Inverse Matrix Games With Unique Quantal Response Equilibrium

Yue Yu, *Member, IEEE*, Jonathan Salfity, David Fridovich-Keil, *Member, IEEE*, and Ufuk Topcu, *Senior Member, IEEE*

***Abstract*—In an inverse game problem, one needs to infer the cost function of the players in a game such that a desired joint strategy is a Nash equilibrium. We study the inverse game problem for a class of multiplayer matrix games, where the cost perceived by each player is corrupted by random noise. We provide sufficient conditions for the players' quantal response equilibrium—a generalization of the Nash equilibrium to games with perception noise—to be unique. We develop efficient optimization algorithms for inferring the cost matrix based on semidefinite programs and bilevel optimization. We demonstrate the application of these methods in encouraging collision avoidance and fair resource allocation.**

***Index Terms*—Game theory, optimization.**

I. Introduction

IN A MULTIPLAYER game, each player tries to find the strategies with the minimum cost, where the cost of each strategy depends on the other players' strategies. The Nash equilibrium is a set of strategies where no player can benefit from unilaterally changing strategies. The Nash equilibrium generalizes minimax equilibrium in two-player zero-sum games [1] to multiplayer general-sum games [2], [3].

Given a joint strategy of the players in a game, the inverse game problem requires inferring the cost function such that the given joint strategy is indeed a Nash equilibrium. The inferred cost function can either rationalize observed player behavior [4], [5], [6] or provide incentives to encourage desired behavior [7], [8]. There have been many results on inverse games in different contexts, including specific games, such as matching [9], network formation [10], and auction [11]; and generic classes of games, such as succinct games [12], dynamic games [4], and general convex games whose Nash equilibria are characterized by variational inequalities [5].

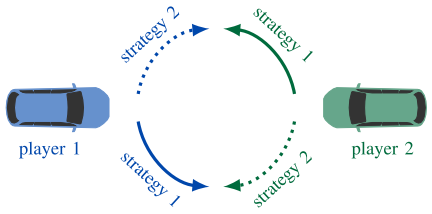


Fig. 1. A two-player game with two Nash equilibrium: joint strategy (1, 1) and (2, 2). The players must align their choice of equilibrium to avoid collisions.

The existing results on inverse games have the following limitations. First, the existing results do not guarantee a unique Nash equilibrium. Such nonuniqueness makes the players' behavior less predictable since there is ambiguity in which equilibrium the players will choose. It also complicates the players' decision-making, since the players need to align their choice of equilibrium with the other players' [13]; see Fig. 1 for an illustrative example. Second, the existing results assume each player has perfect perceptions of the cost of each action. Such an assumption is not reasonable to model human behavior where the players have bounded rationality and imperfect cost estimation [14], [15]. Recent work addressed these limitations in the context of two-player zero-sum games [6]. However, for multi-player general-sum games, treatments to these limitations are, to our best knowledge, still missing.

We study the inverse game problem for a class of multiplayer general-sum matrix games, where each player's strategy is a probability distribution over a finite number of discrete actions, the cost of a strategy is characterized by a matrix, and the cost perceived by each player is corrupted by random noise. Our contributions are as follows.

First, we provide sufficient conditions on the cost matrix for the uniqueness of the quantal response equilibrium—which is a generalization of the Nash equilibrium when the cost perceived by each player is corrupted by noise [14], [15]—and show that one can efficiently compute this unique equilibrium by solving a nonlinear least-squares problem. Second, we develop two numerical methods–one based on semidefinite programs, the other based on bilevel optimization–that infer the cost matrices which optimize the unique quantal response

Manuscript received 17 July 2022; revised 13 September 2022; accepted 28 September 2022. Date of publication 17 October 2022; date of current version 24 October 2022. The work of Yue Yu and Ufuk Topcu was supported by the National Science Foundation under Grant 1652113 and Grant 2211432. The work of Jonathan Salfity was supported by The University of Texas at Austin under the Creating Connections Grant and the Cockrell School of Engineering Doctoral Fellowship. The work of David Fridovich-Keil was supported by the National Science Foundation under Grant 2211548. Recommended by Senior Editor V. Ugrinovskii. *(Corresponding author: Yue Yu.)*

Yue Yu and Ufuk Topcu are with the Oden Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: yueyu@utexas.edu; utopcu@utexas.edu).

Jonathan Salfity is with the Department of Mechanical Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: j.salfity@utexas.edu).

David Fridovich-Keil is with the Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: dfk@utexas.edu).

Digital Object Identifier 10.1109/LCSYS.2022.3214857

equilibrium with respect to a performance function. The latter extends the implicit differentiation methods in [6] from convex-concave saddle point problems to nonconvex equilibrium problems. Finally, we demonstrate the application of these methods in encouraging collision avoidance and fair resource allocation.

Our results are particularly useful for designing incentives that motivate desired behavior when the players' cost function is (partially) known. They also have potential applications in general inverse multiplayer games.

Notation: We let $\mathbb{R}$, $\mathbb{R}_+$, $\mathbb{R}_{++}$, and $\mathbb{N}$ denote the set of real, nonnegative real, positive real, and positive integer numbers, respectively. Given $m, n \in \mathbb{N}$, we let $\mathbb{R}^n$ and $\mathbb{R}^{m\times n}$ denote the set of n-dimensional real vectors and $m \times n$ real matrices; we let $\mathbf{1}_n$ and I_n denote the n-dimensional vector of all 1's and the $n \times n$ identity matrix, respectively. Given positive integer $n \in \mathbb{N}$, we let $[n] \coloneqq \{1, 2, \dots, n\}$ denote the set of positive integers less or equal to n. Given $x \in \mathbb{R}^n$ and $k \in [n]$, we let $[x]_k$ denote the k-the element of vector x, and $\|x\|$ denote the ℓ_2-norm of x. Given a square real matrix $A \in \mathbb{R}^{n\times n}$, we let $A^\top$, A^{-1}, and $A^{-\top}$ denote the transpose, the inverse, and the transpose of the inverse of matrix A, respectively; we say $A \succeq 0$ and $A \succ 0$ if A is symmetric positive semidefinite and symmetric positive definite, respectively; we let $\|A\|_F$ denote the Frobenius norm of matrix A. We let $\mathrm{blkdiag}(A_1, \dots, A_k)$ denote the block diagonal matrix whose diagonal blocks are $A_1, \dots, A_k \in \mathbb{R}^{m\times m}$. Given continuously differentiable functions $f : \mathbb{R}^n \to \mathbb{R}$ and $G : \mathbb{R}^n \to \mathbb{R}^m$, we let $\nabla_x f(x) \in \mathbb{R}^n$ denote the gradient of f evaluated at $x \in \mathbb{R}^n$; the k-th element of $\nabla_x f(x)$ is $\frac{\partial f(x)}{\partial [x]_k}$. Furthermore, we let $\partial_x G(x) \in \mathbb{R}^{m\times n}$ denote the Jacobian of function G evaluated at $x \in \mathbb{R}^n$; the ij-th element of $\partial_x G(x)$ is $\frac{\partial [G(x)]_i}{\partial [x]_j}$.

II. Quantal Response Equilibrium in Matrix Games

We introduce our theoretical model, a multiplayer matrix game where the cost perceived by each player is corrupted by stochastic noise.

A. Multiplayer Matrix Games With Perception Noise

We consider a game with $n \in \mathbb{N}$ players. Each player $i \in [n]$ has $m_i \in \mathbb{N}$ actions. We let $m \coloneqq \sum_{i=1}^n m_i$ denote the total number of actions of all players. Player i's strategy is an m_i-dimensional probability distribution over all possible actions, denoted by $x_i \in \Delta_i$, where

$$\Delta_i \coloneqq \{y \in \mathbb{R}^{m_i} | y^\top \mathbf{1}_{m_i} = 1, y \geq 0\}. \tag{1}$$

Each player's optimal strategy is one that minimizes the perceived cost, which is jointly determined by the strategies of all players and a stochastic perception error. In particular, let $b_i \in \mathbb{R}^{m_i}$ and $C_{ij} \in \mathbb{R}^{m_i\times m_j}$ with $C_{ii} = C_{ii}^\top \succeq 0$ for all $i, j \in [n]$ be cost parameters. Then the cost of action k perceived by player i is given by

$$[b_i + \sum_{j=1}^n C_{ij}x_j]_k + \xi_{ik}, \tag{2}$$

where ξ_{ik} is a random variable that captures perception error in player i's decision-making.

If each ξ_{ik} is independently sampled from the Gumbel distribution with mean $\gamma\lambda$ for some $\lambda \in \mathbb{R}_{++}$ where γ is Euler's constant, then the optimal strategy for player i is

$$x_i = f_i\Big(-\tfrac{1}{\lambda}(b_i + \textstyle\sum_{j=1}^n C_{ij}x_j)\Big) \tag{3}$$

where $f_i(z) \coloneqq \frac{1}{\mathbf{1}_{m_i}^\top \exp(z)} \exp(z)$ for all $z \in \mathbb{R}^{m_i}$, and $\exp(z) \in \mathbb{R}_{++}^{m_i}$ is the elementwise exponential of vector z. The strategy in (3), known as the *logit quantal response*, models the bounded rationality in decision-making, and has been effective in consumer choice problems [16]; see [14], [15] for a detailed discussion. We define the concept of *logit quantal response equilibrium* formally as follows.

Definition 1: A joint strategy $x \coloneqq \begin{bmatrix} x_1^\top & x_2^\top & \cdots & x_n^\top \end{bmatrix}^\top$ is a *quantal response equilibrium* if (3) holds for all $i \in [n]$.

The following lemma provides an optimization-based characterization of the equilibrium in Definition 1.

Lemma 1: If $\lambda > 0$ and $C_{ii} = C_{ii}^\top \succeq 0$ for all $i \in [n]$, then (3) holds if and only if

$$x_i \in \operatorname*{argmin}_{y\in\Delta_i} \left(b_i + \frac{1}{2}C_{ii}y + \sum_{j\neq i} C_{ij}x_j\right)^\top y + \lambda y^\top \ln(y), \tag{4}$$

where $\ln(y)$ denotes the elementwise logarithm of y.

Proof: Since $C_{ii} = C_{ii}^\top \succeq 0$ and the set Δ_i is nonempty, the condition in (4) holds if and only if the following Karush-Kuhn-Tucker conditions hold for some $v_i \in \mathbb{R}$ and $u_i \in \mathbb{R}^{m_i}$

$$\begin{aligned} &b_i + \sum_{j=1}^n C_{ij}x_j + (\lambda + v_i)\mathbf{1}_{m_i} + \lambda \ln(x_i) - u_i = 0_{m_i}, \\ &x_i^\top \mathbf{1}_{m_i} = 1, \quad x_i \geq 0, \quad u_i \geq 0, \quad u_i^\top x_i = 0. \end{aligned} \tag{5}$$

Furthermore, since the logarithm function is only defined for strictly positive numbers, we know that (4) implies that x_i is elementwise positive. Combining this fact with (5) gives $u_i = 0_{m_i}$. Finally, one can directly verify the equivalence between (3) and (5) when $u_i = 0_{m_i}$. ∎

Remark 1: If $n = 2$, $C_{ii} = 0_{m_i\times m_i}$ for $i = 1, 2$, and $C_{12} = -C_{21}$, then Definition 1 reduces to the quatal response equlibrium in two-player zero-sum games [6].

B. Computing the Quantal Response Equilibrium via Nonlinear Least-Squares

We can compute the quantal response equilibrium in Definition 1 by solving the following *nonlinear least-squares problem*:

$$\operatorname*{minimize}_x \sum_{i=1}^n \left\| x_i - f_i\left(-\frac{1}{\lambda}(b_i + \sum_{j=1}^n C_{ij}x_j)\right)\right\|^2 \tag{6}$$

where function f_i is given by (3). If the optimal value of the objective function in optimization (6) is zero, then the corresponding solution x indeed satisfies (3) for all $i \in [n]$.

However, the question remains whether optimization (6) has an optimal value of zero, or whether it has a unique solution. We will answer these questions next.

Throughout we will also use the following notation:

$$b := \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}, \quad C := \begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} \\ C_{21} & C_{22} & \dots & C_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1} & C_{n2} & \dots & C_{nn} \end{bmatrix}. \tag{7}$$

We make the following assumptions on optimization (6).

Assumption 1: $\lambda > 0$, $C+C^\top \succeq 0$, $C_{ii} = C_{ii}^\top$ for all $i \in [n]$.

The following proposition provides sufficient conditions under which the quantal response equilibrium in Definition 1 exists and is unique.

Proposition 1: If Assumption 1 holds, then there exists a unique $x = \begin{bmatrix} x_1^\top & x_2^\top & \cdots & x_n^\top \end{bmatrix}^\top \in \mathbb{R}^m_{++}$ such that (3) holds for all $i \in [n]$.

Proof: We first prove the existence. Since $C+C^\top \succ 0$, all of its principle submatrices are also positive semidefinite. Hence $C_{ii} \succeq 0$ for all $i \in [n]$. In addition, since $y^\top \ln(y)$ is a convex function of y, we conclude that any x that satisfies (4) is the Nash equilibrium of a concave game, which always exists, due to [17, Th. 1].

Next, we prove the uniqueness of x. The logarithm function in (4) ensures that x is elementwise positive. By combining this fact together with Assumption 1, we can show that $C + C^\top + \lambda \operatorname{diag}(x)^{-1}$ is positive definite. Hence any x that satisfies (4) is the Nash equilibrium of a diagonally strict concave game, which is unique [17, Th. 6].

The rest of the proof follows from the equivalence between (3) and (4), due to Lemma 1. ∎

In practice, different cost functions can induce the same equilibrium, even those violating Assumption 1. The cost functions satisfying Assumption 1, however, eliminate the ambiguity in the quantal response equilibrium, as shown by Proposition 1.

III. Numerical Methods for Inverse Matrix Games

We now consider the following inverse game problem: given a desired joint strategy x, how can one infer the cost matrix C that makes x the unique equilibrium in Definition 1. Here we only consider the inferring of the matrix C. We note that one can seamlessly generalize the results in this section to the inference of vector b.

In the following, we will introduce two different approaches for the aforementioned inverse matrix game: one based on semidefinite programs, the other based on the projected gradient method for bilevel optimization.

A. Semidefinite Program Approach

We first consider the case where the desired equilibrium is a pure joint strategy, where each player i has a preferred action $i^\star \in [m_i]$. In particular, suppose there exists $x^\star \in \mathbb{R}^m$ and $i^\star \in [m_i]$ for all $i \in [n]$ such that $[x_i^\star]_k$ equals 1 if $k = i^\star$, and 0 otherwise.

In this case, perhaps the most direct way to ensure $x^\star$ is an equilibrium is to simply make sure that the cost of action $i^\star$ is sufficiently lower than any alternatives for player i. By combining these constraints together with the results in Proposition 1, we obtain the following semidefinite program:

$$\begin{aligned} \underset{C}{\text{minimize}} \quad & \frac{1}{2}\|C\|_F^2 \\ \text{subject to} \quad & C + C^\top \succeq 0, \quad C_{ii} = C_{ii}^\top, \quad i \in [n], \\ & [b_i]_{i^\star} + \sum_{j=1}^{n} C_{i^\star j^\star} + \varepsilon \le [b_i]_k + \sum_{j=1}^{n} C_{kj^\star}, \\ & \forall k \in [m_i] \setminus \{i^\star\}, \quad i \in [n]. \end{aligned} \tag{8}$$

where the objective function penalizes large values of the elements in matrix C, and $\varepsilon \in \mathbb{R}_+$ is a tuning parameter that separates the cost of the best action from the cost of the second best action. Intuitively, as the value of ε increases, the quantal response equilibrium in Definition 1 is more likely to take a pure form.

The drawback of optimization (8) is that it only applies to the case where the desired equilibrium is known and close to be deterministic. If the desired equilibrium is mixed, i.e., each player has a preferred probability distribution over all actions rather than one single preferred action, then the semidefinite program is no longer useful.

B. Bilevel Optimization Approach

We now consider the case where the desired equilibrium is described by a *performance function*, rather than explicitly as a desired joint strategy. In particular, we consider the following continuously differentiable function, denoted by $\psi : \mathbb{R}^m \to \mathbb{R}$, that evaluates the quality of a joint strategy. For example, if $x^\star = \begin{bmatrix} (x_1^\star)^\top & (x_2^\star)^\top & \cdots & (x_n^\star)^\top \end{bmatrix}^\top$ is the desired equilibrium, then a possible choice of function ψ is as follows:

$$\psi(x) = D_{KL}(x, x^\star) := \sum_{i=1}^{n} x_i^\top (\ln(x_i) - \ln(x_i^\star)). \tag{9}$$

The above choice of function $\psi(x)$ measures the sum of the Kullback–Leibler (KL) divergence between each player's strategy and the corresponding desired strategy.

In order to compute the value of matrix C such that the equilibrium in Definition 1 is unique and minimizes the value of performance function $\psi(x)$, we introduce the following *bilevel optimization problem*:

$$\begin{aligned} \underset{x,C}{\text{minimize}} \quad & \psi(x) \\ \text{subject to} \quad & C + C^\top \succeq 0, \quad C_{ii} = C_{ii}^\top, \quad i \in [n], \\ & \|C\|_F \le \rho, \quad x \text{ is optimal for optimization (6).} \end{aligned} \tag{10}$$

Here $\rho \in \mathbb{R}_+$ is a tuning parameter that controls the maximum allowed Frobenius norm of matrix C. Intuitively, the larger the value of ρ, the more choices of matrix C from which we can choose, and the more likely we can achieve a lower value of function $\psi(x)$.

The drawback of optimization (10) is that, unlike the semidefinite program in (8), it is nonconvex and, as a result, one can only hope to obtain a locally optimal solution. Next, we will discuss how to compute a locally optimal solution efficiently using the projected gradient method.

1) Differentiating Through the Equilibrium Condition: The key to solve bilevel optimization (10) is to compute the gradient of $\psi(x)$ with respect to matrix C. In particular, we let $\nabla_C \psi(x) \in \mathbb{R}^{m\times m}$ be the matrix whose pq-th element, denoted by $[\nabla_C \psi(x)]_{pq}$, is given by

$$[\nabla_C \psi(x)]_{pq} \coloneqq \frac{\partial \psi(x)}{\partial [C]_{pq}} \tag{11}$$

for all $p, q \in [m]$. Since function ψ is continuously differentiable, the difficulty in evaluating $\nabla_C \psi(x)$ is to compute the Jacobian of the equilibrium x with respect to matrix C. To this end, we introduce the following notation:

$$u \coloneqq -\frac{1}{\lambda}(b + Cx) \tag{12a}$$

$$f(u) \coloneqq \begin{bmatrix} f_1(u_1)^\top & f_2(u_2)^\top & \cdots & f_n(u_n)^\top \end{bmatrix}^\top \tag{12b}$$

where $u_i \in \mathbb{R}^{m_i}$ for all $i \in [n]$, and f_i is given by Lemma 1. The following result provides a formula to compute $\nabla_C \psi(x)$ using the *implicit function theorem* [18].

Proposition 2: Suppose $C + C^\top \succeq 0$ and $\lambda > 0$. Let $x \coloneqq \begin{bmatrix} x_1^\top & x_2^\top & \cdots & x_n^\top \end{bmatrix}^\top$ be such that (3) holds for all $i \in [n]$, $\psi{:}\mathbb{R}^m \to \mathbb{R}$ be a continuously differentiable function, u and $f(u)$ given by (12). If $I_m + \frac{1}{\lambda}\partial_u f(u)C$ is nonsingular, then

$$\nabla_C \psi(x) = -\frac{1}{\lambda}\partial_u f(u)^\top (I_m + \frac{1}{\lambda}\partial_u f(u)C)^{-\top} \nabla_x \psi(x) x^\top.$$

Proof: Let $F(x, C) \coloneqq x - f(u)$ and C_q denote the q-th column of matrix C. Proposition (1) implies x is the unique vector that satisfies $F(x, C) = 0_m$. Since f is a continuously differentiable function, the implicit function theorem [18, Th. 1B.1] implies the following: if $\partial_x F(x, C)$ is nonsingular, then $\frac{\partial x}{\partial C_q} = -(\partial_x F(x, C))^{-1}\partial_{C_q} F(x, C)$. Using the chain rule we can show $\partial_x F(x, C) = I_m + \frac{1}{\lambda}\partial_u f(u)C$ and $\partial_{C_q} F(x, C) = \frac{1}{\lambda}[x]_q \partial_u f(u)$. The rest of the proof is due to the chain rule and the definition of $\nabla_C \psi(x)$ in (11). ∎

The gradient formula in Proposition (2) requires computing matrix inverse, which can be numerically unstable, In practice, we use the following formula:

$$\nabla_C \psi(x) = -\frac{1}{\lambda}\partial_u f(u)^\top ((I_m + \frac{1}{\lambda}\partial_u f(u)C)^\dagger)^\top \nabla_x \psi(x) x^\top, \tag{13}$$

where † denotes the Moore–Penrose pseudoinverse. Note that if $I_m + \frac{1}{\lambda}\partial_u f(u)C$ is nonsingular, then Proposition 2 implies $\hat{\nabla}_C \psi(x) = \nabla_C \psi(x)$; otherwise, the value of $\hat{\nabla}\psi(x)$ provides only an *approximation* of $\nabla_C \psi(x)$.

2) Approximate Projected Gradient Method: Equipped with Proposition 2 and the projection formula in (13), we are now ready to introduce the approximate projected gradient method for bilevel optimization (10). To this end, we define the set $\mathbb{D} \subset \mathbb{R}^{m\times m}$:

$$\mathbb{D} \coloneqq \{C |\ C + C^\top \succeq 0,\ \ \|C\|_F \le \rho,\ \ C_{ii} = C_{ii}^\top,\ \ i \in [n]\}. \tag{14}$$

We summarize the approximate projected gradient method in Algorithm 1, where the projection map $\Pi_{\mathbb{D}} : \mathbb{R}^{m\times m} \to \mathbb{R}^{m\times m}$ is given by

$$\Pi_{\mathbb{D}}(C) = \operatorname*{argmin}_{X\in\mathbb{D}} \|X - C\|_F \tag{15}$$

Algorithm 1 Approximate Projected Gradient Method

Input: Function $\psi : \mathbb{R}^m \to \mathbb{R}$, vector $b \in \mathbb{R}^m$, scalar weight $\lambda \in \mathbb{R}_{++}$, step size $\alpha \in \mathbb{R}_{++}$, stopping tolerance ϵ.

1: Initialize $C = 0_{m\times m}$, $C^+ = 2\epsilon I_m$
2: **while** $\|C^+ - C\|_F > \epsilon$ **do**
3: $\quad C \leftarrow C^+$
4: $\quad$ Solve optimization (6) for x.
5: $\quad C^+ \leftarrow \Pi_{\mathbb{D}}(C - \alpha\hat{\nabla}_C \psi(x))$
6: **end while**

Output: Equilibrium x and cost matrix C.

for all $C \in \mathbb{R}^{m\times m}$. At each iteration, this method first solve the nonlinear least-squares problem in (6), then update matrix C using the approximate gradient in (13).

A key step in Algorithm 1 is to compute the projection in (15). The following lemma provides the explicit computational formula for computing this projection via eigenvalue decomposition and matrix normalization.

Lemma 2: Let set $\mathbb{D}$ be given by (14) and matrix $C \in \mathbb{R}^{m\times m}$ be partitioned as in (7), where $C_{ij} \in \mathbb{R}^{m_i\times m_j}$, for all $I, j \in [n]$. Let $B = C - \text{blkdiag}(C_{11}, \ldots, C_{nn}^\top)$, $U \in \mathbb{R}^{m\times m}$, and $s \in \mathbb{R}^m$ be such that $U\,\text{diag}(s)U^\top = \frac{1}{2}(C + C^\top)$. Let Then

$$\Pi_{\mathbb{D}}(C) = \frac{\rho}{\max\{\rho, \|A\|_F\}} A, \tag{16}$$

where $A \coloneqq \frac{1}{2}(B - B^\top) + U\,\text{diag}(\max(s, 0))U^\top$.

Proof: Let $\text{tr}(M)$ denote the trace of matrix M. First, let $\mathbb{K}_1 \coloneqq \{M \in \mathbb{R}^{n\times n} | M = M^\top \succeq 0\}$, $\mathbb{K}_2 \coloneqq \{M \in \mathbb{R}^{n\times n} | M = -M^\top,\ \ M_{ii} = M_{ii}^\top,\ \ \forall i \in [n]\}$ where $M_{ii} \subset \mathbb{R}^{m_i\times m_i}$ denotes the i-th principal submatrix of M, and $\mathbb{B} \coloneqq \{M \in \mathbb{R}^{n\times n} | \|M\|_F \le \rho\}$. Then one can verify that $\mathbb{D} = (\mathbb{K}_1 + \mathbb{K}_2) \cap \mathbb{B}$ where $+$ denotes the direct sum, and that $\mathbb{K}_1 + \mathbb{K}_2$ is a closed convex cones. By using the results on the projection onto the intersection of a ball and a closed convex cone [19, Th. 7.1], we can show that $\Pi_{\mathbb{D}}(C) = \Pi_{\mathbb{B}}(\Pi_{\mathbb{K}_1+\mathbb{K}_2}(C))$.

Second, observe that $C \in \mathbb{K}_1 + \mathbb{K}_2$ if and only if $\tilde{C} \coloneqq (\frac{1}{2}(C + C^\top), \frac{1}{2}(C - C^\top)) \in \mathbb{K}_1 \times \mathbb{K}_2$, where $\times$ denotes the Cartesian product. By using the results on the projection onto the Cartesian product of sets [20, Proposition 29.4], we can show that $\Pi_{\mathbb{K}_1\times\mathbb{K}_2}(\tilde{C}) = (\Pi_{\mathbb{K}_1}(\frac{1}{2}(C+C^\top)), \Pi_{\mathbb{K}_2}(\frac{1}{2}(C-C^\top)))$. Hence $\Pi_{\mathbb{K}_1+\mathbb{K}_2}(C) = \Pi_{\mathbb{K}_1}(\frac{1}{2}(C + C^\top)) + \Pi_{\mathbb{K}_2}(\frac{1}{2}(C - C^\top))$. The rest of the proof is a direct application of the formulas for projecting onto norm balls [20, Ex. 3.18] and projecting a symmetric matrix onto the positive semidefinite cone [20, Ex. 29.32]. ∎

IV. Numerical Examples

We demonstrate the application of the numerical methods in Section III using two examples. In these examples, we assume that vector b is known and C is a zero matrix before our design process. We aim to design a nonzero matrix C—which can be interpreted as subsidies and tolls—that encourages the desired behavior. We note that if the value of either vector b or matrix C is unknown before design, one can first infer the values of b and C using the approaches in Section III, then proceed with the results in this section.

Throughout, we compute the entropy regularized equilibrium in Definition 1 by solving optimization (6) using the

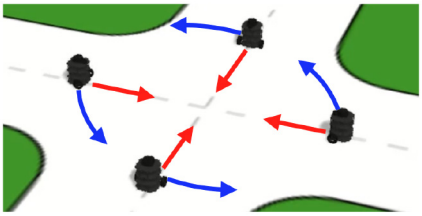


Fig. 2. An illustration of the equilibrium strategies in (17) (red arrows) and (18) (blue arrows).

Gauss-Newton method with line search [21, Sec. 10.3], where we terminated the algorithm when the objective function value in (6) is less than 10^{-10}. We note that, depending on the problem settings, other solution algorithms may have better performance. We refer the interested readers to [21, Ch. 10] for a detailed discussion on nonlinear least-squares.

A. Encouraging Collision Avoidance

We consider four ground rovers placed in a two-dimensional environment, at coordinate $(0, 1)$, $(0, -1)$, $(1, 0)$, and $(-1, 0)$, respectively. Each rover wants to reach the corresponding target position with coordinates $(0, -1)$, $(0, 1)$, $(-1, 0)$ and $(1, 0)$, respectively. Each rover can choose one of three candidate paths that connect its initial position to its target position: a beeline path of length 2; two semicircle paths, each of approximate length π, and one in the clockwise direction, the other one in the counterclockwise direction. We assume all rovers move at the same speed and start at the same time.

We model the decision-making of each rover using the entropy-regularized matrix game in Section II. In particular, we let $\lambda = 0.1$, $n = 4$, $m = 12$, and $b_i = \begin{bmatrix} 2 & \pi & \pi \end{bmatrix}^\top$ for all $i = 1, 2, 3, 4$. Here the elements in b_i denote the length of each candidate path, regardless of other players' strategies; if no other player exists, then action one (beeline path) is the optimal shortest path. If $C = 0_{12\times 12}$, one can verify—by solving an instance of optimization (6)—that the quantal response equilibrium in Definition 1—assuming λ is sufficiently small—is approximately

$$x_i = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, \quad i = 1, 2, 3, 4. \tag{17}$$

In other words, all players tend to choose the beeline path since it has the minimum length. However, this causes collisions among the rovers at coordinate $(0, 0)$.

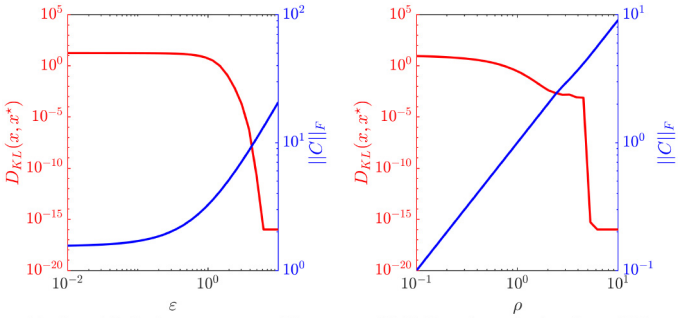
By adjusting the value of matrix C, we aim to change the equilibrium above to the following

$$x_i^\star = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}, \quad i = 1, 2, 3, 4. \tag{18}$$

In other words, we want all players to choose the counterclockwise semicircle path. See Fig. 2 for an illustration and https://www.youtube.com/watch?v=EvtPp_DWqgU for an animation.

We note that one can change the equilibrium from (17) to (18) by simply letting $C = 0_{m\times m}$ and modifying the value of b. However, such a choice of parameter implies that all rovers will voluntarily choose a longer path regardless of other rovers' strategies, which has no meaningful interpretation in path planning.

Since the equilibrium in (18) is of the form in Section III-A, we can compute matrix C using either the semidefinite program (8) or the bilevel optimization (10); in the latter case, we choose the performance function to be the KL-divergence in (9).



(a) Semidefinite program (8). (b) Bilevel optimization (10).

Fig. 3. The trade-off between $D_{KL}(x, x^\star)$ and $\|C\|_F$ when tuning the parameter ϵ in the semidefinite program (8) and the parameter ρ in bilevel optimization (10).

We solve the semidefinite program (8) using the off-the-shelf solver, and the bilevel optimization (10) using Algorithm 1. Fig. 3 shows the trade-off between $D_{KL}(x, x^\star)$—which measures the distance between the equilibrium x that corresponds to matrix C and the desired equilibrium $x^\star$—and $\|C\|_F$ of the computed matrix C when tuning the parameter in (8) and (10). These results confirm that both the semidefinite program approach and the bilevel optimization approach apply to the cases where the desired Nash is pure and known explicitly. Furthermore, both approaches require careful tuning of algorithmic parameters to achieve a preferred trade-off between $D_{KL}(x, x^\star)$ and $\|C\|_F$.

B. Encouraging Fair Resource Allocation

We now consider a case where the desired equilibrium is not of the explicit form in Section III-A. Instead, we only have access to a performance function that implicitly describes the desired equilibrium. To this end, we consider the following three-player game. Each player is a delivery drone company that provides package-delivery service, located in the southwest, southeast, and east area of Austin, respectively. Each strategy demotes the distribution of service allocated to the nine areas of Austin; we assume all three companies have the same amount of service to allocate. For each company, within its home area (where it is located), the operating cost of delivery service is one unit; outside the home area, the operating cost increases by 50% in an area adjacent to the home area, and 80% otherwise. See Fig. 4 for an illustration.[1] We model the joint decision of the three companies using the matrix game in Section II, where $n = 3$, $m_i = 9$ for $i = 1, 2, 3$, and $m = 27$; we set $\lambda = 0.1$ and vector b according to the aforementioned operating cost.

If all companies consider only the operating cost, they will only allocate services to their respective home area. We aim to infer the value of matrix C using Algorithm 1 that encourages a fair allocation to other areas. In particular, we choose the performance function as follows:

$$\psi(x) = \mathbf{1}_9^\top (x_1 + x_2 + x_3)^{-1}, \tag{19}$$

[1]Picture credit: https://en.wikipedia.org/wiki/List_of_Austin_neighborhoods.

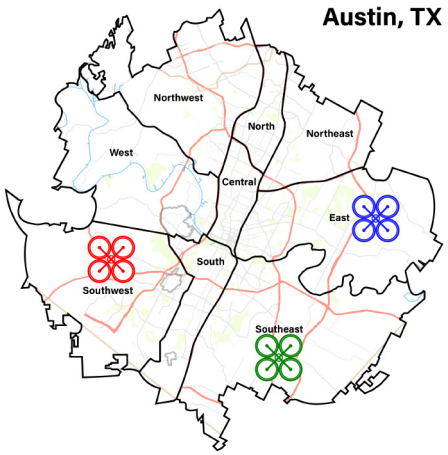


Fig. 4. Three drone delivery companies (red, green, and blue) located in different areas in the city of Austin.

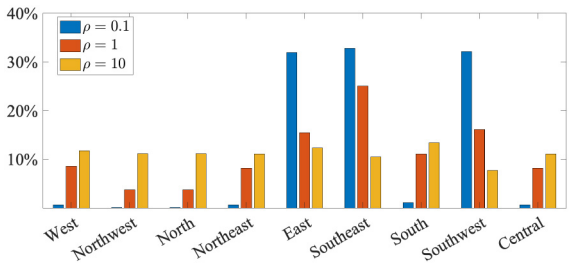


Fig. 5. The percentages of the total amount of delivery service allocated to each area at the equilibrium computed by Algorithm 1.

where vector $(x_1 + x_2 + x_3)^{-1}$ denotes the elementwise reciprocal of vector $x_1 + x_2 + x_3$. Function $\psi(x)$ is based on the the *potential delay function* from the resource allocation literature; the latter is a special case of the more general α-fairness function [22, Sec. 2.4]. Here function $\psi(x)$ measures the overall fairness of the delivery service allocation. Here we implicitly assume that the demand for delivery services is much higher than the supply, and we aim to allocate all supply. Such an assumption is common in the resource allocation literature [22]. The competition is among different suppliers (companies), not between supply and demand.

We compute the cost matrix using Algorithm 1 and illustrate the percentages of the delivery service allocated to each area at the equilibrium in Fig. 5. The results show that when $\rho \approx 0$, all the drone fleets will almost only serve their respective home areas. As we increase the value of ρ, the computed matrix encourages a more fair joint strategy where all nine areas receive an almost equal amount of service.

V. Conclusion

We study the inverse game problem in the context of multi-player matrix games. We develop efficient numerical methods to compute the cost matrices that ensure a unique quantal response equilibrium.

However, the current work only provides a preliminary proof of concept with limited applications. For example, it requires an exhaustive enumeration of all actions, which is computationally unscalable and makes the cost inference of a pure target equilibrium trivial: one can simply assign the lowest cost to the target actions. Furthermore, it only provides the cost functions that induce one desired equilibrium, rather than multiple equilibria with common desired properties. We aim to address these limitations by considering games with more complicated decision-making models. We will also consider simultaneously optimizing multiple equilibria using robust optimization.

Acknowledgment

The authors would like to thank Yigit E. Bayiz, Shenghui Chen, Benjamin Grimmer, Dayou Luo, Shahriar Talebi, and the anonymous editor and reviewers for their feedback.

References

[1] O. Morgenstern and J. Von Neumann, *Theory of Games and Economic Behavior*. Princeton, NJ, USA: Princeton Univ. Press, 1953.
[2] J. Nash, "Equilibrium points in n-person games," *Proc. Nat. Acad. Sci.*, vol. 36, no. 1, pp. 48–49, 1950.
[3] J. Nash, "Non-cooperative games," *Ann. Math.*, vol. 54, no. 2, pp. 286–295, 1951.
[4] T. L. Molloy, J. I. Charaja, S. Hohmann, and T. Perez, *Inverse Optimal Control and Inverse Noncooperative Dynamic Game Theory*. Cham, Switzerland: Springer, 2022.
[5] D. Bertsimas, V. Gupta, and I. C. Paschalidis, "Data-driven estimation in equilibrium using inverse optimization," *Math. Program.*, vol. 153, no. 2, pp. 595–633, 2015.
[6] C. K. Ling, F. Fang, and J. Z. Kolter, "What game are we playing? End-to-end learning in normal and extensive form games," in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 396–402.
[7] T. Başar, "Affine incentive schemes for stochastic systems with dynamic information," *SIAM J. Control Optim.*, vol. 22, no. 2, pp. 199–210, 1984.
[8] N. Nisan and A. Ronen, "Algorithmic mechanism design," *Games Econ. Behav.*, vol. 35, no. 1, pp. 359–379, 2015.
[9] S. Kalyanaraman and C. Umans, "The complexity of rationalizing matchings," in *Proc. Int. Symp. Algorithms Comput.*, 2008, pp. 171–182.
[10] S. Kalyanaraman and C. Umans, "The complexity of rationalizing network formation," in *Proc. Annu. IEEE Symp. Found. Comput. Sci.*, 2009, pp. 485–494.
[11] D. Nekipelov, V. Syrgkanis, and E. Tardos, "Econometrics for learning agents," in *Proc. ACM Conf. Econ. Comput.*, 2015, pp. 1–18.
[12] V. Kuleshov and O. Schrijvers, "Inverse game theory: Learning utilities in succinct games," in *Proc. Int. Conf. Web Internet Econ.*, 2015, pp. 413–427.
[13] L. Peters, D. Fridovich-Keil, C. J. Tomlin, and Z. N. Sunberg, "Inference-based strategy alignment for general-sum differential games," in *Proc. Int. Conf. Auton. Agents MultiAgent Syst.*, 2020, pp. 1037–1045.
[14] R. D. McKelvey and T. R. Palfrey, "Quantal response equilibria for normal form games," *Games Econ. Behav.*, vol. 10, no. 1, pp. 6–38, 1995.
[15] R. D. McKelvey and T. R. Palfrey, "Quantal response equilibria for extensive form games," *Exp. Econ.*, vol. 1, no. 1, pp. 9–41, 1998.
[16] D. McFadden, "Conditional logit analysis of qualitative choice behavior," in *Frontiers in Econometrics*. New York, NY, USA: Academic, 1973, pp. 105–142.
[17] J. B. Rosen, "Existence and uniqueness of equilibrium points for concave N-person games," *Econometrica J. Econ. Soc.*, vol. 33, no. 3, pp. 520–534, 1965.
[18] A. L. Dontchev and R. T. Rockafellar, *Implicit Functions and Solution Mappings: A View From Variational Analysis*. New York, NY, USA: Springer, 2014.
[19] H. H. Bauschke, M. N. Bui, and X. Wang, "Projecting onto the intersection of a cone and a sphere," *SIAM J. Optim.*, vol. 28, no. 3, pp. 2158–2188, 2018.
[20] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. New York, NY, USA: Springer, 2017.
[21] J. Nocedal and S. J. Wright, *Numerical Optimization*. New York, NY, USA: Springer, 1999.
[22] S. Shakkottai and R. Srikant, "Network optimization and control," *Found. Trends Netw.*, vol. 2, no. 3, pp. 271–379, 2008.