Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 9336–9341

Decision-Focused Learning for Inverse Noncooperative Games: Generalization Bounds and Convergence Analysis

Ahmad Al-Tawaha, Harshal Kaushik, Bilgehan Sel, Ruoxi Jia, Ming Jin

* *The Bradley Department of Electrical and Computer Engineering Virginia Tech, Blacksburg, VA, USA, {atawaha, harshaldkaushik, bsel, ruoxijia, jinming}@vt.edu*

Abstract: Finding the equilibrium strategy of agents is one of the central problems in game theory. Perhaps equally intriguing is the inverse of the above problem: from the available finite set of actions at equilibrium, how can we learn the utilities of players competing against each other and eventually use the learned models to predict their future actions? Instead of following an estimate-then-predict approach, this work proposes a decision-focused learning (DFL) method that directly learns the utility function to improve prediction accuracy. The game's equilibrium is represented as a layer and integrated into an end-to-end optimization framework. We discuss the statistical bounds of covering numbers for the set of solution functions corresponding to the solution of a generic parametric variational inequality. Also, we establish the generalization bound for the set of solution functions with respect to the smooth loss function with an improved rate. Moreover, we proposed an algorithm based on the iterative differentiation strategy to forward and backpropagate through the equilibrium layer. The convergence analysis of the proposed algorithm is established. Finally, We numerically validate the proposed framework in the utility learning problem among the agents whose utility functions are approximated by partially input convex neural networks (PICNN).

Keywords: data-driven decision-making, game theories, agent technology for business and economy, social resource planning and management, machine learning, statistical analysis, and multi-agent systems.

1. INTRODUCTION

The concept of equilibrium is fundamental in several disciplines, including economics, management science, operations research, and engineering Heidarkhani et al. (2019). The use of variation inequality (VI) provides a powerful unifying approach for the study of equilibrium problems Kostreva (1990). VI typically arises in network systems where problems are modeled using cooperative and noncooperative game approaches Scutari et al. (2010).

Traditionally, game theory focuses on depicting competing players' behaviors and their interactions using complicated mathematical models Roughgarden (2010). For a set of players in a game, the aim is to optimize their utility functions. These utility functions depend on the players and other players' strategies. Each individual tries to attain an outcome that is best for him/herself (Nash equilibrium) Facchinei and Pang (2003); Roughgarden (2010). However, the utility function used to calculate the equilibrium is not directly observable. While it can be estimated or modeled, a small error can potentially affect the resulting equilibrium Jia et al. (2018).

Equally interesting and practical is the inverse game problem, that is, investigating the utility functions of individuals that lead to the observed equilibrium (see, for example, Ratliff et al. (2014); Kuleshov and Schrijvers (2015); Bertsimas et al. (2015); Jia et al. (2018); Molloy et al. (2022); Ding et al. (2022); Adams et al. (2022)). Since previous equilibrium actions are often observable experimentally, it is possible to construct the agents' utility functions from the observed equilibrium.

Prior studies follow a two-stage approach (i.e., estimate-then-predict), where the utility function parameters are first learned based on specific optimality criteria. Then, a plug-in estimator is used to predict future equilibrium actions Ratliff et al. (2014); Bertsimas et al. (2015). The main issue lies in the propagation of estimation error to the downstream prediction, which is unaccounted for during the learning stage. In contrast, we propose a DFL approach where the downstream optimization problem is plugged into the prediction model. In order to make evidence-based decisions, In this paper, we design a decision-focused learning approach as a mathematical program with equilibrium constraints (MPEC) problem using finite instances of available actions.

The prime advantage of DFL is that the prediction error is directly minimized during the learning process. Such approach has been studied for convex Elmachtoub and

⋆ Sponsor and financial support acknowledgment goes here. Paper titles should be written in uppercase and lowercase letters, not all uppercase.

10.1016/j.ifacol.2023.10.221

Grigas (2020), combinatorial Mandi et al. (2020); Feber et al. (2020), and stochastic Donti et al. (2017) optimization problems. Statistical bounds in these settings have been studied in Balghiti et al. (2021); Wang et al. (2020); Bertsimas and Kallus (2019); Hu et al. (2021). Closely related to our setting is the work on zero-sum, extensive-form games Ling et al. (2018). However, the analysis of statistical complexity and the statistical bounds for these settings has not yet been thoroughly established.

In this work, game equilibrium is represented as a layer and integrated into an end-to-end optimization framework. The key contributions can be summarized as the following:

- We discuss the statistical bounds of covering numbers for the set of solution functions corresponding to the solution of affine parametric variational inequalities.
- We establish a generalization bound for the set of solution functions of affine variational inequalities for smooth loss function with an improved excess risk bound from $\mathrm{O}\left(\sqrt{\frac{\hat{R}}{n}}\log^{1.5}(n) + \frac{\log^3(n)}{n}\right)$ in Srebro et al. (2010) to $\mathrm{O}\left(\sqrt{\frac{\hat{R}}{n}}\log(n) + \frac{\log^2(n)}{n}\right)$, where $\hat{R}$ is the empirical risk of the hypothesis class. For the solution functions of generic variational inequalities, we provide an excess risk bound of $\mathrm{O}\left(\sqrt{\frac{\hat{R}}{n}}\log^{1.5}(n)C_y^{\left(\frac{2k}{n}+1\right)} + \frac{\log^3(n)C_y^{\left(\frac{4k}{n}+2\right)}}{n}\right)$, where C_y is the upper bound value of the solution function and k is the number of times the solution function is piecewise continuously differentiable within each piece.
- We propose an algorithm based on the iterative differentiation (ITD) strategy to calculate the gradient of the decision-focused objective with respect to the learning parameters (i.e., implicit gradient). Specifically, the implicit gradient is obtained by using the notion of merit function (D-gap function) and fixed-point equations in Section 3.
- We extend the convergence analysis of the current literature of bilevel optimization problems Al-Shedivat et al. (2017); Huisman et al. (2021); Finn et al. (2019); Raghu et al. (2019); Franceschi et al. (2018), network games Parise and Ozdaglar (2019), and iterative method of variational inequalities Shehu et al. (2019) to the proposed algorithm with a nonconvex optimization problem as the decision-focused objective. We show that the update of the proposed algorithm for K iterations converges to a stationary point with a rate $\mathcal{O}(1/K)$.

Notation. for convenience, we use $\|\cdot\|$ for the standard Euclidean norm. We use $\mathcal{P}_Y(\omega)$ for the projection of ω onto set Y. In the generalization error bound analysis, we represent the function class as $\mathcal{H}$. A function in a class PC^k is piecewise smooth, and k times continuously differentiable within each piece.

2. PROBLEM FORMULATION

Consider a non-cooperative game among d players, each player j has a strategy vector $y^{(j)}$ selected from a set $Y_{j,u,\omega} \subseteq \mathbb{R}^{p_j}$, where $u \in \mathbb{R}^q$ is the context and $\omega \in \mathbb{R}^m$ is the learning parameter. The utility of agent j depends on $y^{(j)}$, and the strategy vector of other agents $y^{(-j)}$, where $y^{(-j)} = \{y^{(1)}, \ldots, y^{(j-1)}, y^{(j+1)}, \ldots, y^{(d)}\}$ denotes the set of strategies of all agents except agent j. Participants aim to maximize their utility functions and attain an individually optimal strategy Facchinei and Pang (2003); Roughgarden (2010). We provide a framework that supports the parametric estimation of the utility functions.

In parametric estimation, the utility function belongs to a known parametric family. We denote the utility function for parametric estimation with known parametric family as $g_j(\cdot, u, \omega) : Y_{u,\omega} \to \mathbb{R}$, where $Y_{u,\omega} = Y_{1,u,\omega} \times Y_{2,u,\omega} \times \cdots \times Y_{d,u,\omega} \subseteq \mathbb{R}^p$. In a more realistic setting where the true parametric family is unknown, we propose to estimate the unknown utility by a partially input convex neural network (PICNN) Amos et al. (2017) with the learning parameter ω, expressed as $\hat{g}_j(\cdot, u, \omega) : Y_{u,\omega} \to \mathbb{R}$. Also, $\hat{g}_j(\cdot, u, \omega)$ depends on unknown parameter ω and must be inferred form data.

The parameters of the utility functions are learned through observations. In particular, we would like to learn the parameter ω from a dataset $\{(u_1, y_1), \ldots, (u_n, y_n)\}$ that consists of n pairs of context u and agent actions y at the equilibrium by minimizing some loss represented by $f(m(u, \omega), y)$. The loss function $f(m(u, \omega), y)$ represents a measure of the quality of prediction by comparing the objective value of the solution generated using the prediction model and the observed actions at the equilibrium.

$$\underset{\omega \in \Omega}{\text{minimize}}\ \Phi(\omega) := f(m(u, \omega), y) \tag{1}$$

where $m(u, \omega) \in Y_{u,\omega} \subseteq \mathbb{R}^p$ is the predictor function of users' action used for estimating y with a learning parameter ω. The goal of decision-focused utility learning is to find ω that parameterizes the utility function such that the prediction error is minimized.

One question is how to choose the prediction model $m(u, \omega)$. Variation inequality is a modeling tool that captures the decision-making in game theory. Because we know the structure of our problem is a game, we make a structural assumption that the prediction model is a solution function of some governing variational inequality, where the parameters of the solution functions are trained in an end-to-end fashion. We start by defining the parametric variational inequality as the following

$$\mathrm{VI}(Y_{u,\omega}, F_{u,\omega}), \tag{2}$$

where $F_{u,\omega} : Y_{u,\omega} \to \mathbb{R}^p$ is an equilibrium map formed by the gradients of individual agent utility functions. For clarification, the set $Y_{u,\omega}$ and mapping $F_{u,\omega}$ are represented as follows

$$Y_{u,\omega} \triangleq \prod_{j=1}^{d} Y_{j,u,\omega_j} \quad \text{and} \quad F_{u,\omega} \triangleq \begin{bmatrix} \nabla_{y^{(1)}} \hat{g}_1(y, u, \omega_1) \\ \vdots \\ \nabla_{y^{(d)}} \hat{g}_d(y, u, \omega_d) \end{bmatrix}. \tag{3}$$

Solving a parametric $\mathrm{VI}(Y_{u,\omega}, F_{u,\omega})$ is to find $y^* \in \mathrm{SOL}(Y_{u,\omega}, F_{u,\omega})$; i.e., $y^* \in \mathrm{SOL}(Y_{u,\omega}, F_{u,\omega})$ if and only if $y^* \in Y_{u,\omega}$ and satisfies the following inequality

$$F_{u,\omega}(y^*)^T (z - y^*) \geq 0, \text{ for all } z \in Y_{u,\omega}, \tag{4}$$

where $y^*(u, \omega)$ is the true Nash function. In our case, the goal is to find a solution function $m(u, \omega)$ that approximates the true Nash function $y^*(u, \omega)$ well. The solution function plays a significant role in modeling such complex phenomena and decision-making processes. Moreover, the solution function is differentiable so that the parameters of the solution function can be trained in an end-to-end framework through the implicit gradient as developed in section 3. From the above discussion, the general framework is illustrated in Fig. 1.

3. METHODOLOGY

The decision focus utility learning problem in our setting is to learn the utility functions parameters ω such that the prediction error in the final stage is minimized; such a model should be trained robustly and in an end-to-end fashion.

A gradient-based method is used to solve (1). In forward propagation, we evaluate the prediction loss function, which in turn depends on the solution function of VI in (2). In order to solve the VI problem. We begin with necessary assumptions on the problem (2) structure.

Assumption 1. *The following hold for problem* (2)*:*

(a) *For any $u \in \mathcal{U}$ and $\omega \in \Omega$, the map $F_{u,\omega}(\cdot)$ is continuous differentiable, $L-$ Lipschitz, and $\mu-$ strongly monotone with respect to $y \in Y_{u,\omega}$.*
(b) *Sets Ω and $\mathcal{U}$ are closed, convex, and bounded such that for finite scalars $\bar{\mathcal{U}}$ and $\bar{\Omega}$, we have $\mathcal{U} \triangleq \{u \in \mathcal{U} \mid \|u\| \leq \bar{\mathcal{U}}\}$, $\Omega \triangleq \{\omega \in \Omega \mid \|\omega\| \leq \bar{\Omega}\}$.*
(c) *For any $i \in [n_{ineq}]$, $j \in [n_{eq}]$, we have linear functions $\theta_i^{ineq} : \mathbb{R}^p \times \mathbb{R}^m \to \mathbb{R}$ and $\theta_j^{eq} : \mathbb{R}^p \times \mathbb{R}^m \to \mathbb{R}$ such that that the set-valued map $Y_{u,\omega}$ is bounded polyhedral given as*

$$Y_{u,\omega} = \{y \in \mathbb{R}^p : \begin{array}{ll} \theta_i^{ineq}(y,u,\omega) \leq 0, & \textit{for all } i \in [n_{ineq}] \\ \theta_j^{eq}(y,u,\omega) = 0, & \textit{for all } j \in [n_{eq}]\} \end{array} \quad (5)$$

Under Assumption 1 on $F_{u,\omega}$ and $Y_{u,\omega}$, we establish the required conditions for the existence and uniqueness of the solution of $\mathrm{VI}(Y_{u,\omega}, F_{u,\omega})$ using the proposed approach. In this paper, the regularized D-gap function, which is a metric to characterize the optimality of the solution of the VI problem in (2), is considered and defined as the following.

Definition 1. *For any scalars $b > a > 0$, $y \in \mathbb{R}^p$, the gap function of $\phi_{ab}(y,\omega)$ is defined as $\phi_{ab}(y,\omega) \triangleq \phi_a(y,\omega) - \phi_b(y,\omega)$, where for some $c > 0$ and any positive definite matrix G, $\phi_c(y,\omega)$ is given by*

$$\phi_c(y,\omega) \triangleq \sup_{z \in Y_{u,\omega}} \left\{\langle F_{u,\omega}(y), y - z\rangle - \tfrac{c}{2}(y-z)^T G (y-z)\right\}. \quad (6)$$

The advantages of using the regularized gap functions appear in analyzing the convergence rate of various iterative techniques. Also, considering the regularized gap functions is useful to derive the implicit gradient, as we will show later in Lemma 1.

Considering Definition 1, for some $y(u,\omega)$, if the value function $\phi_{ab}(y(u,\omega),\omega) = 0$, then y solves $\mathrm{VI}(Y_{u,\omega}, F_{u,\omega}(\cdot))$ Facchinei and Pang (2003). Using the definition of the D-gap function, in the following result, we show that the solution of VI can be neatly obtained by solving a fixed-point equation via a projector operator, which paved the way to accomplish forward propagation. Note that, implicit differentiation can be used to derive $\nabla_\omega y$ to support backpropagation. That is, to update the value of ω_k, we obtain the gradient of the objective function with respect to the parameter ω. As the solution function is also a function of ω, the key is to obtain $\nabla_\omega y$.

Lemma 1. *Let Assumption 1 hold and $y \in Y_{u,\omega}$ be a solution of the VI, i.e., $y \in \mathrm{SOL}(Y_{u,\omega}, F_{u,\omega}(\cdot))$. Then for scalars $b > 0$, we have the following*

(a) *For scalar $b > 0$, we have*
$$y = z_b^*(y,\omega), \quad (7)$$
where $z_b^*(y,\omega) = \mathcal{P}_{Y_{u,\omega}}\left(y - \frac{1}{b}F_{u,\omega}(y)\right)$ is the unique solution of $\phi_b(y,\omega)$.
(b) *The implicit gradient $\nabla_\omega y$ can be obtained by solving the following linear equation:*
$$\nabla_\omega y = \langle \underbrace{\nabla_y z_b^*(y,\omega)}_{term\ 1}, \nabla_\omega y\rangle + \underbrace{\nabla_\omega z_b^*(y,\omega)}_{term\ 2}, \quad (8)$$
where terms 1, and 2 can be obtained from differentiating through the solution of the projection problem in (a).

Due to space limitations, we provide all the proofs in this online document Al-Tawaha et al. (2022). Lemma 1 implies that finding a solution to $\mathrm{VI}(Y_{u,\omega}, F_{u,\omega})$ is equivalent to finding a fixed point of $z_b^*(y,\omega)$, that accomplishes the task of forward propagation through the variation inequality. The existence and uniqueness of the solution function, which can be established under Assumption 1, enables us to implicitly differentiate through $z_b^*(y,\omega)$ to derive $\nabla_\omega y$ that fulfills the backward propagation through the variational inequality. Note that we avoid the backpropagation by unrolling the forward computations within an automatic differentiation in evaluating the implicit gradient. We obtain the implicit gradient by using the ideas of the D-gap function and fixed-point equations. Therefore, the proposed approach does not require the storage of intermediate terms of the iterative method to compute the fixed point, making it computationally efficient.

4. PROPERTIES OF THE SOLUTION FUNCTION OF VI

This section provides a mathematical characterization of the properties of the solution functions of parametric variation inequalities. Specifically, we answer the question: What is the class of the solution functions of variation inequalities? We start by discussing a simple case where $F_{u,\omega}$ is affine mapping and Assumption 1 (b and c) hold on set $Y_{u,\omega}$. As we will show later, solving these variation inequalities using the proposed approach is equivalent to solving multi-parametric quadratic programming.

Lemma 2 (Theorem 3.1 Pistikopoulos et al. (2020))**.** *Considering a multi-parametric quadratic programming problem (mp-QP), and let Assumption 1 (c) hold on set $Y_{u,\omega}$, then the optimizer $z_b^*(y,\omega)$ is continuous and piecewise affine.*

Note that, from Lemma 1, the solution of the parametric variation inequality is given by the projection of $y - \frac{1}{b}F_{u,\omega}(y)$ on $Y_{u,\omega}$, then this projection problem is an mp-QP.

Next, we discuss the class of solution functions for generic variational inequalities. For the general case analysis, we extend assumption 1(c). We assume that the set valued-map $Y_{u,\omega}$ satisfies constraint qualifications (CQs), including Mangasarian-Fromovitz constraint qualification (MFCQ),

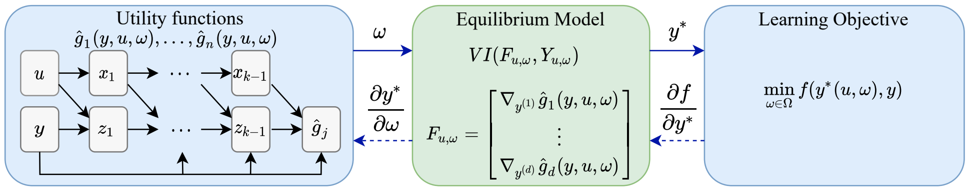

Fig. 1. In the general framework, players' utility functions are approximated using (PICNN). A VI is an embedding layer in the learning, which can capture proper inductive bias, such as the equilibrium of a game. This framework connects a learning model and a VI in an end-to-end differentiable learning framework.

Constant Rank Constraint Qualification (CRCQ), and Strong Coherent Orientation Condition (SCOC). Under these assumptions, we can show that the solution function is piecewise continuous PC^1.

Lemma 3 (Theorem 4.2.16 Luo et al. (1996)). *Let the set valued-map $Y_{u,\bar{\omega}}$ such that constraint qualifications (MFCQ, CRCQ, SCOC) are satisfied. Now let $y^*(\bar{u},\bar{\omega})$ be the solution of $\mathrm{SOL}(Y_{u,\bar{\omega}}, F_{\bar{u},\bar{\omega}}(\cdot))$. Then, there exists a neighborhood $\bar{\Omega} \times \bar{Y}$ of $(\bar{\omega}, \bar{y})$, such that $y : \bar{\Omega} \to \bar{Y}$ is piecewise smooth PC^1 and y is a unique solution map of $\mathrm{VI}(Y_{u,\omega}, F_{u,\omega}(\cdot))$.*

5. BOUNDS ON GENERALIZATION ERROR

One fundamental theoretical question is about the learnability of the solution functions. In this section, we start by discussing the covering number of the set of solution functions, which are the cornerstones to establish the generalization error boundsMohri et al. (2018).

5.1 Covering number bound

We obtain the L_2 covering number bounds for the affine parametric variational inequality solution function. The L_2 covering number $\mathcal{N}_2(\epsilon, \mathcal{H}, \mathcal{D}_n)$ of the set of solution function $\mathcal{H}$ define as $\mathcal{H} = \{m(.,\omega) : \omega \in \Omega\}$. at ϵ accuracy with respect to L_2 metric defined over n data points as follows

Definition 2 (Definition 1 Zhang (2002)). *Given observations $\mathcal{D}_n = \{u_1, \ldots, u_n\}$ and vectors $m(\mathcal{D}_n, \omega) = [m(u_1,\omega), \ldots, m(u_n,\omega)] \in \mathbb{R}^n$ parameterized by ω for any $m \in \mathcal{H}$, the L_2 covering number, denoted as $\mathcal{N}_2(\epsilon, \mathcal{H}, \mathcal{D}_n)$, is the minimum number l of a collection of vectors $v_1, \ldots, v_l \in \mathcal{H}$ such that $\forall \omega \in \Omega, v \in \mathcal{H}$ there exists an v_j such that*

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(m\left(u_i,\omega\right) - v_j\left(u_i,\omega\right)\right)^2} \leq \epsilon.$$

We define $\mathcal{N}_p(\epsilon, \mathcal{H}, n) = \sup_{\mathcal{D}_n} \mathcal{N}_p\left(\epsilon, \mathcal{H}, \mathcal{D}_n\right)$.

Note that based on Lemma 3, the solution functions are piecewise affine functions with parameter $\omega \in \Omega$. We provide an important result for L_2 covering number bounds for the set of solution functions $\mathcal{H}$.

Lemma 4. *Consider problem (2). Provided Assumption 1 holds, we bound the L_2 covering number for the set of solution function $\mathcal{H}$ as*

$$\log(\mathcal{N}_2(\epsilon, \mathcal{H}, n)) \leq \sum_{i=n_{eq}}^{p} \binom{n_{ineq}}{i} \frac{2M^2\bar{\Omega}^2\bar{\mathcal{U}}^2}{\epsilon^2},$$

where $M > 0$ is the universal constant, $\bar{\Omega}$, and $\bar{\mathcal{U}}$ are the nonnegative scalars, introduced in Assumption 1(b).

The above bound implies that the number of inequality constraints increases the complexity of the class of solution functions. The proof of the proposed bound consists of two stages: we start by bounding the number of critical regions, then we combine it with the covering number within each region. Also note that for each critical region, the covering number is bounded by a constant depending on the parameter and the input spaces' bounds.

To sum up this section, the critical implications of Lemma 4 is that the solution function of affine parametric variational inequalities is statistically learnable.

5.2 Generalization Bound

In this section, we start by deriving the generalization bound with an improved rate for the function class corresponding to the solution function of the affine variational inequality. Then, we derive the generalization bound for the solution function of the generic variational inequalities. We consider the input space as context $U \subseteq \mathbb{R}^q$ and the output space is the equilibrium actions of players $Y_{u,\omega} \subseteq \mathbb{R}^p$. The available finite set of samples is $\mathcal{D}_n = \{(u_1,y_1), \ldots, (u_n,y_n)\}$ (sampled in i.i.d. fashion). We also specify the loss function $\ell : \mathcal{H} \times Y_{u,\omega} \to \mathbb{R}$, to be L_ℓ Lipschitz smooth, bounded, and nonnegative. Let the empirical risk be denoted by $\hat{R}(m) = \frac{1}{n}\sum_{i=1}^{n} \ell(m(u_i,\omega), y_i)$, and the true risk as $R(m) = \mathbb{E}[\ell(m(u,\omega), y)]$. We start by defining an empirically restricted class.

Definition 3. *For the set of solution function $\mathcal{H}$, loss function ℓ, dataset $\{(u_i,y_i)\}_{i=1}^{n}$, and a nonnegative scalar r, we define the following empirically restricted class*

$$\mathcal{L}_\ell(r) \triangleq \left\{\ell : (u,\omega,y) \to \ell(m(u,\omega),y) : m \in \mathcal{H}, \hat{R}(m) \leq r\right\}.$$

Restricted function class is machinery considered in Bartlett et al. (2005); Srebro et al. (2010) for proving possible fast rates based on local Rademacher complexity. We also use empirically restricted class and provide a generalization bound with further improvement over the existing generalization bound in Srebro et al. (2010) for smooth, nonnegative, bounded loss function.

Theorem 1. *For an L_ℓ-smooth, nonnegative, and bounded loss function such that $|\ell| \leq \ell_{max}$, for any $\delta \in (0,1)$, we have that with probability at least $1-\delta$ over a random sample size n, for any $m \in \mathcal{H}$ corrosponding to the solution function of affine variational inequality*

$$R\left(m\right) \lesssim \hat{R}\left(m\right) + \mathcal{O}\left(\sqrt{\frac{\hat{R}(m)}{n}}\log\left(n\right) + \frac{\log^2\left(n\right)}{n}\right).$$

To prove the rate of generalization bound in the Theorem 1, we start by bounding the Rademacher complexity of the empirically restricted class in terms of L_2 covering number. Then, instead of bounding L_2 covering number of the empirically restricted class in terms of fat-shattering dimension, we bound L_2 covering number of the empirically restricted class directly in terms of L_2 covering number of the hypothesis class. Specifically, by using the fat-shattering dimension, the rate of generalization error is given by

$$R(m) \lesssim \hat{R}(m) + \mathcal{O}\left(\sqrt{\frac{\hat{R}(m)}{n}}\log^{1.5}(n) + \frac{\log^3(n)}{n}\right).$$

Note that the previous generalization bound is obtained by bounding fat-shattering dimension Alon et al. (1997), which leads to bounds worse than the ones that can be obtained in terms of L_2 covering number. Next, we extend the generalization bound for the solution function of generic parametric variation inequalities, where the solution function is a piecewise smooth function.

Theorem 2. *For a function class with L_ℓ-smooth, non-negative, and bounded loss function such that for all u, ω, we have $\|m(u,\omega)\| \le \alpha_0$, than with $1-\delta$ confidence, for any $m \in \mathcal{H}$, the empirical loss is bounded as*

$$R(m) \lesssim \hat{R}(m) + \mathcal{O}\left(\sqrt{\frac{\hat{R}(m)}{n}}\log^{1.5}(n)\alpha_0^{\left(\frac{2k}{n}+1\right)} + \frac{\log^3(n)\alpha_0^{\left(\frac{4k}{n}+2\right)}}{n}\right).$$

The proof of Theorem 2 is based on the result provided in Srebro et al. (2010); we start by bounding the Rademacher complexity by the covering number of the solution functions of generic variational inequalities.

6. ERROR BOUNDS AND CONVERGENCE ANALYSIS

In this section, we discuss the error bounds on the gradients of the decision focus objective in (1), obtained from Algorithm 1 and provide the convergence results in Theorem 3. Note that for notational simplicity, in this section, we assume that $y \in Y_{u,\omega} \subseteq \mathbb{R}^{pn}$ and $F_{u,\omega}(y) : \mathbb{R}^{pn} \to \mathbb{R}^{pn}$ by considering a batch learning set up.

Algorithm 1 Decision-focused iterative implicit gradient

Input: ω_1, scalar $b > 0$, and stepsize β.

1: **for** $k = 1, \ldots, K$ **do**
2: **for** $t = 1, \ldots, T$ **do**
3:

$$z_b^*(y_t, \omega_k) = \underset{z \in y_{u,\omega}}{\operatorname{argmax}} \left\{ \langle F_{u,\omega_k}(y_t), y_t - z\rangle - \tfrac{b}{2}\|y_t - z\|^2 \right\} \quad (9)$$

$$y_{t+1}(u, \omega_k) := z_b^*(y_t, \omega_k). \quad (10)$$

4: **end for**
5: Obtain $\nabla_y z_b(y_k, \omega_k)$ and $\nabla_\omega z_b(y_k, \omega_k)$ through the differentiation of the optimization problem (9).
6: Evaluate $\nabla_\omega y_k$ from (8).
7: Evaluate the gradient for problem (1) objective as

$$\nabla_\omega \Phi(\omega_k) = \nabla_\omega f(y_k(\omega_k), y) + \langle \nabla_y f(y_k(\omega_k), y), \nabla_\omega y_k(\omega_k)\rangle$$

8: Update ω_k using the following gradient update

$$\omega_{k+1} = \mathcal{P}_\Omega\{\omega_k - \beta \nabla_\omega \Phi(\omega_k)\},$$

9: **end for**

We start here by providing a set of standard assumptions on function, f, and on the fixed-point in problem (1) and (7), respectively.

Assumption 2. *Consider problem* (1). *The gradient of the objective function $f(y,\omega)$ has the following properties:*

(a) *We assume the Lipschitz smoothness property for $f(y, \bar{\omega})$ with respect to y, i.e. for any $\bar{\omega} \in \Omega$, and $y_1, y_2 \in Y_{u,\omega}$, we have*

$$\|\nabla_\omega f(y_1, \bar{\omega}) - \nabla_\omega f(y_2, \bar{\omega})\| \le L_{f_\omega}\|y_1 - y_2\|$$
$$\textit{and} \ \ \|\nabla_y f(y_1, \bar{\omega}) - \nabla_y f(y_2, \bar{\omega})\| \le L_{f_y}\|y_1 - y_2\|.$$

(b) *We assume the Lipschitz smoothness for $f(\omega, \bar{y})$ with respect to ω for any $\bar{y} \in Y_{u,\omega}$, i.e. for any $\omega_1, \omega_2 \in \Omega$, and $y \in Y_{u,\omega}$, we have*

$$\|\nabla_\omega f(\bar{y}, \omega_1) - \nabla_\omega f(\bar{y}, \omega_2)\| \le \bar{L}_{f_\omega}\|\omega_1 - \omega_2\|$$
$$\textit{and} \ \ \|\nabla_y f(\bar{y}, \omega_1) - \nabla_y f(\bar{y}, \omega_2)\| \le \bar{L}_{f_y}\|\omega_1 - \omega_2\|.$$

(c) *Function f is $M-$Lipschitz with respect to both parameter $\omega \in \Omega$ and $y \in y_{u,\omega}$.*

(d) *Jacobians $\nabla_\omega z_b^*(y,\omega)$ and $\nabla_y z_b^*(y,\omega)$ are Lipschitz continuous with constants $L_{\omega_{in}}$ and $L_{y_{in}}$, respectively.*

Lipschitzness in Assumption 2(c) of the function f is to ensure the gradient is bounded; also, the other assumptions characterized the smoothness of the objective function. Moreover, we provide an assumption on the fixed-point problem such that the jacobian of the projection operator is smooth with respect to y and ω. We also assume there exists a bound on the update from equation (10), such that $\|y\|$ is bound by C_y, then form Grazzi et al. (2020) for all y the value of $\|\nabla_\omega z_b^*(y,\omega)\|$ is bounded by $C'_{\omega_{in}}$. We start by obtaining the contraction constant of the fixed-point equation. Under Assumption (1) on the mapping $F_{u,\omega}(\cdot)$, if we let $b = \frac{L^2}{\mu}$, then the fixed-point equation obtained from 10 is contraction with constant $q_\omega = \sqrt{1 - \frac{\mu^2}{L^2}} \le 1$.

In the following result, we comment on the Lipschitz continuity of the solution function.

Lemma 5. *Consider problem* (**??**). *The solution function of the VI denoted by $m(u,\omega)$ is Lipschitz continuous with respect to ω with parameter L_S, where $L_S = \frac{C'_{\omega_{in}}}{1-q_\omega}$.*

In Algorithm 1, the solution of the VI is characterized by fixed-point iterations. In the following result, we characterize the tracking error defined by $\|y_t - y^*\|$ after t number of steps, and we show that y_t converges to y^* at least R-linearly.

Lemma 6 (Theorem 12.6.1 Facchinei and Pang (2003))**.** *For $\omega \in \Omega$, the iterative update of y_t, obtained from equation* (9) *in Algorithm 1 converges to the limit point y^* with an R-linear rate, after iteration t of the inner loop in Algorithm 1*

$$\|y_t - y^*\| \le \sqrt{\frac{\phi_{ab}(y_0, \omega)}{\eta_1}} \frac{1}{1 - \sqrt{\frac{\eta_2}{\eta_1 + \eta_2}}} \left(\sqrt{\frac{\eta_2}{\eta_1 + \eta_2}}\right)^t,$$

where η_1, η_2, and δ are the nonnegative scalars such that for any $\omega \in \mathbb{R}^m$, and $y \in \mathbb{R}^{pn}$ we have

$$\phi_{ab}(y_t, \omega) - \phi_{ab}(z_b^*(y_t,\omega), \omega) \ge \eta_1\|y_t - z^*(y_t,\omega)\|^2 \textit{and}$$
$$\min(\phi_{ab}(y_t,\omega), \phi_{ab}(z_b^*(y_t,\omega),\omega)) \le \eta_2\|y_t - z_b^*(y_t,\omega)\|^2$$
for all x with $\|y_t - z_b^*(y_t,\omega)\| \le \delta$.

With the contraction property of the fixed-point equation, we can obtain the error bound between the implicit gradient from iterative update (9) in Algorithm 1 and the actual implicit gradient, which is an essential step to establish the final bound.

Proposition 1 (Proposition 2.1 Grazzi et al. (2020))**.** *Let Assumptions 1, and 2 hold. Then, we have that the error bound of the implicit gradient at the iterative update* *obtained from equation* (9) *after* T *iterations, and the true gradient of the fixed-point of the VI in problem* (2) *as follows*

$$\|\nabla_\omega y_T - \nabla_\omega y^*\| \leq (L_{\omega_{in}} + L_{y_{in}} L_s)\, C_y q_\omega^{T-1} T + L_s q_\omega^T.$$

Next, we will discuss one of the main results of this work. We show that the update from Algorithm 1 converges to local optimum with $\mathcal{O}(1/K)$. Because the loss function is generally nonconvex, we use the gradient norm as the convergence criterion, which is standard in nonconvex optimization.

Theorem 3. *Let Assumption 1 and 2 hold. Consider the update from step 6 of Algorithm 1. We show that sequence* $\{\omega_k\}$ *converges to a stationary point with a rate* $\mathcal{O}(1/K)$ *for* K *iterations*

$$\min_{k\in\{0,\ldots,K\}} \|\nabla_\omega \Phi(\omega_k)\|^2$$
$$\leq \frac{\Phi(\omega_0)-\Phi(\omega_{K+1})}{\beta(\frac{1}{2}-\beta L)K} + M\left(\frac{\frac{\beta}{2}+\beta^2 L_\Phi}{\frac{\beta}{2}-\beta^2 L_\Phi}\right)\left((L_{\omega_{in}} + L_{y_{in}} L_s)\, C_y q_\omega^T (T+1) + L_s q_\omega^{T+1}\right)$$
$$+ \frac{L_{f_\omega} + L_{f_y} L_S}{1-\sqrt{\frac{\eta_2}{\eta_1+\eta_2}}}\left(\frac{\frac{\beta}{2}+\beta^2 L_\Phi}{\frac{\beta}{2}-\beta^2 L_\Phi}\right)\sqrt{\frac{z_b(y_0,\omega_k)}{\eta_1}}\left(\sqrt{\frac{\eta_2}{\eta_1+\eta_2}}\right)^{T+1}.$$

where $L_\Phi \triangleq L_{f_\omega} L_S + \bar{L}_{f_\omega} + L_{f_y} L_S^2 + \bar{L}_{f_y} L_S$.

Note that the last two terms above go to zero with an increasing number of inner iterations T. We hereby focus on establishing the nonasymptotic convergence analysis of the outer-level update $\{\omega_k\}$ from Algorithm 1. Therefore, assuming the inner-level converges R-linearly, we bound the last two terms with ϵ, and we secure the rate of $\mathcal{O}\left(\frac{1}{K}\right)$.

7. NUMERICAL EXPERIMENTS: ESTIMATING UTILITY FUNCTION OF 2 PLAYERS

Consider a Cournot competition of d number of players. In this experiment, we let the number of players $d = 2$ with their combined strategy vector $y = [y^{(i)}; y^{(-i)}] \in \mathbb{R}^4$. Then, a data set is generated $\{(u_i, y_i)\}_{i=1}^{10}$, where the y_i is at Nash equilibrium. After we get hold of the data set, in the actual implementation of Algorithm 1, we assume that utility functions are *not known* to us. We use and modify the ideas of the PICNN to approximate the utility of agents and estimate utility functions using Algorithm 1.

PICNNs, proposed in Amos et al. (2017), are convex neural networks in some of their inputs, provided that the activation functions are convex and non-decreasing. Also, all the weights in the convex path are non-negative. From the inherent convexity assumption of PICNN in y, the map $F_{u,\omega}$, formed by the gradient of utility functions, is a monotone map. We leverage automatic differentiation, a utility already implemented in Tensorflow and PyTorch, to compute the gradients of $\hat{g}_i$ with respect to y_i for all agents. To be in line with Assumption 1, we add a regularization term to $\hat{g}_i(\cdot, u, \omega)$ to be strongly convex, which in turn yields a strongly monotone map $F_{u,\omega}$. In the construction of PICNN, each agent's utility function has 4 layers. Each layer consists of 32 neurons for both convex and nonconvex paths. We use the soft plus activation function, a continuous, differentiable, and convex function.

After constructing the parametric equilibrium map $F_{u,\omega}$, we solve the variation inequality using the fixed point method. The optimization problem in (9) is solved using **CvxpyLayer** Agrawal et al. (2019), iteratively, with ϵ accuracy such that the stop criteria is $\|y_{t+1} - y_t\| \leq \epsilon$. Then, we obtain $\nabla_y z_b(y_k, \omega_k)$ and $\nabla_\omega z_b(y_k, \omega_k)$ and by solving the linear system in (8), we compute $\nabla_\omega y$.

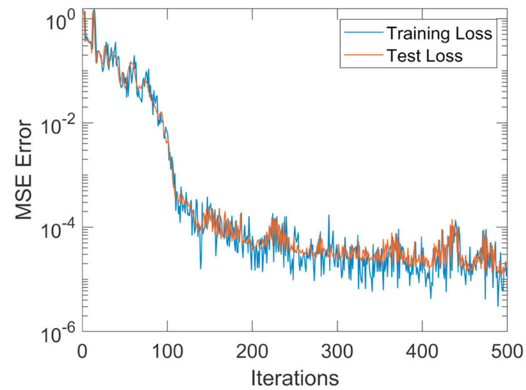

Fig. 2. Mean squared error for training and testing loss on a logarithmic scale.

The utility functions of the agents, represented by PICNNs, are updated using gradient descent with adaptive learning rate β using ADAM optimizer such that the parameters in the convex direction are non-negative.

A test data set of $\{(u_i, y_i)\}_{i=1}^{1000}$ samples is generated to validate the estimated parameters' quality. At each iteration, the mean square error of training and testing error for learning PICNN parameters are reported on a log scale as shown in Fig. 2. We can see that the PICNN with VI has the expression capability to fit the data and predict the equilibrium actions completely. Moreover, the learning with the VI models is explainable and robust. A relatively small number of training samples was enough to capture the utility function and accurately predict players' equilibrium actions.

8. CONCLUSION AND FUTURE DIRECTIONS

In this paper, a decision-focused learning approach was investigated. In order to make evidence-based predictions, An algorithm based on the iterative differentiation strategy to calculate the implicit gradient was proposed. A numerical example was carried out to show the advantages of the proposed approach. In our settings, PICNNs were designed and modified for estimating the utility functions of individual agents, then auto-differentiation was used to construct $F_{u,\omega}$. The following conclusions can be drawn

- The covering number for the set of solution functions of an affine parametric variational inequality can be bounded.
- We derived the implicit gradient using the parametric D-gap function and claimed the existence and uniqueness of the gradient.
- The generalization bound for the set of solution functions with respect to smooth loss function with an improved rate can be established.
- The error bounds on the gradients and the convergence results based on the proposed algorithm was provided.