartheta - \left\{ \frac{\phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))}{2} \right. \\ & \quad \left. \cdot \frac{2 \left( -\langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \right)}{\left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2^2 + 4 \langle \dot{\Phi}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle} \right\} \\ &= \vartheta - \frac{\left( -\langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \right) \left( \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right)}{\left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2^2 + 4 \langle \dot{\Phi}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle}. \quad (14) \end{aligned}$$

30 To simplify (14), we rewrite  $\dot{\Phi}^j \in \mathbb{R}^p$  in the following way.

$$\begin{aligned} \dot{\Phi}^j &= \left[ -\frac{1}{2} \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)^\top \left( \mathbf{x}_i^j - \hat{\mathbf{x}}_i^j \right) \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) \right]_{i=1}^p \\ &= -\frac{1}{2} \left[ \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)^\top \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) \right]_{i=1}^p \odot (\mathbf{x}^j - \hat{\mathbf{x}}^j). \end{aligned}$$

31 Then, we can derive  $4 \langle \dot{\Phi}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle$  as follows.

$$\begin{aligned} & 4 \langle \dot{\Phi}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \\ &= -2 \left\langle \left[ \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)^\top \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) \right]_i \odot (\mathbf{x}^j - \hat{\mathbf{x}}^j), \mathbf{x}^j - \hat{\mathbf{x}}^j \right\rangle \\ &= -2 \left\langle \left[ \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)^\top \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) \right]_i, (\mathbf{x}^j - \hat{\mathbf{x}}^j) \odot (\mathbf{x}^j - \hat{\mathbf{x}}^j) \right\rangle \\ &= -2 \left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2^2. \quad (15) \end{aligned}$$

32 Plugging (15) into (14), we get  $\tilde{\vartheta}$  denoted by  $\omega(\vartheta, \mathbf{x}^j)$ :

$$\begin{aligned} \omega(\vartheta, \mathbf{x}^j) &:= \arg \min_{\langle \Phi_{\tilde{\vartheta}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle = 0} \left\| \vartheta - \tilde{\vartheta} \right\|_2^2 \\ &= \vartheta - \frac{\langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \left( \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right)}{\left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2^2}. \quad (16) \end{aligned}$$

33 Then, we can obtain a solution to the minimization problem  
34 inside the curly brackets in (3) as follows.

$$\arg \min_{\substack{\langle \Phi_{\tilde{\vartheta}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle = 0 \\ \|\tilde{\vartheta}\|_2 = 1}} \left\| \vartheta - \tilde{\vartheta} \right\|_2^2 = \frac{\omega(\vartheta, \mathbf{x}^j)}{\|\omega(\vartheta, \mathbf{x}^j)\|_2}. \quad (17)$$

35 It can be easy to further prove (17) by contradiction.

Combining (2) and (17), we have the equation:

36

$$\min_{\substack{\tilde{\vartheta} \in \text{int}(\mathbb{C}) \\ \|\tilde{\vartheta}\|_2 = 1}} \left\| \vartheta - \tilde{\vartheta} \right\|_2^2 = \min_{\substack{\mathbf{x}^j \in \mathcal{X}^j \\ j \in [N]}} \left\| \vartheta - \frac{\omega(\vartheta, \mathbf{x}^j)}{\|\omega(\vartheta, \mathbf{x}^j)\|_2} \right\|_2^2.$$

Since we can easily derive that  $\langle \vartheta, \omega(\vartheta, \mathbf{x}^j) \rangle \geq 0$ , i.e.,  
37  $a(\vartheta, \omega(\vartheta, \mathbf{x}^j)) \leq \frac{\pi}{2}$ , and  $\omega(\vartheta, \mathbf{x}^j) \perp (\vartheta - \omega(\vartheta, \mathbf{x}^j))$ , we  
38 then derive the following equation.  
39

$$\begin{aligned} & \arg \min_{\mathbf{x}^j} \left\| \vartheta - \frac{\omega(\vartheta, \mathbf{x}^j)}{\|\omega(\vartheta, \mathbf{x}^j)\|_2} \right\|_2^2 = \arg \min_{\mathbf{x}^j} a \left( \vartheta, \frac{\omega(\vartheta, \mathbf{x}^j)}{\|\omega(\vartheta, \mathbf{x}^j)\|_2} \right) \\ &= \arg \min_{\mathbf{x}^j} a(\vartheta, \omega(\vartheta, \mathbf{x}^j)) = \arg \min_{\mathbf{x}^j} \sin(a(\vartheta, \omega(\vartheta, \mathbf{x}^j))) \\ &= \arg \min_{\mathbf{x}^j} \left\| \vartheta - \omega(\vartheta, \mathbf{x}^j) \right\|_2. \quad (18) \end{aligned}$$

Combining (17) and (18), we further derive that

40

$$\begin{aligned} & \arg \min_{\mathbf{x}^j} \left\| \vartheta - \frac{\omega(\vartheta, \mathbf{x}^j)}{\|\omega(\vartheta, \mathbf{x}^j)\|_2} \right\|_2^2 = \arg \min_{\mathbf{x}^j} \left\| \vartheta - \omega(\vartheta, \mathbf{x}^j) \right\|_2 \\ &= \arg \min_{\mathbf{x}^j} \left\| \frac{\langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \left( \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right)}{\left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2^2} \right\|_2 \\ &= \arg \min_{\mathbf{x}^j} \frac{\left| \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \right|}{\left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2}. \end{aligned}$$

Therefore, we can further express (2) as follows.

41

$$\begin{aligned} & \arg \max_{\|\vartheta\|_2 = 1} \min_{\substack{\tilde{\vartheta} \in \text{int}(\mathbb{C}) \\ \|\tilde{\vartheta}\|_2 = 1}} a(\vartheta, \tilde{\vartheta}) \\ &= \arg \max_{\|\vartheta\|_2 = 1} \min_{\substack{\mathbf{x}^j \in \mathcal{X}^j \\ j \in [N]}} \frac{\left| \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \right|}{\left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2}. \quad (19) \end{aligned}$$

Next, by applying an epigraph reformulation for (19), we obtain  
42 the following equivalent optimization problem:  
43

$$\begin{aligned} & \arg \max_{\|\vartheta\|_2 = 1} r \\ & \text{s.t. } r = \min_{\substack{\mathbf{x}^j \in \mathcal{X}^j \\ j \in [N]}} \frac{\left| \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \right|}{\left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2}, \quad (20) \\ & \quad \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N]. \end{aligned}$$

We rewrite the constraints in (20) as the following form.

44

$$\begin{aligned} & r \leq \frac{\left| \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \right|}{\left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2}, \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N], \\ & \Rightarrow - \left| \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \right| + r \left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2 \leq 0 \\ & \Rightarrow \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + r \left\| \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \right\|_2 \leq 0. \end{aligned}$$

Based on the above form of the constraints, we can derive the

45

46 equivalent optimization form for (19) as follows.

$$\begin{aligned}
& \arg \max_{\boldsymbol{\vartheta}, r} \quad r \\
& \text{s.t.} \quad \langle \Phi_{\boldsymbol{\vartheta}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + r \|\phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))\|_2 \leq 0, \\
& \quad \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N], \\
& \quad \|\boldsymbol{\vartheta}\|_2 = 1.
\end{aligned} \tag{21}$$

47 Next, we remove the constraint  $\|\boldsymbol{\vartheta}\|_2 = 1$  in (21) and re-  
48 formulate it as a convex optimization problem.

49 We first illustrate the assumption of a nonempty interior of  
50  $\mathbb{C}$  indicates that the optimal  $r$  in problem (21) is positive. To  
51 clarify this point, under the assumption, there exists some  $\boldsymbol{\vartheta}$   
52 such that  $\langle \Phi_{\boldsymbol{\vartheta}}^j, \hat{\mathbf{x}}^j - \mathbf{x}^j \rangle > 0, \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N]$ . Hence,  
53 the following expression

$$r = \frac{\min_{\mathbf{x}^j \in \mathcal{X}^j, j \in [N]} \langle \Phi_{\boldsymbol{\vartheta}}^j, \hat{\mathbf{x}}^j - \mathbf{x}^j \rangle}{\max_{\mathbf{x}^j \in \mathcal{X}^j, j \in [N]} \|\phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))\|_2} > 0$$

54 is a feasible solution to (21), confirming that the optimal  $r >$   
55  $0$ . Given  $\|\boldsymbol{\vartheta}\|_2 = 1$ , problem (21) can be further reformulated  
56 as the following minimization problem.

$$\begin{aligned}
& \min_{\boldsymbol{\vartheta}, r} \quad \|\boldsymbol{\vartheta}\|_2 / r \\
& \text{s.t.} \quad \langle \Phi_{\boldsymbol{\vartheta}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + r \|\phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))\|_2 \leq 0, \\
& \quad \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N], \\
& \quad \|\boldsymbol{\vartheta}\|_2 = 1.
\end{aligned}$$

57 Let  $\tilde{\boldsymbol{\vartheta}} = \boldsymbol{\vartheta}/r$ , we can rewrite the above problem as

$$\begin{aligned}
& \min_{\tilde{\boldsymbol{\vartheta}}, r} \quad \|\tilde{\boldsymbol{\vartheta}}\|_2 \\
& \text{s.t.} \quad \langle \Phi_{\tilde{\boldsymbol{\vartheta}}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + r \|\phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))\|_2 \leq 0, \\
& \quad \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N], \\
& \quad \|\tilde{\boldsymbol{\vartheta}}\|_2 = 1/r.
\end{aligned}$$

58 Since  $r$  only appears in the constraint in above problem. Let  
59  $r = 1/\|\tilde{\boldsymbol{\vartheta}}\|_2$  for any  $\tilde{\boldsymbol{\vartheta}}$ , it will not affect the optimal value of  
60  $\tilde{\boldsymbol{\vartheta}}$ . So we can simply drop the constraint  $\|\tilde{\boldsymbol{\vartheta}}\|_2 = 1/r$ .  $\square$

## 61 2 Robustness of Incenter to Perturbations

62 We provide additional details supporting the robustness in-  
63 terpretation of the incenter in Section 4.2. We formalize this  
64 robustness notion as follows. First, recall the definition of the  
65 set of consistent parameter vectors

$$\mathbb{C} := \left\{ \boldsymbol{\vartheta} : \langle \Phi_{\boldsymbol{\vartheta}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \leq 0, \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N] \right\}.$$

66 Here, we can rewrite  $\langle \Phi_{\boldsymbol{\vartheta}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle$  as follows.

$$\begin{aligned}
\langle \Phi_{\boldsymbol{\vartheta}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle &= \langle [\boldsymbol{\theta}_i^\top \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)]_{i=1}^p, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \\
&= \langle \boldsymbol{\vartheta}, \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \rangle,
\end{aligned}$$

67 Thus, the set  $\mathbb{C}$  can be rewritten as the following form.

$$\mathbb{C} := \left\{ \boldsymbol{\vartheta} : \langle \boldsymbol{\vartheta}, \phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j)) \rangle \leq 0, \forall \mathbf{x}^j, \forall j \right\}.$$

Next, we define vectors  $\Delta(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j, \mathbf{x}^j)$  by normalizing the  
vectors  $\phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))$  within the set  $\mathbb{C}$ :

$$\Delta(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j, \mathbf{x}^j) = \begin{cases} \frac{\phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))}{\|\phi^j \odot \text{vec}(\mathbb{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))\|_2}, & \text{if } \mathbf{x}^j \neq \hat{\mathbf{x}}^j; \\ \mathbf{0}, & \text{Otherwise.} \end{cases}$$

We notice that this normalization does not change the set  $\mathbb{C}$ ,  
so we can further rewrite  $\mathbb{C}$  using  $\Delta(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j, \mathbf{x}^j)$  as

$$\mathbb{C} := \left\{ \boldsymbol{\vartheta} : \langle \boldsymbol{\vartheta}, \Delta(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j, \mathbf{x}^j) \rangle \leq 0, \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N] \right\}.$$

It can be observed that the vectors  $\Delta(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j, \mathbf{x}^j)$  are deter-  
mined by the dataset  $\hat{\mathcal{D}} = \{(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)\}_{j=1}^N$ . Hence, the vector  
 $\boldsymbol{\vartheta} \in \mathbb{C}$  which is most robust to perturbations in the data (i.e.,  
perturbation of  $\Delta(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j, \mathbf{x}^j)$ ) can be found by solving

$$\begin{aligned}
& \arg \max_{\substack{\|\boldsymbol{\vartheta}\|_2=1 \\ \boldsymbol{\vartheta} \in \mathbb{C}}} \min_{\substack{\gamma \in \mathbb{R}^{np} \\ (\hat{\mathbf{x}}^k, \hat{\mathbf{s}}^k) \in \{(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)\}_{j=1}^N \\ \mathbf{x}^j \in \mathcal{X}^j}} \|\gamma\|_2^2 \\
& \text{s.t.} \quad \langle \boldsymbol{\vartheta}, \Delta(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j, \mathbf{x}^j) + \gamma \rangle = 0.
\end{aligned} \tag{22}$$

In problem (22), the outer maximization player optimizes  
 $\boldsymbol{\vartheta} \in \mathbb{C}$  to require the largest perturbation  $\gamma$  for it to lie on the  
perturbed facet of  $\mathbb{C}$ . The inner minimization player seeks the  
most vulnerable facet of  $\mathbb{C}$ , which requires the smallest per-  
turbation vector  $\gamma$  to satisfy  $\langle \boldsymbol{\vartheta}, \Delta(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j, \mathbf{x}^j) + \gamma \rangle = 0$ . It  
can be easily illustrated that Problem (22) is equivalent to the  
incenter  $\boldsymbol{\vartheta}^{\text{in}}$  reformulation in Theorem 1.

## 63 3 Convergence of Algorithm 1

We can derive the convergence rate of the mirror descent  
algorithm for solving problem (8) as follows. We assume  
that Bregman divergence function  $\mathcal{B}_\omega(\boldsymbol{\vartheta}, \boldsymbol{\nu}) \leq R$  for some  
 $R > 0$ , and  $\mathbb{E}\|g_t(\boldsymbol{\vartheta})\|_*^2 \leq G$  for all  $\boldsymbol{\vartheta}, \boldsymbol{\nu}$ . Let  $f(\boldsymbol{\vartheta})$  be the  
objective of problem (8). With learning rate  $\eta_t = \frac{c}{\sqrt{t}}$  for some  
constant  $c > 0$ , the learning algorithm guarantees that

$$\mathbb{E} \left[ f \left( \frac{1}{T} \sum_{t=1}^T \boldsymbol{\vartheta}_t \right) \right] - \min_{\boldsymbol{\vartheta}} f(\boldsymbol{\vartheta}) \leq \frac{1}{\sqrt{T}} \left( \frac{R}{c} + cG \right). \tag{23}$$

*Proof.* Let  $\boldsymbol{\vartheta}^* \in \arg \min_{\boldsymbol{\vartheta} \in \Theta} f(\boldsymbol{\vartheta})$ . By the definition of the  
subgradient, we have that

$$\mathbb{E} [f(\boldsymbol{\vartheta}_t) - f(\boldsymbol{\vartheta}^*)] \leq \mathbb{E} [\langle g(\boldsymbol{\vartheta}_t), \boldsymbol{\vartheta}_t - \boldsymbol{\vartheta}^* \rangle] \tag{24}$$

Next, we apply the standard inequality for analyzing the mir-  
ror descent algorithm:

$$\begin{aligned}
\langle g(\boldsymbol{\vartheta}_t), \boldsymbol{\vartheta}_t - \boldsymbol{\vartheta}^* \rangle &\leq \frac{1}{\eta_t} (\mathcal{B}_\omega(\boldsymbol{\vartheta}^*, \boldsymbol{\vartheta}_t) - \mathcal{B}_\omega(\boldsymbol{\vartheta}^*, \boldsymbol{\vartheta}_{t+1})) \\
&\quad + \frac{\eta_t}{2} \|g(\boldsymbol{\vartheta}_t)\|_*^2.
\end{aligned} \tag{25}$$

94 Then, we can further derive the expectation form of (25) as

$$\begin{aligned}
& \mathbb{E} \left[ \sum_{t=1}^T \langle g(\boldsymbol{\vartheta}_t), \boldsymbol{\vartheta}_t - \boldsymbol{\vartheta}^* \rangle \right] \\
& \leq \mathbb{E} \left[ \sum_{t=1}^T \frac{1}{\eta_t} (\mathcal{B}_\omega(\boldsymbol{\vartheta}^*, \boldsymbol{\vartheta}_t) - \mathcal{B}_\omega(\boldsymbol{\vartheta}^*, \boldsymbol{\vartheta}_{t+1})) + \frac{1}{2} \sum_{t=1}^T \eta_t \|g\|_*^2 \right] \\
& \leq \frac{R}{\eta_1} + R \sum_{t=2}^T \left( \frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} \right) + \frac{G}{2} \sum_{t=1}^T \eta_t \\
& = \frac{R}{\eta_T} + \frac{G}{2} \sum_{t=1}^T \eta_t \leq \left( \frac{R}{c} + cG \right) \sqrt{T}. \tag{26}
\end{aligned}$$

95 Finally, we use Jensen's inequality to further derive that

$$\begin{aligned}
\mathbb{E} \left[ f \left( \frac{1}{T} \sum_{t=1}^T \boldsymbol{\vartheta}_t \right) - f(\boldsymbol{\vartheta}^*) \right] & \leq \frac{1}{T} \mathbb{E} \left[ \sum_{t=1}^T (f(\boldsymbol{\vartheta}_t) - f(\boldsymbol{\vartheta}^*)) \right] \\
& \leq \frac{1}{T} \mathbb{E} \left[ \sum_{t=1}^T \langle g(\boldsymbol{\vartheta}_t), \boldsymbol{\vartheta}_t - \boldsymbol{\vartheta}^* \rangle \right] \\
& \leq \frac{1}{\sqrt{T}} \left( \frac{R}{c} + cG \right). \tag{27}
\end{aligned}$$

96 □

## 97 4 Proof of Lemma 1

98 *Proof.* In the two-firm Bertrand-Nash competition with a linear demand function, the corresponding Jacobian matrix

$$\nabla \mathbf{F}(\mathbf{x}) = - \begin{bmatrix} 2\theta_{11} & \theta_{12} \\ \theta_{21} & 2\theta_{22} \end{bmatrix}$$

100 is symmetric due to the constraint that  $\theta_{21} = \theta_{12} = 1$ .

101 Let variables  $\mathbf{\Xi} = - \begin{bmatrix} \theta_{11} & \theta_{21}/2 \\ \theta_{12}/2 & \theta_{22} \end{bmatrix}$ ,  $\tilde{\mathbf{\Xi}} = \begin{bmatrix} \theta_{13} & \theta_{23} \\ \theta_{14} & \theta_{24} \end{bmatrix}$ ,

102 then problem (8) in Section 4.3 can be reformulated as

$$\begin{aligned}
\min_{\mathbf{\Xi}, \tilde{\mathbf{\Xi}}} & \frac{1}{N} \sum_{j=1}^N \ell_{\mathbf{\Xi}, \tilde{\mathbf{\Xi}}}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) + \alpha \mathcal{R}(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}) \\
\text{s.t.} & \quad \mathbf{\Xi}_{ii} \geq 0, \quad \mathbf{\Xi}_{i,-i} = -\frac{1}{2}, \forall i = 1, 2, \\
& \quad \mathbf{\Xi} \geq 0, \quad \tilde{\mathbf{\Xi}} \geq 0. \tag{28}
\end{aligned}$$

103 Here,  $\mathbf{\Xi}_{ii} \geq 0$  for  $i = 1, 2$  imply that  $\theta_{11}$  and  $\theta_{22}$  are non-  
104 positive, ensuring the concavity of each utility function  $U_i$ .  
105 The equality constraints  $\mathbf{\Xi}_{i,-i} = -\frac{1}{2}$  for  $i = 1, 2$  indicate  
106 that  $\theta_{21} = \theta_{12} = 1$ , ensuring symmetry in the cross-effects  
107 between the firms. The conditions  $\mathbf{\Xi} \geq 0$  and  $\tilde{\mathbf{\Xi}} \geq 0$  ensure  
108 that both  $\mathbf{\Xi}$  and  $\tilde{\mathbf{\Xi}}$  are positive semidefinite matrices.

109 The loss function  $\ell_{\mathbf{\Xi}, \tilde{\mathbf{\Xi}}}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  defined in Definition 3 is  
110  $\max_{\mathbf{x}^j \in \mathcal{X}^j} \left\{ -\text{Tr}(\Psi^j \mathbf{\Xi}) + \text{Tr}(\tilde{\Psi}^j \tilde{\mathbf{\Xi}}) + \|\phi^j \odot \text{vec}(\mathbf{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))\|_2 \right\}$ ,  
111 where the vector  $\mathbf{v}^j = \phi^j \odot \text{vec}(\mathbf{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))$ , the  
112 matrices  $\Psi^j = \begin{bmatrix} v_1^j & 2v_2^j \\ 2v_5^j & v_6^j \end{bmatrix}$  and  $\tilde{\Psi}^j = \begin{bmatrix} v_3^j & v_4^j \\ v_7^j & v_8^j \end{bmatrix}$ .

Since the constraints  $\mathbf{\Xi}_{ii} \geq 0$  for all  $i = 1, 2$  can be  
rewritten as  $\text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \mathbf{\Xi}) \geq 0$ , for  $i = 1, 2$ . Constraints  
 $\mathbf{\Xi}_{i,-i} = -\frac{1}{2}$  for all  $i = 1, 2$  are equivalent to  $\text{Tr}(A \mathbf{\Xi}) = -1$ ,  
we can rewrite (28) as follows. 113  
114  
115  
116

$$\begin{aligned}
\min_{\mathbf{\Xi}, \tilde{\mathbf{\Xi}}} & \frac{1}{N} \sum_{j=1}^N \ell_{\mathbf{\Xi}, \tilde{\mathbf{\Xi}}}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) + \alpha \mathcal{R}(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}) \\
\text{s.t.} & \quad \text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \mathbf{\Xi}) \geq 0, \forall i = 1, 2, \quad \text{Tr}(A \mathbf{\Xi}) = -1, \\
& \quad \mathbf{\Xi} \geq 0, \quad \tilde{\mathbf{\Xi}} \geq 0. \tag{29}
\end{aligned}$$

In (29), matrix  $A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$  is an anti-diagonal matrix. The  
vector  $\mathbf{e}_i$  has a one in the  $i$ -th position and zeros in all other  
positions. The loss function  $\ell_{\mathbf{\Xi}, \tilde{\mathbf{\Xi}}}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  is expressed as: 117  
118  
119

$$\max_{\mathbf{x}^j \in \mathcal{X}^j} \left\{ -\text{Tr}(\Psi_s^j \mathbf{\Xi}) + \text{Tr}(\tilde{\Psi}_s^j \tilde{\mathbf{\Xi}}) + \|\phi^j \odot \text{vec}(\mathbf{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))\|_2 \right\}, \tag{120}$$

$$\text{where } \Psi_s^j = \begin{bmatrix} v_1^j & v_2^j + v_5^j \\ v_2^j + v_5^j & v_6^j \end{bmatrix}, \tilde{\Psi}_s^j = \begin{bmatrix} v_3^j & \frac{v_4^j + v_7^j}{2} \\ \frac{v_4^j + v_7^j}{2} & v_8^j \end{bmatrix}. \tag{121}$$

□ 122

## 5 Proof of Proposition 1 123

*Proof.* First, we rewrite the Lagrangian of the primal problem  
(14) in Lemma 1 of Section 4.5 as follows: 124  
125

$$\begin{aligned}
\mathcal{L}(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}, \boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi}) & = \frac{1}{N} \sum_{j=1}^N \ell_{\mathbf{\Xi}, \tilde{\mathbf{\Xi}}}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) + \alpha \mathcal{R}(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}) - \\
& \sum_{i=1}^2 \lambda_i \text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \mathbf{\Xi}) + \nu (\text{Tr}(A \mathbf{\Xi}) + 1) - \text{Tr}(\Xi \mathbf{\Xi}) - \text{Tr}(\tilde{\Xi} \tilde{\mathbf{\Xi}}). \tag{30}
\end{aligned}$$

Here,  $\boldsymbol{\lambda}$ ,  $\nu$ ,  $\Xi$ , and  $\tilde{\Xi}$  represent the dual variables. The terms  
 $\text{Tr}(\Xi \mathbf{\Xi})$  and  $\text{Tr}(\tilde{\Xi} \tilde{\mathbf{\Xi}})$  enforce the constraints  $\mathbf{\Xi} \geq 0$  and  
 $\tilde{\mathbf{\Xi}} \geq 0$ , respectively. The regularization term is defined by 126  
127

the Frobenius norm, i.e.,  $\mathcal{R}(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}) = \frac{1}{2} (\|\mathbf{\Xi}\|_F^2 + \|\tilde{\mathbf{\Xi}}\|_F^2)$ . 128  
129

For ease of presentation, we denote  $h(\mathbf{x}^j, \hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) :=$   
 $\|\phi^j \odot \text{vec}(\mathbf{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))\|_2$ , and the maximizer  $\bar{\mathbf{x}}^j =$   
 $\arg \max_{\mathbf{x}^j \in \mathcal{X}^j} \left\{ -\text{Tr}(\Psi_s^j \mathbf{\Xi}) + \text{Tr}(\tilde{\Psi}_s^j \tilde{\mathbf{\Xi}}) + h(\mathbf{x}^j, \hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) \right\}$ . 130  
131  
132

Let the symmetric matrices  $\Upsilon = \frac{1}{N} \sum_{j=1}^N \Psi_s^j(\bar{\mathbf{x}}^j)$  and  
 $\tilde{\Upsilon} = \frac{1}{N} \sum_{j=1}^N \tilde{\Psi}_s^j(\bar{\mathbf{x}}^j)$ . We can then rewrite the Lagrangian  
 $\mathcal{L}(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}, \boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi})$ , as given in equation (30), as follows 133  
134  
135

$$\begin{aligned}
\mathcal{L} & = \frac{\alpha}{2} \|\mathbf{\Xi}\|_F^2 - \langle \Upsilon + \sum_{i=1}^2 \lambda_i (\mathbf{e}_i \mathbf{e}_i^\top) + \Xi - \nu A, \mathbf{\Xi} \rangle \\
& + \frac{\alpha}{2} \|\tilde{\mathbf{\Xi}}\|_F^2 + \langle \tilde{\Upsilon} - \tilde{\Xi}, \tilde{\mathbf{\Xi}} \rangle + \frac{1}{N} \sum_{j=1}^N h(\bar{\mathbf{x}}^j, \hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) + \nu. \tag{31}
\end{aligned}$$

136 Based on (31), we derive the first order conditions as

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathbf{\Xi}} &= \alpha \mathbf{\Xi} - \Upsilon - \sum_{i=1}^2 \lambda_i (\mathbf{e}_i \mathbf{e}_i^\top) - \Xi + \nu A = \mathbf{0}, \\ \frac{\partial \mathcal{L}}{\partial \tilde{\mathbf{\Xi}}} &= \alpha \tilde{\mathbf{\Xi}} + \tilde{\Upsilon} - \tilde{\Xi} = \mathbf{0}. \end{aligned} \quad (32)$$

137 Next, we derive the Lagrange dual problem for the primal  
138 problem (14) in Lemma 1 of Section 4.5. We begin by writing  
139 the corresponding Lagrange dual function as follows.

$$\mathcal{G}(\boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi}) = \inf_{\mathbf{\Xi}, \tilde{\mathbf{\Xi}}} \mathcal{L}(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}, \boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi}).$$

140 By substituting (32) into (31), we can derive  $\mathcal{G}(\boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi})$   
141 as follows. First, we express  $\frac{\alpha}{2} \|\mathbf{\Xi}\|_F^2$  and  $\frac{\alpha}{2} \|\tilde{\mathbf{\Xi}}\|_F^2$  as:

$$\begin{aligned} \frac{\alpha}{2} \|\mathbf{\Xi}\|_F^2 &= \frac{\alpha}{2} \text{Tr}(\mathbf{\Xi}^\top \mathbf{\Xi}) = \frac{1}{2\alpha} \left\| \Upsilon + \sum_{i=1}^2 \lambda_i (\mathbf{e}_i \mathbf{e}_i^\top) + \Xi - \nu A \right\|_F^2, \\ \frac{\alpha}{2} \|\tilde{\mathbf{\Xi}}\|_F^2 &= \frac{\alpha}{2} \text{Tr}(\tilde{\mathbf{\Xi}}^\top \tilde{\mathbf{\Xi}}) = \frac{1}{2\alpha} \left\| -\tilde{\Upsilon} + \tilde{\Xi} \right\|_F^2. \end{aligned}$$

142 Based on the above equations of  $\frac{\alpha}{2} \|\mathbf{\Xi}\|_F^2$  and  $\frac{\alpha}{2} \|\tilde{\mathbf{\Xi}}\|_F^2$ , we can  
143 further derive the corresponding dual problem as

$$\begin{aligned} \max_{\boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi}} \quad & \mathcal{G}(\boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi}) \\ \text{s.t.} \quad & \lambda_i \geq 0, \forall i = 1, 2, \\ & \Xi \geq 0, \tilde{\Xi} \geq 0, \end{aligned} \quad (33)$$

144 where  $\mathcal{G}(\boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi})$  is the following equation:

$$\begin{aligned} \mathcal{G}(\boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi}) &= -\frac{1}{2\alpha} \left\| \Upsilon + \sum_{i=1}^2 \lambda_i (\mathbf{e}_i \mathbf{e}_i^\top) + \Xi - \nu A \right\|_F^2 \\ &\quad - \frac{1}{2\alpha} \left\| -\tilde{\Upsilon} + \tilde{\Xi} \right\|_F^2 + \frac{1}{N} \sum_{j=1}^N h(\bar{\mathbf{x}}^j, \hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) + \nu. \end{aligned}$$

145 To employ the primal-dual interior-point algorithm, we  
146 need to derive the corresponding perturbed KKT conditions.  
147 The basic idea is to utilize a logarithmic barrier function to  
148 penalize the constraints in both the primal and dual problems.  
149 Specifically, using the logarithmic barrier, we first approxi-  
150 mate the primal problem (14) in Lemma 1 as follows.

$$\begin{aligned} \min_{\mathbf{\Xi}, \tilde{\mathbf{\Xi}}} \quad & \frac{\alpha}{2} \|\mathbf{\Xi}\|_F^2 - \langle \Upsilon, \mathbf{\Xi} \rangle + \frac{\alpha}{2} \|\tilde{\mathbf{\Xi}}\|_F^2 + \langle \tilde{\Upsilon}, \tilde{\mathbf{\Xi}} \rangle + \frac{1}{N} \sum_{j=1}^N h(\bar{\mathbf{x}}^j, \hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) \\ & - \frac{1}{\mu} \left( \log \det(\mathbf{\Xi}) + \log \det(\tilde{\mathbf{\Xi}}) + \sum_{i=1}^2 \log(\text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \mathbf{\Xi})) \right) \\ \text{s.t.} \quad & \text{Tr}(A\mathbf{\Xi}) = -1. \end{aligned} \quad (34)$$

151 Here,  $\mu$  represents the logarithmic barrier parameter. By em-  
152 ploying the penalized barrier function, the inequality con-  
153 straints of the original problem are seamlessly integrated into  
154 the objective function, as shown in (34).

Let  $\mathcal{G} = \mathcal{G}(\boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi})$  as defined in (33). By applying the  
logarithmic barrier, (33) can also be approximated as

$$\max_{\boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi}} \mathcal{G} + \frac{1}{\mu} \left( \sum_{i=1}^2 \log \lambda_i + \log(\det(\Xi)) + \log(\det(\tilde{\Xi})) \right). \quad (35)$$

Next, we derive the perturbed KKT conditions based on  
(34) and (35). We start by expressing the Lagrangian  
 $\mathcal{L}_\mu(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}, \nu)$  for the primal barrier problem (34) as follows.

$$\begin{aligned} \mathcal{L}_\mu(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}, \nu) &= \frac{\alpha}{2} \|\mathbf{\Xi}\|_F^2 - \langle \Upsilon, \mathbf{\Xi} \rangle + \frac{\alpha}{2} \|\tilde{\mathbf{\Xi}}\|_F^2 + \langle \tilde{\Upsilon}, \tilde{\mathbf{\Xi}} \rangle \\ &\quad - \frac{1}{\mu} \left( \log \det(\mathbf{\Xi}) + \log \det(\tilde{\mathbf{\Xi}}) + \sum_{i=1}^2 \log(\text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \mathbf{\Xi})) \right) \\ &\quad + \nu (\text{Tr}(A\mathbf{\Xi}) + 1) + \frac{1}{N} \sum_{j=1}^N h(\bar{\mathbf{x}}^j, \hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j). \end{aligned}$$

Setting  $\frac{\partial \mathcal{L}_\mu(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}, \nu)}{\partial \mathbf{\Xi}} = \mathbf{0}$ ,  $\frac{\partial \mathcal{L}_\mu(\mathbf{\Xi}, \tilde{\mathbf{\Xi}}, \nu)}{\partial \tilde{\mathbf{\Xi}}} = \mathbf{0}$ , we can obtain

$$\begin{aligned} \alpha \mathbf{\Xi} - \Upsilon - \frac{1}{\mu} \left( \mathbf{\Xi}^{-1} + \sum_{i=1}^2 \frac{\mathbf{e}_i \mathbf{e}_i^\top}{\text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \mathbf{\Xi})} \right) + \nu A^\top &= \mathbf{0}, \\ \alpha \tilde{\mathbf{\Xi}} + \tilde{\Upsilon} - \frac{1}{\mu} (\tilde{\mathbf{\Xi}}^{-1}) &= \mathbf{0}. \end{aligned} \quad (36)$$

Let  $\mathcal{G}_\mu(\boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi})$  represent the objective function in prob-  
lem (35). By setting  $\frac{\partial \mathcal{G}_\mu(\boldsymbol{\lambda}, \nu, \Xi, \tilde{\Xi})}{\partial \Xi} = \mathbf{0}$ , we then have

$$-\frac{1}{\alpha} \left( \Upsilon + \sum_{i=1}^2 \lambda_i (\mathbf{e}_i \mathbf{e}_i^\top) + \Xi - \nu A \right) + \frac{1}{\mu} \Xi^{-1} = \mathbf{0}. \quad (37)$$

Since the primal and dual variables that satisfy the KKT con-  
ditions are optimal solutions, they must satisfy all the equa-  
tions in (32), (36) and (37). Accordingly, we can further de-  
rive the following three equations.

$$\mathbf{\Xi} = \frac{1}{\mu} \Xi^{-1}, \tilde{\mathbf{\Xi}} = \frac{1}{\mu} \tilde{\Xi}^{-1}, \lambda_i = \frac{1}{\mu \text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \mathbf{\Xi})}, \forall i = 1, 2. \quad (38)$$

Hence, we derive the equations in Proposition 1, which stem  
from (32) and (38). Specifically,  $\text{Tr}((\mathbf{e}_i \mathbf{e}_i^\top) \mathbf{\Xi}^*) \geq 0$  for all  
 $i = 1, 2$ ,  $\text{Tr}(A\mathbf{\Xi}^*) = -1$ , and  $\mathbf{\Xi}^*, \tilde{\mathbf{\Xi}}^* \geq 0$  represent primal  
feasibility. Additionally,  $\lambda_i^* \geq 0, \forall i = 1, 2$ , and  $\Xi^*, \tilde{\Xi}^* \geq 0$   
ensure dual feasibility. The equations in (32) and (38) corre-  
spond to stationarity and complementarity, respectively.  $\square$

## 6 Newton Updates in Algorithm 2

In the primal-dual interior-point method, a sequence of it-  
erates is generated that approximates the central path, ulti-  
mately converging to the optimal primal and dual solutions.  
In this context, the basic iterative step is derived using New-  
ton's method. Ideally, each Newton step updates the points,  
progressively refining the solution until the optimal points for

181 both the primal and dual problems are found. The detailed  
182 Newton updates are presented as follows.

183 For problem (14) in Lemma 1, points  $(\bar{\alpha}, \bar{\beta}, \bar{\gamma}, \bar{\delta}, \bar{\lambda}, \nu)$  on  
184 the central path satisfy the nonlinear equations in Proposition  
185 1. To find such points, the essence of the Newton step lies in  
186 linearizing these nonlinear equations along the central path.  
187 We begin by listing these equations as follows.

$$\begin{aligned} \text{Tr}(A\bar{\alpha}) + 1 &= 0. \\ \lambda_i \text{Tr}((e_i e_i^\top) \bar{\alpha}) - \frac{1}{\mu} &= 0, \forall i = 1, 2. \\ \bar{\beta} \bar{\alpha} - \frac{1}{\mu} \mathbf{I} &= \mathbf{0}, \\ \bar{\delta} \bar{\alpha} - \frac{1}{\mu} \mathbf{I} &= \mathbf{0}. \\ \alpha \bar{\alpha} - \bar{\beta} + \frac{1}{N} \sum_{j=1}^N \tilde{\Psi}_s^j(\bar{x}^j) &= \mathbf{0}. \\ \alpha \bar{\alpha} - \sum_{i=1}^2 \lambda_i (e_i e_i^\top) - \bar{\beta} + \nu A &= \frac{1}{N} \sum_{j=1}^N \Psi_s^j(\bar{x}^j). \end{aligned}$$

188 To solve these equations, we seek to find a zero of the vector-  
189 valued function  $\mathbf{F}(\bar{\alpha}, \bar{\beta}, \bar{\gamma}, \bar{\delta}, \bar{\lambda}, \nu)$ , which is defined as

$$\mathbf{F}(\bar{\alpha}, \bar{\beta}, \bar{\gamma}, \bar{\delta}, \bar{\lambda}, \nu) = \begin{pmatrix} \text{Tr}(A\bar{\alpha}) + 1 \\ \lambda_1 \text{Tr}((e_1 e_1^\top) \bar{\alpha}) - \frac{1}{\mu} \\ \lambda_2 \text{Tr}((e_2 e_2^\top) \bar{\alpha}) - \frac{1}{\mu} \\ \bar{\beta} \bar{\alpha} - \frac{1}{\mu} \mathbf{I} \\ \bar{\delta} \bar{\alpha} - \frac{1}{\mu} \mathbf{I} \\ \alpha \bar{\alpha} - \sum_{i=1}^2 \lambda_i (e_i e_i^\top) - \bar{\beta} - \frac{1}{N} \sum_{j=1}^N \Psi_s^j(\bar{x}^j) + \nu A \\ \alpha \bar{\alpha} - \bar{\beta} + \frac{1}{N} \sum_{j=1}^N \tilde{\Psi}_s^j(\bar{x}^j) \end{pmatrix}.$$

190 However, solving this problem directly is challenging, so we  
191 employ Newton's method to first linearize it and then solve  
192 it approximately. Specifically, we aim to find its Jacobian  
193  $J(\bar{\alpha}, \bar{\beta}, \bar{\gamma}, \bar{\delta}, \bar{\lambda}, \nu)$ , and then iteratively solve the following  
194 system for update direction  $(\Delta\bar{\alpha}, \Delta\bar{\beta}, \Delta\bar{\gamma}, \Delta\bar{\delta}, \Delta\bar{\lambda}, \Delta\nu)$ :

$$J(\bar{\alpha}, \bar{\beta}, \bar{\gamma}, \bar{\delta}, \bar{\lambda}, \nu) \begin{pmatrix} \Delta\bar{\alpha} \\ \Delta\bar{\beta} \\ \Delta\bar{\gamma} \\ \Delta\bar{\delta} \\ \Delta\bar{\lambda} \\ \Delta\nu \end{pmatrix} = -\mathbf{F}(\bar{\alpha}, \bar{\beta}, \bar{\gamma}, \bar{\delta}, \bar{\lambda}, \nu).$$

By finding the Jacobian, we derive the following system:

$$\begin{pmatrix} \langle A, \Delta\bar{\alpha} \rangle \\ \lambda_1 \langle e_1 e_1^\top, \Delta\bar{\alpha} \rangle + \text{Tr}((e_1 e_1^\top) \bar{\alpha}) \Delta\lambda_1 \\ \lambda_2 \langle e_2 e_2^\top, \Delta\bar{\alpha} \rangle + \text{Tr}((e_2 e_2^\top) \bar{\alpha}) \Delta\lambda_2 \\ \bar{\beta} \Delta\bar{\alpha} + \Delta\bar{\beta} \bar{\alpha} \\ \bar{\delta} \Delta\bar{\alpha} + \Delta\bar{\delta} \bar{\alpha} \\ \alpha \Delta\bar{\alpha} - \Delta\bar{\beta} - (e_1 e_1^\top) \Delta\lambda_1 - (e_2 e_2^\top) \Delta\lambda_2 + A \Delta\nu \\ \alpha \Delta\bar{\alpha} - \Delta\bar{\beta} \end{pmatrix} = -\mathbf{F}. \quad (39)$$

Next, we solve the equation in (39) to derive the Newton up-  
update directions, i.e.,  $(\Delta\bar{\alpha}, \Delta\bar{\beta}, \Delta\bar{\gamma}, \Delta\bar{\delta}, \Delta\bar{\lambda}, \Delta\nu)$ . These  
directions are subsequently utilized to iteratively refine the  
primal and dual variables during each step of Algorithm 2.

## 7 Multi-Player Bertrand-Nash Competition

We first explain the constraint  $\theta_{12} = \theta_{21} = 1$  in Lemma 1 for  
the two-firm Bertrand-Nash competition. This constraint ac-  
counts for the flexibility of the utility parameters. To be more  
specific, let  $\theta_i^o$  be the unknown parameters for all  $i = 1, 2$ .  
In this two-firm competition, it is common for  $\theta_{12}^o$  to differ  
from  $\theta_{21}^o$ . However, as shown in the proof of Lemma 1, the  
matrix  $\nabla \mathbf{F}$  needs to be symmetric to formulate the semidefinite  
programming problem. To ensure symmetry, we normal-  
ize each parameter vector  $\theta_i^o$  by dividing it by  $\theta_{i,-i}^o$ , where  
 $\theta_{i,-i}^o = \theta_{1,2}^o$  if  $i = 1$ , and  $\theta_{i,-i}^o = \theta_{2,1}^o$  if  $i = 2$ . This nor-  
malization operator does not change the corresponding equi-  
librium solutions. Consequently, the new parameters satisfy  
 $\theta_{12} = \theta_{21} = 1$ . Therefore, the parameters  $\theta_i$  in Section 4.5  
can be interpreted as those derived from the original param-  
eters  $\theta_i^o$ , where  $\theta_{1,2}^o \neq \theta_{2,1}^o$ .

Next, we provide the proofs for the semidefinite program-  
ming formulation in three-firm Bertrand-Nash competitions.  
We first present their parameter structures. Specifically, the  
linear demand function  $U_i$  for  $i = 1, 2, 3$  is defined as

$$U_i(\mathbf{x}, \mathbf{s}; \theta_i^o) = \theta_{i1}^o x_1 + \theta_{i2}^o x_2 + \theta_{i3}^o x_3 + \theta_{i4}^o s + \theta_{i5}^o. \quad (40)$$

Here,  $\theta_{i,i}^o \leq 0$ , and  $\theta_{i,-i}^o \geq 0$ , where, for instance, if  $i = 1$ ,  
then the vector  $\theta_{i,-i}^o = (\theta_{1,2}^o, \theta_{1,3}^o)$ .

Let each parameter vector  $\theta_i^o = (\theta_{i1}^o, \theta_{i2}^o, \theta_{i3}^o, \theta_{i4}^o, \theta_{i5}^o) \in \mathbb{R}^5$ . We can derive its Jacobian matrix  $\nabla F(\mathbf{x})$  as

$$\nabla \mathbf{F}(\mathbf{x}) = - \begin{bmatrix} 2\theta_{11}^o & \theta_{12}^o & \theta_{13}^o \\ \theta_{21}^o & 2\theta_{22}^o & \theta_{23}^o \\ \theta_{31}^o & \theta_{32}^o & 2\theta_{33}^o \end{bmatrix}.$$

To ensure the symmetry of  $\nabla \mathbf{F}(\mathbf{x})$ , we construct each  $\theta_i$  by  
multiplying each row by a non-negative number  $a_1, a_2, a_3$ :

$$\begin{bmatrix} a_1 * 2\theta_{11}^o & a_1 * \theta_{12}^o & a_1 * \theta_{13}^o \\ a_2 * \theta_{21}^o & a_2 * 2\theta_{22}^o & a_2 * \theta_{23}^o \\ a_3 * \theta_{31}^o & a_3 * \theta_{32}^o & a_3 * 2\theta_{33}^o \end{bmatrix}.$$

If we require this matrix to be symmetric, we need to impose  
three constraints, one for each off-diagonal element. This re-

228 sults in the following system of three equations:

$$\begin{cases} a_1 * \theta_{12}^o = a_2 * \theta_{21}^o; \\ a_1 * \theta_{13}^o = a_3 * \theta_{31}^o; \\ a_2 * \theta_{23}^o = a_3 * \theta_{32}^o. \end{cases} \quad (41)$$

229 In (41), there are three equations with three unknowns, i.e.,  
230  $a_1$ ,  $a_2$ , and  $a_3$ . This system typically has a solution, except  
231 in degenerate cases such as when  $\theta_i^o = 0$ . Once the values  
232 of  $a_1$ ,  $a_2$ , and  $a_3$  are determined, we can construct  $\theta_i$  for  
233 all  $i = 1, 2, 3$ , for example,  $\theta_{12} = a_1 * \theta_{12}^o$ . The resulting  
234 Jacobian  $\nabla F(\mathbf{x})$  will be symmetric. Then, we can formulate  
235 the estimation of all  $\theta_i$  as a similar SDP as in Lemma 1.

236 In a four-firm Bertrand competition, the Jacobian  $\nabla F(\mathbf{x})$   
237 will be a  $4 \times 4$  matrix. Enforcing symmetry requires six con-  
238 straints, but only four coefficients, i.e.,  $a_1$ ,  $a_2$ ,  $a_3$ , and  $a_4$ , are  
239 available. This issue also arises with more than four play-  
240 ers. In such cases, an additional constraint must be imposed,  
241 requiring the matrix of parameters to be symmetric from the  
242 outset. This will restrict the search to symmetric matrices.

## 243 8 Our Methods for Cournot Competition

244 In the Cournot competition, each parameter vector  $\theta_i$  can be  
245 represented in the form  $(a, b, d, c_i) \in \mathbb{R}^4$ . Given that some  
246 elements are shared across each  $\theta_i$ , we redefine the parameter  
247 vector  $\vartheta$  and the corresponding loss function as follows.

248 Let  $\vartheta = (a, b, d, c_1, c_2, \dots, c_p) \in \mathbb{R}^{p+3}$  be the parameter  
249 vector to be estimated. In the definition of consistent param-  
250 eter vectors  $\mathbb{C}$  in Section 4.2, we can further express vectors  
251  $\Phi_{\vartheta}^j \in \mathbb{R}^p$  and  $\phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) \in \mathbb{R}^{p+3}$  as follows.

$$\Phi_{\vartheta}^j = (\theta^{\top} \phi_1(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j), \theta^{\top} \phi_2(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j), \dots, \theta^{\top} \phi_p(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j))^{\top};$$

$$\phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) = \left( -\left( \sum_k \hat{x}_k^j + \hat{x}_i^j \right), 1, \hat{s}^j, 0, \dots, -1, \dots, 0 \right)^{\top}.$$

252 According to the proof of Theorem 1, we derive that com-  
253 puting the incenter of the set  $\mathbb{C}$  is equivalent to solving the  
254 following convex optimization problem:  
255

$$\begin{aligned} \min_{\vartheta} \quad & \|\vartheta\|_2 \\ \text{s.t.} \quad & \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + \left\| \sum_{i=1}^p \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) (\mathbf{x}_i^j - \hat{\mathbf{x}}_i^j) \right\|_2 \leq 0, \\ & \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N]. \end{aligned} \quad (42)$$

256 The main difference from Theorem 1 is the inequality con-  
257 straint, which stems from a different definition of  $\vartheta$ . Corre-  
258 spondingly, we define loss function  $\ell_{\vartheta}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  of  $\vartheta$  in this  
259 competition, similar to the one in Definition 3, as

$$\max_{\mathbf{x}^j \in \mathcal{X}^j} \left\{ \langle \Phi_{\vartheta}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + \left\| \sum_{i=1}^p \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) (\mathbf{x}_i^j - \hat{\mathbf{x}}_i^j) \right\|_2 \right\}.$$

260 Based on this loss function, we then leverage Algorithm 1 in  
261 Section 4.4 to estimate  $\vartheta$  for the Cournot competition.

262 Next, we present the prior incorporation of monotonicity of  
263  $F$  using the semidefinite programming framework. Specifi-

cally, in this Cournot game, Jacobian  $\nabla F(\mathbf{x})$

$$\nabla F(\mathbf{x}) = - \begin{bmatrix} -2a & -a & \dots & -a \\ -a & -2a & \dots & -a \\ \vdots & \vdots & \ddots & \vdots \\ -a & -a & \dots & -2a \end{bmatrix}$$

is a symmetric matrix.

$$\text{Let } \sqsupset = \begin{bmatrix} 2a & a & \dots & a \\ a & 2a & \dots & a \\ \vdots & \vdots & \ddots & \vdots \\ a & a & \dots & 2a \end{bmatrix}, \Lambda_1 = \begin{bmatrix} c_1 & 0 & \dots & 0 \\ 0 & c_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & c_p \end{bmatrix},$$

and  $\Lambda_2 = \begin{bmatrix} b & 0 \\ 0 & d \end{bmatrix}$ , estimating  $\vartheta$  can be reformulated as

$$\begin{aligned} \min_{\sqsupset, \Lambda_1, \Lambda_2} \quad & \frac{1}{N} \sum_{j=1}^N \ell_{\sqsupset, \Lambda_1, \Lambda_2}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) + \alpha \mathcal{R}(\sqsupset, \Lambda_1, \Lambda_2) \\ \text{s.t.} \quad & \sqsupset \geq 0, \Lambda_1 \geq 0, \Lambda_2 \geq 0. \end{aligned} \quad (43)$$

Here, loss function  $\ell_{\sqsupset, \Lambda_1, \Lambda_2}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j)$  is defined as the form:

$$\begin{aligned} \ell_{\sqsupset, \Lambda_1, \Lambda_2}(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) = \max_{\mathbf{x}^j \in \mathcal{X}^j} \left\{ \frac{1}{2p} \text{Tr}(\Psi_0^j \sqsupset) + \text{Tr}(\Psi_1^j \Lambda_1) \right. \\ \left. + \text{Tr}(\Psi_2^j \Lambda_2) + \left\| \sum_{i=1}^p \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) (\mathbf{x}_i^j - \hat{\mathbf{x}}_i^j) \right\|_2 \right\}, \end{aligned}$$

where diagonal matrices  $\Psi_0^j = \text{diag}(v_1^j, \dots, v_1^j) \in \mathbb{R}^{p \times p}$ ,

$\Psi_1^j = \text{diag}(v_4^j, \dots, v_{p+3}^j) \in \mathbb{R}^{p \times p}$ ,  $\Psi_2^j = \text{diag}(v_2^j, v_3^j) \in$

$\mathbb{R}^2$ , and the vector  $\mathbf{v}^j = \sum_{i=1}^p \phi_i(\hat{\mathbf{x}}^j, \hat{\mathbf{s}}^j) (\mathbf{x}_i^j - \hat{\mathbf{x}}_i^j)$ .

To solve the problem (43), we leverage the primal-dual  
interior-point approach as in Section 4.5. We begin by deriv-  
ing the perturbed KKT conditions in this game as follows.

$$\sqsupset^*, \Xi^* \geq 0; \quad \alpha \sqsupset + \frac{1}{2pN} \sum_{j=1}^N \Psi_0^j(\bar{\mathbf{x}}^j) - \Xi = \mathbf{0}.$$

$$\Lambda_1^*, \Lambda_2^* \geq 0; \quad \alpha \Lambda_1 + \frac{1}{N} \sum_{j=1}^N \Psi_1^j(\bar{\mathbf{x}}^j) - \Gamma_1 = \mathbf{0}.$$

$$\Gamma_1^*, \Gamma_2^* \geq 0; \quad \alpha \Lambda_2 + \frac{1}{N} \sum_{j=1}^N \Psi_2^j(\bar{\mathbf{x}}^j) - \Gamma_2 = \mathbf{0}.$$

$$\Gamma_1^* \Lambda_1^* - \frac{1}{\mu} \mathbf{I} = \mathbf{0}; \quad \Xi^* \sqsupset^* - \frac{1}{\mu} \mathbf{I} = \mathbf{0}; \quad \Gamma_2^* \Lambda_2^* - \frac{1}{\mu} \mathbf{I} = \mathbf{0}.$$

Here,  $\sqsupset^*$ ,  $\Lambda_1^*$ , and  $\Lambda_2^*$  are the optimal solutions to the primal  
problem (43).  $\Xi^*$ ,  $\Gamma_1^*$ , and  $\Gamma_2^*$  are the optimal solutions to the  
corresponding dual problem. And  $\mu$  is the barrier parameter.

To solve the above equations, we apply Newton's method  
by finding Jacobian  $J(\sqsupset, \Lambda_1, \Lambda_2, \Xi, \Gamma_1, \Gamma_2)$ , and repeatedly  
solving the system for  $(\Delta \sqsupset, \Delta \Lambda_1, \Delta \Lambda_2, \Delta \Xi, \Delta \Gamma_1, \Delta \Gamma_2)$ :

$$J(\sqsupset, \Lambda_1, \Lambda_2, \Xi, \Gamma_1, \Gamma_2) \begin{pmatrix} \Delta \sqsupset \\ \Delta \Lambda_1 \\ \Delta \Lambda_2 \\ \Delta \Xi \\ \Delta \Gamma_1 \\ \Delta \Gamma_2 \end{pmatrix} = -F(\sqsupset, \Lambda_1, \Lambda_2, \Xi, \Gamma_1, \Gamma_2).$$

---

**Algorithm 1** Primal-Dual Interior-Point for Problem (43)

```

1: function PD-IP( $\widehat{\mathcal{D}}, \alpha, \epsilon, \Xi^0, \Lambda_1^0, \Lambda_2^0, \Xi^0, \Gamma_1^0, \Gamma_2^0$ ).
2:    $\frac{1}{\mu^0} \leftarrow \frac{\text{Tr}(\Xi^0) + \text{Tr}(\Lambda_1^0 \Gamma_1^0)}{3p} + \frac{\text{Tr}(\Lambda_2^0 \Gamma_2^0)}{6}$ ;
3:    $k \leftarrow 0$ .
4:   while  $\frac{1}{\mu^k} > \epsilon$  do
5:     Compute  $\Psi_0^j(\bar{\mathbf{x}}_k^j)$ ,  $\Psi_1^j(\bar{\mathbf{x}}_k^j)$ , and  $\Psi_2^j(\bar{\mathbf{x}}_k^j)$ ;
6:     Compute  $(\Delta \Xi^k, \Delta \Lambda_1^k, \Delta \Lambda_2^k, \Delta \Xi^k, \Delta \Gamma_1^k, \Delta \Gamma_2^k)$ ;
7:     Backtracking line search for step-sizes  $\eta_p^k, \eta_d^k$ ;
8:     Compute  $(\Xi^{k+1}, \Lambda_1^{k+1}, \Lambda_2^{k+1}, \Xi^{k+1}, \Gamma_1^{k+1}, \Gamma_2^{k+1})$ ;
9:     Compute  $\frac{1}{\mu^{k+1}}$ ;
10:     $k \leftarrow k + 1$ ;
11:  end while
12:  return  $(\Xi^k, \Lambda_1^k, \Lambda_2^k, \Xi^k, \Gamma_1^k, \Gamma_2^k)$ .
13: end function

```

---

Section 5.1, we derive the corresponding Jacobian matrix as

$$\nabla \mathbf{F}(\mathbf{x}) = \begin{bmatrix} \frac{\gamma}{(C_1)^\gamma} \theta_1^1 x_1^{\gamma-1} & 0 & \dots & 0 \\ 0 & \frac{\gamma}{(C_2)^\gamma} \theta_2^1 x_2^{\gamma-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{\gamma}{(C_{|\mathcal{A}|})^\gamma} \theta_{|\mathcal{A}|}^1 x_{|\mathcal{A}|}^{\gamma-1} \end{bmatrix}.$$

We observe that the matrix  $\nabla \mathbf{F}(\mathbf{x})$  is positive semidefinite when all elements are non-negative. Let variables  $\Lambda_0 =$

$$\begin{bmatrix} \theta_1^0 & 0 & \dots & 0 \\ 0 & \theta_2^0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \theta_{|\mathcal{A}|}^0 \end{bmatrix}, \text{ and } \Lambda_1 = \begin{bmatrix} \theta_1^1 & 0 & \dots & 0 \\ 0 & \theta_2^1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \theta_{|\mathcal{A}|}^1 \end{bmatrix}.$$

Note that the positive semidefiniteness of the variables  $\Lambda_0$  and  $\Lambda_1$  is equivalent to the positive semidefiniteness of  $\nabla \mathbf{F}(\mathbf{x})$ . Similar to Cournot games, we can reformulate our estimation problem using the proposed loss function as follows.

$$\min_{\Lambda_0, \Lambda_1} \frac{1}{N} \sum_{j=1}^N \ell_{\Lambda_0, \Lambda_1}(\hat{\mathbf{x}}^j) + \alpha \mathcal{R}(\Lambda_0, \Lambda_1) \quad (45)$$

$$\text{s.t. } \Lambda_0 \geq 0, \Lambda_1 \geq 0.$$

Here, the loss function  $\ell_{\Lambda_0, \Lambda_1}(\hat{\mathbf{x}}^j)$  is defined as

$$\ell_{\Lambda_0, \Lambda_1}(\hat{\mathbf{x}}^j) = \max_{\mathbf{x}^j \in \mathcal{X}^j} \left\{ -\text{Tr}(\Psi_0^j \Lambda_0) - \text{Tr}(\Psi_1^j \Lambda_1) + \|\phi^j \odot \text{vec}(\mathbf{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))\|_2 \right\},$$

where the matrices  $\Psi_0^j = \text{diag}(v_1^j, v_3^j, \dots, v_{2|\mathcal{A}|-1}^j) \in \mathbb{R}^{|\mathcal{A}| \times |\mathcal{A}|}$

$\Psi_1^j = \text{diag}(v_2^j, v_4^j, \dots, v_{2|\mathcal{A}|}^j) \in \mathbb{R}^{|\mathcal{A}| \times |\mathcal{A}|}$ , and

vector  $\mathbf{v}^j$  is defined as:  $\mathbf{v}^j = \phi^j \odot \text{vec}(\mathbf{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))$ .

To solve the problem (45), we employ the primal-dual interior-point approach. We begin by deriving the perturbed KKT conditions in this scenario. Let  $(\Lambda_0^*, \Lambda_1^*)$  and  $(\Gamma_0^*, \Gamma_1^*)$  be the primal and dual optimal solutions for problem (45), respectively. The optimality conditions for the corresponding logarithmic barrier centering problem are then given by

$$\Gamma_0^* \Lambda_0^* - \frac{1}{\mu} \mathbf{I} = \mathbf{0}; \quad \Gamma_1^* \Lambda_1^* - \frac{1}{\mu} \mathbf{I} = \mathbf{0}.$$

$$\Lambda_0^*, \Lambda_1^* \geq 0; \quad \alpha \Lambda_0^* - \frac{1}{N} \sum_{j=1}^N \Psi_0^j(\bar{\mathbf{x}}^j) - \Gamma_0^* = \mathbf{0}.$$

$$\Gamma_0^*, \Gamma_1^* \geq 0; \quad \alpha \Lambda_1^* - \frac{1}{N} \sum_{j=1}^N \Psi_1^j(\bar{\mathbf{x}}^j) - \Gamma_1^* = \mathbf{0}.$$

Next, we utilize Newton's method to solve the four equations, following a process similar to Cournot games. Algorithm 2 summarizes the whole approach for solving (45).

## 10 Numerical Experiments

### 10.1 Feasibility for Utility Estimation

This approach selects a  $\vartheta$  from the set of consistent parameter vectors, i.e.,  $\mathcal{C} \setminus \{\mathbf{0}\}$ , by solving the following feasibility

Here,  $\mathbf{F}$  is the vector-valued function composed of the six equations derived in the perturbed KKT condition. By finding the Jacobian  $J$ , we rewrite the above equation as follows.

$$\begin{pmatrix} \Xi \Delta \Xi + \Delta \Xi \Xi \\ \Gamma_1 \Delta \Lambda_1 + \Delta \Gamma_1 \Lambda_1 \\ \Gamma_2 \Delta \Lambda_2 + \Delta \Gamma_2 \Lambda_2 \\ \alpha \Delta \Xi - \Delta \Xi \\ \alpha \Delta \Lambda_1 - \Delta \Gamma_1 \\ \alpha \Delta \Lambda_2 - \Delta \Gamma_2 \end{pmatrix} = - \begin{pmatrix} \Xi \Xi - \frac{1}{\mu} \mathbf{I} \\ \Gamma_1 \Lambda_1 - \frac{1}{\mu} \mathbf{I} \\ \Gamma_2 \Lambda_2 - \frac{1}{\mu} \mathbf{I} \\ \alpha \Xi + \frac{1}{2pN} \sum_{j=1}^N \Psi_0^j(\bar{\mathbf{x}}^j) - \Xi \\ \alpha \Lambda_1 + \frac{1}{N} \sum_{j=1}^N \Psi_1^j(\bar{\mathbf{x}}^j) - \Gamma_1 \\ \alpha \Lambda_2 + \frac{1}{N} \sum_{j=1}^N \Psi_2^j(\bar{\mathbf{x}}^j) - \Gamma_2 \end{pmatrix}. \quad (44)$$

In Algorithm 1, we present the primal-dual interior-point algorithm for solving (43). In line 5, we compute the matrices  $\Psi_0^j(\bar{\mathbf{x}}_k^j)$ ,  $\Psi_1^j(\bar{\mathbf{x}}_k^j)$ , and  $\Psi_2^j(\bar{\mathbf{x}}_k^j)$  for each data point using current parameters. In line 6, the Newton update direction is determined. In line 7, we utilize the backtracking line search to compute current step sizes  $\eta_p^k$  and  $\eta_d^k$  for the primal and dual variables, respectively. In lines 8 to 9, the current variables and the barrier parameter are updated for the next iteration.

## 9 Our Methods for Traffic Games

In the traffic game of Section 5.1, let  $\vartheta = (\theta_1; \dots; \theta_{|\mathcal{A}|})$ . Given an observed Wardrop equilibrium  $\hat{\mathbf{x}}^j$ , we define the loss function  $\ell_\vartheta(\hat{\mathbf{x}}^j)$  of  $\vartheta$  as the following form.

$$\max_{\mathbf{x}^j \in \mathcal{X}^j} \left\{ -\langle \Phi_\vartheta^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle + \|\phi^j \odot \text{vec}(\mathbf{1}_n \otimes (\mathbf{x}^j - \hat{\mathbf{x}}^j))\|_2 \right\}.$$

This function is derived from Theorem 1, differing by a minus sign in the first term. Here,  $\Phi_\vartheta^j$ , and  $\phi^j$  are expressed as

$$\Phi_\vartheta^j = \left( \theta_1^\top \phi_1(\hat{\mathbf{x}}^j), \dots, \theta_{|\mathcal{A}|}^\top \phi_{|\mathcal{A}|}(\hat{\mathbf{x}}^j) \right)^\top \in \mathbb{R}^{|\mathcal{A}|}, \quad \phi^j = \left[ \phi_1^\top(\hat{\mathbf{x}}^j), \dots, \phi_{|\mathcal{A}|}^\top(\hat{\mathbf{x}}^j) \right]^\top \in \mathbb{R}^{2|\mathcal{A}|}.$$

Next, we formulate the estimation of  $\vartheta$  using the semidefinite programming framework. Under the cost function in

---

**Algorithm 2** Primal-Dual Interior-Point for Problem (45)

---

```

1: function PD-IP( $\widehat{\mathcal{D}}, \alpha, \epsilon, \Lambda_0^0, \Lambda_1^0, \Gamma_0^0, \Gamma_1^0$ ).
2:    $\frac{1}{\mu^0} \leftarrow \frac{\text{Tr}(\Lambda_0^0 \Gamma_0^0) + \text{Tr}(\Lambda_1^0 \Gamma_1^0)}{2p}$ ;
3:    $k \leftarrow 0$ .
4:   while  $\frac{1}{\mu^k} > \epsilon$  do
5:     Compute  $\Psi_0^j(\bar{\mathbf{x}}_k^j), \Psi_1^j(\bar{\mathbf{x}}_k^j)$  using  $\widehat{\mathcal{D}}, \Lambda_0^k$ , and  $\Lambda_1^k$ ;
6:     Compute  $(\Delta\Lambda_0^k, \Delta\Lambda_1^k, \Delta\Gamma_0^k, \Delta\Gamma_1^k)$ ;
7:     Backtracking line search for step-sizes  $\eta_p^k, \eta_d^k$ ;
8:     Compute  $(\Lambda_0^{k+1}, \Lambda_1^{k+1}, \Gamma_0^{k+1}, \Gamma_1^{k+1})$ ;
9:     Compute  $\frac{1}{\mu^{k+1}}$ ;
10:     $k \leftarrow k + 1$ ;
11:  end while
12:  return  $(\Lambda_0^k, \Lambda_1^k, \Gamma_0^k, \Gamma_1^k)$ .
13: end function

```

---

327 optimization problem (*feasibility program*):

$$\begin{aligned} \min_{\boldsymbol{\vartheta}} \quad & 0 \\ \text{s.t.} \quad & \langle \Phi_{\boldsymbol{\vartheta}}^j, \mathbf{x}^j - \hat{\mathbf{x}}^j \rangle \leq 0, \forall \mathbf{x}^j \in \mathcal{X}^j, \forall j \in [N]. \end{aligned} \quad (46)$$

328 Here, the constraints represent the variational inequalities that  
329 parameters  $\boldsymbol{\vartheta}$  should satisfy based on observed dataset  $\widehat{\mathcal{D}}$ . To  
330 implement this program, we discretize each  $\mathcal{X}^j$  to obtain dis-  
331 crete actions, especially when each  $\mathcal{X}^j$  is infinite.

## 332 10.2 Bertsimas for Utility Estimation

333 The **Bertsimas** estimation method is applicable to both  
334 two-firm Bertrand-Nash competition and three-firm Cournot  
335 games. In this subsection, we focus on the Bertrand compe-  
336 tition to illustrate the approach. While the approach can be  
337 similarly extended to Cournot games, the details are omitted  
338 here for brevity due to their similarity. Specifically, in the  
339 two-firm Bertrand-Nash competition, this approach targets to  
340 solve the following inverse optimization problem:

$$\begin{aligned} \min_{\boldsymbol{\vartheta}, \epsilon, \mathbf{y}} \quad & \|\epsilon\|_{\infty} \\ \text{s.t.} \quad & M_i(\hat{x}_1^j, \hat{x}_2^j, \hat{s}^j; \boldsymbol{\theta}_i) \leq y_i^j, \forall i = 1, 2, \forall j = 1, \dots, N, \\ & \sum_{i=1}^2 (\hat{x}^j y_i^j - \hat{x}_i^j M_i(\hat{x}_1^j, \hat{x}_2^j, \hat{s}^j; \boldsymbol{\theta}_i)) \leq \epsilon_j, j = 1, \dots, N, \\ & \mathbf{y}^j \geq \mathbf{0}, \forall j = 1, \dots, N, \\ & M_1(\hat{x}_1^j, \hat{x}_2^{\text{med}}, \hat{s}^{\text{med}}; \boldsymbol{\theta}_1) \geq M_1(\hat{x}_1^k, \hat{x}_2^{\text{med}}, \hat{s}^{\text{med}}; \boldsymbol{\theta}_1), \\ & \hat{x}_1^j \leq \hat{x}_1^k, \forall j, k = 1, \dots, N, \\ & M_2(\hat{x}_1^{\text{med}}, \hat{x}_2^j, \hat{s}^{\text{med}}; \boldsymbol{\theta}_2) \geq M_2(\hat{x}_1^{\text{med}}, \hat{x}_2^k, \hat{s}^{\text{med}}; \boldsymbol{\theta}_2), \\ & \hat{x}_2^j \leq \hat{x}_2^k, \forall j, k = 1, \dots, N, \\ & M_1(\hat{x}_1, \hat{x}_2^{\text{med}}, \hat{s}^{\text{med}}; \boldsymbol{\theta}_1) = M_1^*(\hat{x}_1, \hat{x}_2^{\text{med}}, \hat{s}^{\text{med}}; \boldsymbol{\theta}_1^*), \\ & M_2(\hat{x}_1^{\text{med}}, \hat{x}_2, \hat{s}^{\text{med}}; \boldsymbol{\theta}_2) = M_2^*(\hat{x}_1^{\text{med}}, \hat{x}_2, \hat{s}^{\text{med}}; \boldsymbol{\theta}_2^*). \end{aligned} \quad (47)$$

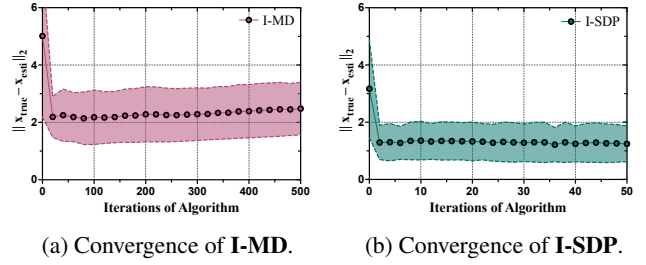


Figure 1: Convergence on the Cournot Testing Dataset.

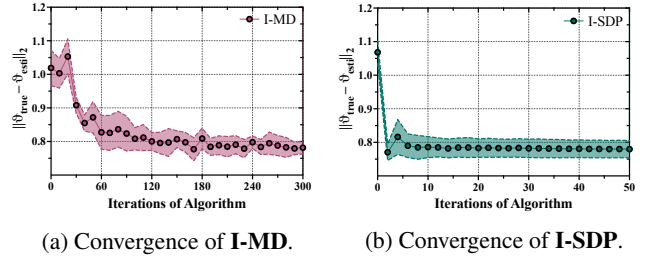


Figure 2: Convergence Results on the Traffic Dataset.

Here,  $\bar{x}$  represents the upper bound of the price,  $M_i$  denotes firm  $i$ 's marginal revenue function, and  $\boldsymbol{\theta}_i^*$  is its true parameter vector. Let  $\hat{s}^{\text{med}}$  be the median value of  $\hat{s}$  over the dataset. Breaking ties arbitrarily,  $\hat{s}^{\text{med}}$  corresponds to some observation  $j = j^{\text{med}}$ . Then,  $\hat{x}_1^{\text{med}}$  and  $\hat{x}_2^{\text{med}}$  are the corresponding prices at time  $j = j^{\text{med}}$ . The prices  $\hat{x}_1$  and  $\hat{x}_2$  represent the minimum prices observed in the dataset  $\widehat{\mathcal{D}}$ .

## 10.3 Implementation Details

We implement Algorithm 1 in Section 4.4 with  $\omega(\boldsymbol{\vartheta}) = \frac{1}{2} \|\boldsymbol{\vartheta}\|_2^2$ ,  $\mathcal{R}(\boldsymbol{\vartheta}) = \|\boldsymbol{\vartheta}\|_2$ , and  $\alpha = 0.01$ . For Algorithm 2 in Section 4.5, we set  $\epsilon = 0.001$  and  $\alpha = 0.01$ . All experiments are conducted on an Apple M1 Pro with 10-core CPU, 14-core GPU, 16-core Neural Engine, and 32GB of RAM. The datasets and codes are included with this material.

## 10.4 Convergence on Cournot Competition

In Figure 1, we depict the convergence performance of our **I-MD** and **I-SDP** methods on the Cournot testing dataset. The figures show the average error between the true equilibrium  $\mathbf{x}_{\text{true}}$  and the equilibrium  $\mathbf{x}_{\text{esti}}$  computed using  $\boldsymbol{\vartheta}_{\text{esti}}$ . Our **I-SDP** converges to lower errors with much smaller variances compared to **I-MD**, which can be attributed to the incorporation of priors and the primal-dual interior-point algorithm.

## 10.5 Convergence on Traffic Game

In Figure 2, we depict the convergence performance of our two methods, evaluated using the metric  $\|\boldsymbol{\vartheta}_{\text{esti}} - \boldsymbol{\vartheta}_{\text{true}}\|_2$  over 30 trials. In Figure 2a and Figure 2b, after fewer iterations, our **I-SDP** converges to a lower error than **I-MD**.