

# GENERATIVE ADVERSARIAL EQUILIBRIUM SOLVERS

Denizalp Goktas\*

Brown University  
denizalp-goktas@brown.edu

David C. Parkes, Ian Gemp, Luke Marris,

Georgios Piliouras, Romuald Elie, Guy Lever, Andrea Tacchetti

Google DeepMind  
{parkesd, imgemp, marris, parkesd, imgemp, marris, gpil, relie, guylever, atacchet}@google.com

## ABSTRACT

We introduce the use of generative adversarial learning to compute equilibria in general game-theoretic settings, specifically the *generalized Nash equilibrium* (GNE) in *pseudo-games*, and its specific instantiation as the *competitive equilibrium* (CE) in Arrow-Debreu competitive economies. Pseudo-games are a generalization of games in which players’ actions affect not only the payoffs of other players but also their feasible action spaces. Although the computation of GNE and CE is intractable in the worst-case, i.e., PPAD-hard, in practice, many applications only require solutions with high accuracy in expectation over a distribution of problem instances. We introduce *Generative Adversarial Equilibrium Solvers* (GAES): a family of generative adversarial neural networks that can learn GNE and CE from only a sample of problem instances. We provide computational and sample complexity bounds for Lipschitz-smooth function approximators in a large class of concave pseudo-games, and apply the framework to finding Nash equilibria in normal-form games, CE in Arrow-Debreu competitive economies, and GNE in an environmental economic model of the Kyoto mechanism.

## 1 INTRODUCTION

Economic models and equilibrium concepts are critical tools to solve practical problems, including capacity allocation in wireless and network communication (Han et al., 2011; Pang et al., 2008), energy resource allocation (Hobbs & Pang, 2007; Jing-Yuan & Smeers, 1999), and cloud computing (Gutman & Nisan, 2012; Lai et al., 2005; Zahedi et al., 2018; Ardagna et al., 2017). Many of these economic models are instances of what are known as *pseudo-games*, in which the actions taken by each player affect not only the other players’ payoffs, as in games, but also the other players’ strategy sets.<sup>†</sup> The formalism of pseudo-games was introduced by Arrow & Debreu (1954), who used it in studying their foundational microeconomic equilibrium model, the competitive economy.

The standard solution concept for pseudo-games is the *generalized Nash equilibrium (GNE)* (Arrow & Debreu, 1954; Facchinei & Kanzow, 2010a), which is an action profile from which no player can improve their payoff by unilaterally deviating to another action in the space of admissible actions determined by the actions of other players. Important economic models can often be formulated as a pseudo-game, with their set of solutions equal to the set of GNE of the pseudo-game: for instance, the set of *competitive equilibria (CE)* (Walras, 1896; Arrow & Debreu, 1954) of an Arrow-Debreu competitive economy corresponds to the set of GNE of an associated pseudo-game.

A large literature has been devoted to the computation of GNE in certain classes of pseudo-games but unfortunately many algorithms that are guaranteed to converge in theory have in practice been observed to converge slowly in ill-conditioned or large problems or fail numerically (Facchinei &

\*Research conducted while the author was an intern at Google DeepMind.

<sup>†</sup>Also, School of Engineering and Applied Sciences, Harvard University.

<sup>‡</sup>In many games, such as chess, the action taken by one player affects the actions available to the others, but these games are sequential, while in pseudo-games actions are chosen simultaneously. Additionally, even if one constructs a game with payoffs that penalize the players for actions that are not allowed, the NE of the ensuing game will in general not correspond to the GNE of the original pseudo-game and can often be trivial. We refer the reader to Appendix A for a mathematical example.

Kanzow, 2010b; Jordan et al., 2022; Goktas & Greenwald, 2022). Additionally, all known algorithms have hyperparameters that have to be optimized individually for every pseudo-game instance (Facchinei & Kanzow, 2010a), deteriorating the performance of these algorithms when used to solve multiple pseudo-games. These issues point to a need to develop methods to compute GNE, for a distribution of pseudo-games, reliably and quickly.

We reformulate the problem of computing GNE in pseudo-games (and CE in Arrow-Debreu competitive economies) as a learning problem for a generative adversarial network (GAN) called the *Generative Adversarial Equilibrium Solver (GAES)*, consisting of a generator and a discriminator network. The generator takes as input a parametric representation of a pseudo-game, and predicts a solution that consists of a tuple of actions, one per player. The discriminator takes as input both the pseudo-game and the output of the generator, and outputs a best-response for each player, seeking to find a useful unilateral deviation for all players; this also gives the sum of regrets, with which to evaluate the generator (see Figure 1). GAES predicts GNE and CE in batches and in order to minimize the expected exploitability, across a distribution of pseudo-games. GAES amortizes computational cost up-front in training, and allows for near constant evaluation time for inference. Our approach is inspired by previous methods that cast the computation of an equilibrium in normal-form games as an unsupervised learning problem (Duan et al., 2021a; Marris et al., 2022). These methods train a network to predict a strategy profile that minimizes the *exploitability* (i.e., the sum of the players’ payoff-maximizing unilateral deviations w.r.t. a given strategy profile) over a distribution of games. These methods become inefficient in pseudo-games, since in contrast to regular games, the exploitability in pseudo-games (1) requires solving a non-linear optimization problem, (2) is not Lipschitz-continuous, in turn making it hard to learn from samples, and (3) has unbounded gradients, which might lead to exploding gradients in neighborhoods of GNE. Our GAN formulation circumvents all three of these issues.

Although the computation of GNE is intractable in the worst-case (Chen & Deng, 2006; Daskalakis et al., 2009; Chen & Teng, 2009; Vazirani & Yannakakis, 2011; Garg et al., 2017), in practice, applications may only require a solver that gives solutions with high accuracy in expectation over a realistic distribution of problem instances. In particular, a decision maker may need to compute a GNE for a sequence of pseudo-games from some family or *en masse* over a set of pseudo-games sampled from some distribution of interest. An example of such an application is the problem of resource allocation on cloud computing platforms (Hindman et al., 2011; Isard et al., 2009; Burns et al., 2016; Vavilapalli et al., 2013) where a significant number of methods make use of repeated computation of competitive equilibrium (Gutman & Nisan, 2012; Lai et al., 2005; Budish, 2011; Zahedi et al., 2018; Varian, 1973) and generalized Nash equilibrium (Ardagna et al., 2017; 2011b;a; Anselmi et al., 2014). In such settings, as consumers request resources from the platform, the platforms have to find a new equilibrium while handling all numerical failures within a given time frame. Another example is policy makers who often want to understand the equilibria induced by a policy for different distributions of agent preferences in a pseudo-game allowing them to study the impact of a policy for a distribution on different possible kinds of participants. For example, in studying the impact of a protocol such as the *Kyoto joint implementation mechanism* (see Section 5), one might be interested in understanding how the emission levels of countries would change based on their productivity levels (Jones et al., 2000). Other applications include computing competitive equilibria in stochastic market environments. For example, recently proposed algorithms work through a series of equilibrium problems, each of which has to be solved quickly (Liu et al., 2022).

**Contributions.** Earlier approaches (Duan et al., 2021a; Marris et al., 2022) do not extend even to continuous (non pseudo-)games, since evaluating the expected exploitability and its gradient over a distribution of pseudo-games requires solving as many convex programs as examples in the data set. Additionally, in pseudo-games, the exploitability is not Lipschitz-continuous, and thus its gradient is unbounded (Appendix D), hindering the use of standard tools to prove sample complexity and convergence bounds, and making training hard due to exploding gradients. By delegating the task of computing a best-response to a discriminator, our method circumvents the issue of solving a convex

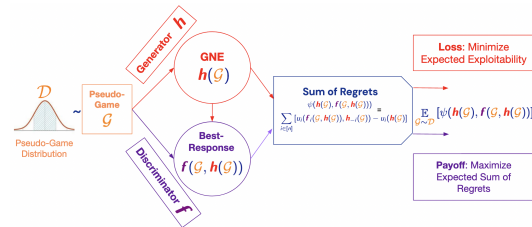


Figure 1: Summary of the Architecture of GAES. By delegating the task of computing a best-response to a discriminator, our method circumvents the issue of solving a convex

program, yielding a training problem given by a min-max optimization problem whose objective is Lipschitz-continuous, for which gradients can be guaranteed to be bounded under standard assumptions on the discriminator and the payoffs of players.

Our approach also extends the class of (non pseudo-)games that can be solved through deep learning methods from normal-form games to simultaneous-move continuous action games, since the non-linear program involved in the computation of exploitability in previous methods makes them inefficient in application to continuous-action games. We prove polynomial-time convergence of our training algorithm for arbitrary Lipschitz-smooth function approximators with the joint action space dimensions larger than parameter space dimensions (Theorem 4.1, Section 4), and provide generalization bounds for arbitrary function approximators (Theorem 4.2, Section 4). Finally, we provide empirical evidence that GAES outperforms state of the art baselines in Arrow-Debreu competitive economies, and show that GAES can replicate existing qualitative analyses for pseudo-games, suggesting that GAES makes predictions that not only have low expected exploitability, but also are qualitatively correct, i.e., close to the true GNE in action space (Section 5).

**Additional related work.** We refer the reader to Appendix B for a survey of methods to compute GNEs, and to Appendix C for a survey of applications of GNE. Our contributions generally relate to a line of work on *differentiable economics*, which seeks to use methods of neural computation for problems of economic design and equilibrium computation. In regard to finding optimal economic designs, deep learning has been used for problems of auction design (Dütting et al., 2019; Curry et al., 2022c; Iacchetti et al., 2019; Curry et al., 2022a; Gemp et al., 2022; Rahme et al., 2020; Duan et al., 2022) and matching (Ravindranath et al., 2021). In regard to solving for equilibria, some recent works have tried to solve for Nash equilibria in auctions (Heidekrüger et al., 2019; Bichler et al., 2021), and dynamic stochastic general equilibrium models (Curry et al., 2022b; Chen et al., 2021; Hill et al., 2021).

## 2 PRELIMINARIES

**Notation.** All notation for variable types, e.g., vectors, are clear from context, if any confusions arise see Appendix E. We denote the set of integers  $\{0, \dots, n-1\}$  by  $[n]$ , the set of natural numbers by  $\mathbb{N}$ , the set of real numbers by  $\mathbb{R}$ , and the positive and strictly positive elements of a set by a subscript  $+$  and  $++$ , e.g.,  $\mathbb{R}_+$  and  $\mathbb{R}_{++}$ . We denote by  $\Delta_n = \{\mathbf{x} \in \mathbb{R}_+^n \mid \sum_{i=1}^n x_i = 1\}$ , and by  $\Delta(A)$ , the set of probability measures on the set  $A$ .

**Pseudo-Games.** A (concave) pseudo-game (Arrow & Debreu, 1954)  $\mathcal{G} \doteq (n, \mathcal{A}, \mathcal{X}^{\mathcal{G}}, \mathbf{h}^{\mathcal{G}}, \mathbf{u}^{\mathcal{G}})$ , denoted  $(n, \mathcal{A}, \mathcal{X}, \mathbf{h}, \mathbf{u})$  when clear from context, comprises  $n \in \mathbb{N}_+$  players, where player  $i \in [n]$  chooses an action  $\mathbf{a}_i$  from a non-empty, compact, and convex action space  $\mathcal{A}_i \subset \mathbb{R}^m$ . We denote the players' joint action space by  $\mathcal{A} = \times_{i \in [n]} \mathcal{A}_i \subset \mathbb{R}^{nm}$ . Each player  $i \in [n]$  aims to maximize their continuous payoff,  $u_i : \mathcal{A} \rightarrow \mathbb{R}$ , which is concave in  $\mathbf{a}_i$ , by choosing a feasible action from a set of actions,  $\mathcal{X}_i(\mathbf{a}_{-i}) \subseteq \mathcal{A}_i$ , this depending on the actions  $\mathbf{a}_{-i} \in \mathcal{A}_{-i} \subset \mathbb{R}^{(n-1)m}$  of the other players. Here,  $\mathcal{X}_i : \mathcal{A}_{-i} \rightrightarrows \mathcal{A}_i$  is a non-empty, continuous, compact- and convex-valued (feasible) action correspondence. It is this dependence on each others' actions that makes this a pseudo-game, and not just a game. In applications, we will represent  $\mathcal{X}_i$  as  $\mathcal{X}_i(\mathbf{a}_{-i}) = \{\mathbf{a}_i \in \mathcal{A}_i \mid h_{ip}(\mathbf{a}_i, \mathbf{a}_{-i}) \geq \mathbf{0}, \text{ for all } p \in [d]\}$ , where for all  $i \in [n]$ , and  $p \in [d]$ ,  $h_{ip}$  is a continuous and quasi-concave function in  $\mathbf{a}_i$ , which defines the constraints. We denote the product (feasible) action correspondence by  $\mathcal{X}(\mathbf{a}) = \times_{i \in [n]} \mathcal{X}_i(\mathbf{a}_{-i})$ , which we note is guaranteed to be continuous, and non-empty-, compact-valued. We denote  $\mathcal{X}$  the set of jointly feasible strategies, i.e.,  $\mathcal{X} = \{\mathbf{a} \in \mathcal{A} \mid h_{ip}(\mathbf{a}) \geq \mathbf{0}, \forall i \in [n], p \in [d]\}$ . We denote the class of all pseudo-games by  $\Gamma$ <sup>2</sup>.

Given a pseudo-game  $\mathcal{G}$ , a generalized Nash equilibrium (GNE) is an action profile  $\mathbf{a}^* \in \mathcal{X}$ , s.t. for all  $i \in [n]$  and  $\mathbf{a}_i \in \mathcal{X}_i(\mathbf{a}_{-i}^*)$ ,  $u_i(\mathbf{a}^*) \geq u_i(\mathbf{a}_i, \mathbf{a}_{-i}^*)$ . An equilibrium mapping,  $\mathbf{h} : \Gamma \rightarrow \mathcal{X}$  is a mapping that takes as input a pseudo-game  $\mathcal{G} \in \Gamma$  and outputs a GNE,  $\mathbf{h}(\mathcal{G})$ , for that game. Given a pseudo-game  $\mathcal{G}$ , we define the regret for player  $i \in [n]$  for action  $\mathbf{a}_i$  as compared to another action  $\mathbf{b}_i$ , given the action profile  $\mathbf{a}_{-i}$  of other players, as  $\text{Regret}_i^{\mathcal{G}}(\mathbf{a}_i, \mathbf{b}_i; \mathbf{a}_{-i}) \doteq u_i^{\mathcal{G}}(\mathbf{b}_i, \mathbf{a}_{-i}) - u_i^{\mathcal{G}}(\mathbf{a}_i, \mathbf{a}_{-i})$ . Additionally, the cumulative regret, between two action profiles  $\mathbf{a} \in \mathcal{A}$  and  $\mathbf{b} \in \mathcal{A}$  is given by

<sup>2</sup>A game (Nash, 1950) is a pseudo-game where, for all players  $i \in [n]$ ,  $\mathcal{X}_i$  is a constant correspondence with value  $\mathcal{A}_i$ . A discrete action game is a game where  $\mathcal{A}_i = \Delta_m$  and  $u_i$  is multilinear for all  $i \in [n]$ .

$\psi(\mathbf{a}, \mathbf{b}) \doteq \sum_{i \in [n]} \text{Regret}_i^{\mathcal{G}}(\mathbf{a}_i, \mathbf{b}_i; \mathbf{a}_{-i})$ . Further, the *exploitability* (Nikaido & Isoda, 1955), of an action profile  $\mathbf{a}$  is defined as  $\varphi^{\mathcal{G}}(\mathbf{a}) \doteq \sum_{i \in [n]} \max_{\mathbf{b}_i \in \mathcal{X}_i(\mathbf{a}_{-i})} \text{Regret}_i^{\mathcal{G}}(\mathbf{a}_i, \mathbf{b}_i; \mathbf{a}_{-i})$ . Note that an action profile  $\mathbf{a}^*$  is a GNE iff  $\varphi^{\mathcal{G}}(\mathbf{a}^*) = 0$ .

**Mathematical preliminaries.** For any function  $f : \mathcal{X} \rightarrow \mathcal{Y}$ , we denote its Lipschitz-continuity constant by  $\ell_f$ . For two arbitrary sets  $\mathcal{H}, \mathcal{H}' \subset \mathcal{F}$ , the set  $\mathcal{H}'$  *r-covers*  $\mathcal{H}$  (w.r.t. some norm  $\|\cdot\|$ ) if for any  $\mathbf{h} \in \mathcal{H}$  there exists  $\mathbf{h}' \in \mathcal{H}'$  such that  $\|\mathbf{h} - \mathbf{h}'\| \leq r$ . The *r-covering number*,  $\rho(\mathcal{H}, r)$ , of a set  $\mathcal{H}$  is the cardinality of the smallest set  $\mathcal{H}' \subset \mathcal{F}$  that *r-covers*  $\mathcal{H}$ . A set  $\mathcal{H}$  is said to have a *bounded covering number*, if for all  $r \in \mathbb{R}_+$ , we have that the logarithm of its covering number is polynomially bounded in  $1/r$ , that is  $\log(\rho(\mathcal{H}, r)) \leq \text{poly}(1/r)$ . Additional background can be found in Appendix E.

### 3 GENERATIVE ADVERSARIAL LEARNING OF EQUILIBRIUM MAPPINGS

In this section, we revisit previous formulations of the problem of learning an equilibrium mapping, discuss the computational difficulties associated with these formulations when used to learn GNE, and introduce our generative adversarial learning formulation.

As creating a sufficiently diverse sample of (pseudo-game, GNE) pairs, while performing adequate equilibrium selection, is intractable both theoretically and computationally, we forgo of supervised learning methods, and formulate the equilibrium mapping learning problem as an *unsupervised* learning problem, following the approach adopted by Marris et al. (2022); Duan et al. (2021b) for finding Nash equilibria. Given a hypothesis class  $\mathcal{H} \subseteq \mathcal{X}^I$ , and a distribution over pseudo-games  $\mathcal{D} \in \Delta(\Gamma)$ , the *unsupervised learning problem* for an equilibrium mapping consists of finding a hypothesis  $\mathbf{h}^* \in \arg \min_{\mathbf{h} \in \mathcal{H}} \mathbb{E}_{\mathcal{G} \sim \mathcal{D}} [\ell(\mathcal{G}, \mathbf{h}(\mathcal{G}))]$  where  $\ell : \Gamma \times \mathcal{A} \rightarrow \mathbb{R}$  is a loss function that outputs the distance of  $\mathbf{a} \in \mathcal{A}$  from a GNE, such that for any pseudo-game  $\mathcal{G} \in \Gamma$ ,  $\ell(\mathcal{G}, \mathbf{a}^*) = 0$  iff  $\mathbf{a}^*$  is a GNE of  $\mathcal{G}$ . In particular, Marris et al. (2022); Duan et al. (2021b) suggest to use exploitability as the loss function. However, a number of issues arise when trying to minimize the expected exploitability over a distribution of pseudo-games:

- (1) Computing the gradient of the exploitability, when it exists, for even only one pseudo-game requires solving a concave maximization problem (this reasoning also applies to continuous games).
- (2) The exploitability in pseudo-games, is in general not Lipschitz-continuous (unlike in regular games), even when payoffs are Lipschitz-continuous, since the inputs of the exploitability parameterize the constraints in the optimization problem defining each player’s maximal regret computation. This makes it unclear how to efficiently approximate  $\mathbb{E}_{\mathcal{G} \sim \mathcal{D}} [\varphi^{\mathcal{G}}(\mathbf{a})]$  from samples, without knowledge of the distribution.
- (3) The exploitability in pseudo-games is absolutely continuous and hence differentiable almost everywhere (Afriat (1971)), but in contrast to games, the gradients cannot be bounded. This in turn precludes the convergence of first-order methods.<sup>3</sup>

To address the aforementioned issues, we propose a generative adversarial learning formulation of the associated unsupervised learning problem for equilibrium mappings. The formulation relies on the following observation, whose proof is deferred to Appendix F: the exploitability can be computed *ex post* after computing the expected cumulative regret by optimizing over the space of best-response functions from pseudo-games to actions, rather than the space of actions individually for every pseudo-game.

**Observation 1.** For any  $\mathcal{D} \in \Delta(\Gamma)$ , we have  $\min_{\mathbf{h} \in \mathcal{X}^I} \mathbb{E}_{\mathcal{G} \sim \mathcal{D}} [\varphi^{\mathcal{G}}(\mathbf{h}(\mathcal{G}))] = \min_{\mathbf{h} \in \mathcal{X}^I} \max_{\mathbf{f} \in \mathcal{A}^I: \forall \mathcal{G} \in \Gamma, \mathbf{f}(\mathcal{G}) \in \mathcal{X}^{\mathcal{G}}(\mathbf{h}(\mathcal{G}))} \mathbb{E}_{\mathcal{G} \sim \mathcal{D}} [\psi^{\mathcal{G}}(\mathbf{h}(\mathcal{G}), \mathbf{f}(\mathcal{G}))]$

By Arrow-Debreu’s lemma on abstract economies (Arrow & Debreu (1954)),  $\mathbf{h}^*$  is guaranteed to exist and is an equilibrium mapping iff  $\mathbf{h}^* \in \arg \min_{\mathbf{h} \in \mathcal{X}^I} \max_{\mathbf{f} \in \mathcal{A}^I: \forall \mathcal{G} \in \Gamma, \mathbf{f}(\mathcal{G}) \in \mathcal{X}^{\mathcal{G}}(\mathbf{h}(\mathcal{G}))} \mathbb{E}_{\mathcal{G} \sim \mathcal{D}} [\psi^{\mathcal{G}}(\mathbf{h}(\mathcal{G}), \mathbf{f}(\mathcal{G}))]$ .

<sup>3</sup>We refer the reader to Appendix D for an example in which exploitability is not Lipschitz-continuous and has unbounded gradients.

This problem formulation allows us to overcome issues (1) and (2). For (1), rather than solve a concave program to compute the exploitability for each pseudo-game and action profile, we can learn a function that maps action profiles to their associated best-response profiles (see for example Lanctot et al. (2017) for training best-response oracles). For (2), the objective function in the Equation on the right hand side of Observation 1, i.e., the cumulative regret, is Lipschitz-continuous in the action profiles when payoff functions are, which opens the doors to use standard proof techniques to learn the objective from a polynomial sample of pseudo-games.

Still, the gradient of  $\max_{\mathbf{f}} \mathbb{E}_{\mathcal{G} \sim \mathcal{D}} [\psi^{\mathcal{G}}(\mathbf{h}(\mathcal{G}), \mathbf{f}(\mathcal{G}; \mathbf{h}(\mathcal{G})))]$  with respect to  $\mathbf{h}$  is in general unbounded even when it exists, due to the constraint  $\forall \mathcal{G} \in \Gamma, \mathbf{f}(\mathcal{G}) \in \mathcal{X}(\mathbf{h}(\mathcal{G}))$ . However, since any solution  $\mathbf{f}^*(\mathcal{G}, \mathbf{h}(\mathcal{G}))$  to the inner optimization problem  $\max_{\mathbf{f} \in \mathcal{A}^{\Gamma}: \forall \mathcal{G} \in \Gamma, \mathbf{f}(\mathcal{G}) \in \mathcal{X}(\mathbf{h}(\mathcal{G}))} \mathbb{E} [\psi^{\mathcal{G}}(\mathbf{h}(\mathcal{G}), \mathbf{f}(\mathcal{G}))]$  is implicitly parameterized by the choice of equilibrium mapping  $\mathbf{h}$ , we can represent this dependence explicitly in the optimization problem, and restrict our selection of  $\mathbf{f}$  to a continuously differentiable hypothesis class  $\mathcal{F} \subset \mathcal{A}^{\Gamma \times \mathcal{X}}$ , and overcome issue (3).

With these observations in mind, given hypothesis classes  $\mathcal{H} \subset \mathcal{X}^{\Gamma}$ , and  $\mathcal{F} \subset \mathcal{A}^{\Gamma \times \mathcal{X}}$ , the *generative adversarial learning problem* is to find a tuple  $(\mathbf{h}^*, \mathbf{f}^*) \in \mathcal{H} \times \mathcal{F}$  that consists of a *generator* and *discriminator* to solve the following optimization problem:

$$\min_{\mathbf{h} \in \mathcal{H}} \max_{\mathbf{f} \in \mathcal{F}: \forall \mathcal{G} \in \Gamma, \mathbf{f}(\mathcal{G}; \mathbf{h}(\mathcal{G})) \in \mathcal{X}^{\mathcal{G}}(\mathbf{h}(\mathcal{G}))} \mathbb{E}_{\mathcal{G} \sim \mathcal{D}} [\psi^{\mathcal{G}}(\mathbf{h}(\mathcal{G}), \mathbf{f}(\mathcal{G}; \mathbf{h}(\mathcal{G})))]. \quad (1)$$

This problem can be interpreted as a zero-sum game between the generator and the discriminator. The generator takes as input a parametric representation of a pseudo-game, and predicts a solution that consists of an *action profile*, i.e., a tuple of actions, one per agent. The discriminator takes the game and the output of the generator as input, and outputs a best-response for each agent (Figure 1). The optimal mappings  $(\mathbf{h}^*, \mathbf{f}^*)$  for Equation (1) are then called the *Generative Adversarial Equilibrium Solver (GAES)*.

## 4 CONVERGENCE AND SAMPLE COMPLEXITY

For training, we propose a stochastic variant of the multistep gradient descent ascent algorithm (Nouiehed et al., 2019; Sanjabi et al., 2018), which we call *stochastic exploitability descent*. Our algorithm computes the optimal generator and discriminator by estimating the gradient of the expected cumulative regret and exploitability on a training set of pseudo-games.

---

### Algorithm 1 Stochastic Exploitability Descent

---

**Inputs:**  $\mathcal{B}, \eta_{\mathbf{h}}, \eta_{\mathbf{f}}, T_{\mathbf{h}}, T_{\mathbf{f}}, \mathbf{w}^{\mathbf{h},(0)}, \mathbf{w}^{\mathbf{f},(0)}$

**Outputs:**  $(\mathbf{w}^{\mathbf{h},(t)}, \mathbf{w}^{\mathbf{f},(t)})_{t=0}^{T_{\mathbf{h}}}$

- 1: **for**  $t = 0, \dots, T_{\mathbf{h}} - 1$  **do**
  - 2:   Receive batch  $\mathcal{B}^{(t)} \subset \mathcal{S}$ .
  - 3:    $\mathbf{w}^{\mathbf{h},(t+1)} = \mathbf{w}^{\mathbf{h},(t)} - \eta_{\mathbf{h}}^{(t)} \left( 1/|\mathcal{B}_{\mathbf{h}}^{(t)}| \sum_{\mathcal{G} \in \mathcal{B}_{\mathbf{h}}^{(t)}} \left[ \nabla_{\mathbf{w}^{\mathbf{h}}} \hat{\psi}(\mathbf{w}^{\mathbf{h},(t)}, \mathbf{w}^{\mathbf{f},(t)}) \right] \right)$
  - 4:   **for**  $s = 0, \dots, T_{\mathbf{f}} - 1$  **do**
  - 5:     Receive batch  $\mathcal{B}^{(s)} \subset \mathcal{S}$ .
  - 6:      $\mathbf{w}^{\mathbf{f}} = \mathbf{w}^{\mathbf{f}} + \eta_{\mathbf{f}}^{(s)} \left( 1/|\mathcal{B}_{\mathbf{f}}^{(s)}| \sum_{\mathcal{G} \in \mathcal{B}_{\mathbf{f}}^{(s)}} \nabla_{\mathbf{w}^{\mathbf{f}}} \hat{\psi}(\mathbf{w}^{\mathbf{h},(t)}, \mathbf{w}^{\mathbf{f}}) \right)$
  - 7:   **end for**
  - 8:    $\mathbf{w}^{\mathbf{f},(t+1)} = \mathbf{w}^{\mathbf{f}}$
  - 9: **end for**
  - 10: Return  $(\mathbf{w}^{\mathbf{h},(t)}, \mathbf{w}^{\mathbf{f},(t)})_{t=0}^{T_{\mathbf{h}}}$
- 

**Training Algorithm.** For purposes of applicability, going forward, we will assume that we have access to the distribution of pseudo-games  $\mathcal{D}$  only indirectly through a *training set*  $\mathcal{S} \sim \mathcal{D}$  of  $k \in \mathbb{N}_+$  sampled pseudo-games. Additionally, we will assume that the generator  $\mathbf{h} \in \mathcal{H}$  and discriminator



$\mathbf{f} \in \mathcal{F}$  are parameterized by parameter vectors,  $\mathbf{w}^h, \mathbf{w}^f \in \mathbb{R}^\omega$  such that for all pseudo-games  $\mathcal{G} \in \Gamma$ , and weight vectors  $\mathbf{w} \in \mathbb{R}^\omega$ ,  $\mathbf{h}(\mathcal{G}; \mathbf{w}^h) \in \mathcal{X}$  and  $\mathbf{f}(\mathcal{G}; \mathbf{h}(\mathcal{G}; \mathbf{w}^h), \mathbf{w}^f) \in \mathcal{A}$ .

For notational simplicity, we define the *expected cumulative regret* and the *empirical cumulative regret*, respectively, as  $\bar{\psi}(\mathbf{w}^h, \mathbf{w}^f) \doteq \mathbb{E} [\psi^{\mathcal{G}}(\mathbf{h}(\mathcal{G}; \mathbf{w}^h), \mathbf{f}(\mathcal{G}; \mathbf{h}(\mathcal{G}; \mathbf{w}^h), \mathbf{w}^f))]$ , and  $\hat{\psi}(\mathbf{w}^h, \mathbf{w}^f) \doteq \mathbb{E} [\psi^{\mathcal{G}}(\mathbf{h}(\mathcal{G}; \mathbf{w}^h), \mathbf{f}(\mathcal{G}; \mathbf{h}(\mathcal{G}; \mathbf{w}^h), \mathbf{w}^f))]$  where the expectation is over the distribution of pseudo-games  $\mathcal{G} \sim \mathcal{D}$  and the uniform distribution over the training set,  $\mathcal{G} \sim \text{unif}(\mathcal{S})$  respectively. Similarly, we define the *expected exploitability* and the *empirical exploitability* respectively as:

$$\bar{\varphi}(\mathbf{w}^h) \doteq \max_{\substack{\mathbf{w}^f \in \mathbb{R}^\omega: \forall \mathcal{G} \in \Gamma, \\ \mathbf{f}(\mathcal{G}; \mathbf{w}^f) \in \mathcal{X}(\mathbf{h}(\mathcal{G}; \mathbf{w}^h))}} \bar{\psi}(\mathbf{w}^h, \mathbf{w}^f) \quad \Bigg| \quad \hat{\varphi}(\mathbf{w}^h) \doteq \max_{\substack{\mathbf{w}^f \in \mathbb{R}^\omega: \forall \mathcal{G} \in \Gamma, \\ \mathbf{f}(\mathcal{G}; \mathbf{w}^f) \in \mathcal{X}(\mathbf{h}(\mathcal{G}; \mathbf{w}^h))}} \hat{\psi}(\mathbf{w}^h, \mathbf{w}^f),$$

Putting this together, our training problem becomes:

$$\min_{\mathbf{w}^h \in \mathbb{R}^\omega} \max_{\mathbf{w}^f \in \mathbb{R}^\omega: \forall \mathcal{G} \in \Gamma, \mathbf{f}(\mathcal{G}; \mathbf{h}(\mathcal{G}; \mathbf{w}^h), \mathbf{w}^f) \in \mathcal{X}(\mathbf{h}(\mathcal{G}; \mathbf{w}^h))} \hat{\psi}(\mathbf{w}^h, \mathbf{w}^f). \quad (2)$$

We propose Algorithm 1 to solve this optimization problem. This is a nested stochastic gradient descent-ascent algorithm, which for each generator descent step runs multiple stochastic gradient ascent steps on the weights of the discriminator to approximate the empirical cumulative regret by processing the pseudo-games in the training set in *batches*, i.e., as mutually exclusive subsets of the training set. After the stochastic gradient ascent steps are done, the algorithm then takes a step of stochastic gradient descent on the empirical exploitability w.r.t. the weights of the generator using the discriminator’s weights computed by the stochastic gradient ascent steps to compute the gradient. Note that when the hypothesis class of the discriminator is assumed to contain only differentiable functions, the gradient of the generator with respect to its weights exist, and they are given by the implicit function theorem.

**Convergence and generalization bounds.** We give assumptions under which our algorithm converges to a stationary point of the empirical exploitability in polynomial-time.

**Assumption 1.** For any player  $i \in [n]$  and  $\mathcal{G} \in \text{supp}(\mathcal{D})$ : 1. (Lipschitz-smoothness) their payoff  $u_i^{\mathcal{G}}$  is  $\ell_{\nabla_u}$ -Lipschitz smooth, 2. (Strong concavity) their payoff  $u_i^{\mathcal{G}}$  is  $\mu_{u_i}$ -strongly-concave in  $\mathbf{a}_i$ , and 3. (Lipschitz-smooth hypothesis classes) For all  $\mathbf{h} \in \mathcal{H} \subset \mathcal{X}^{\Gamma \times \mathbb{R}^\omega}$ ,  $\mathbf{f} \in \mathcal{F} \subset \mathcal{A}^{\Gamma \times \mathcal{X} \times \mathbb{R}^\omega}$ ,  $\mathbf{h}(\mathcal{G}; \cdot)$ ,  $\mathbf{f}(\mathcal{G}; \cdot)$  are injective, and Lipschitz-smooth functions.

We note that strong concavity can further be weakened to concavity, and our results can directly extend to any concave pseudo-game by replacing the cumulative regret  $\psi$  with its regularized counterpart  $\psi_\alpha^{\mathcal{G}}(\mathbf{a}, \mathbf{b}) \doteq \psi_\alpha(\mathbf{a}, \mathbf{b}) - \alpha/2 \|\mathbf{a} - \mathbf{b}\|_2^2$ , which is strongly-concave in  $\mathbf{b}$  without modifying the solutions of the optimization (Von Heusinger & Kanzow, 2009). However, as in our experiments non-regularized cumulative regret performed better, for consistency, we present our theoretical results under the strong concavity assumption. That said, strong concavity in each player’s action is a much weaker assumption than strong monotonicity of the pseudo-game which is commonly used in the GNE literature (Jordan et al., 2022). For omitted definitions, and proofs/results we refer the reader to Appendix E and Appendix F respectively.

Theorem 4.1 tells us that our algorithm converges to a stationary point of the empirical exploitability at a  $\tilde{O}(1/\sqrt{T_h})$  rate, up to an error term that depends linearly on the distance,  $\varepsilon$ , of the discriminator computed by the algorithm w.r.t. to the optimal discriminator. A smaller  $\varepsilon$  results in higher accuracy, at the expense of a longer run time. <sup>4</sup>

**Theorem 4.1** (Convergence to Stationary Point). Suppose that Assumption 1 holds. Let  $\varepsilon > 0$ . If Algorithm 1 is run with inputs satisfying  $\eta_h^{(t)} \in \Theta(1/\sqrt{t})$ ,  $\eta_f^{(s)} \in \Theta(1/s)$ ,  $T_h \in \mathbb{N}_{++}$ , and  $T_f \in O(1/\varepsilon)$ , for all  $t \in [T_h]$ ,  $s \in [T_f]$ . Then, the outputs  $(\mathbf{w}^{h,(t)}, \mathbf{w}^{f,(t)})_{t=0}^{T_h}$  satisfy

$$\mathbb{E} \left[ \min_{t=0, \dots, T_h-1} \|\nabla_{\mathbf{a}} \hat{\varphi}(\mathbf{w}^{h,(t)})\|_2^2 \right] \in O \left( \frac{\log(T_h)}{\sqrt{T_h}} + \varepsilon \right).$$

<sup>4</sup>Our min-max problem is non-convex-PL, for which single loop stochastic gradient descent ascent algorithms are not guaranteed to converge to stationary points of the exploitability (Lin et al., 2020; Daskalakis et al., 2009). Additionally, our algorithm’s computational complexity  $\tilde{O}(1/\varepsilon^3)$  is orders of magnitude faster than the best known complexity of  $\tilde{O}(1/\varepsilon^6)$  for such problems (Lin et al., 2020).

We also give a sample complexity result showing how cumulative regret can be approximated with a sample of pseudo-games that is polynomial in the parameters of the game distribution,  $1/\varepsilon$  and  $1/\delta$ . The novelty of the result comes from the context: we mentioned earlier that expected exploitability need not be Lipschitz-continuous, making it hard to use any standard and simple machinery to prove a sample complexity bound. However, by reframing this problem as one of learning the expected cumulative regret, which is Lipschitz-continuous, we can obtain the result.

**Theorem 4.2** (Sample Complexity of Expected Cumulative Regret). *Suppose that part 1. of Assumption 1 holds. Let  $\varepsilon, \delta \in (0, 1)$ ,  $r \in O(\varepsilon/\ell_\psi)$  and consider hypothesis classes  $\mathcal{H}, \mathcal{F}$  given by the minimum  $r$ -covering sets of  $\mathcal{X}^\Gamma, \mathcal{A}^{\mathcal{X} \times \Gamma}$  respectively. For any  $\mathbf{h} \in \mathcal{X}^\Gamma$  and  $\mathbf{f} \in \mathcal{A}^{\mathcal{X} \times \Gamma}$ , take the closest hypotheses  $\mathbf{h}^r \in \arg \min_{\mathbf{h}' \in \mathcal{H}} \|\mathbf{h} - \mathbf{h}'\|$  and  $\mathbf{f}^r \in \arg \min_{\mathbf{f}' \in \mathcal{F}} \|\mathbf{f} - \mathbf{f}'\|$  to respectively. Then, for any pseudo-game distribution  $\mathcal{D} \in \Delta(\Gamma)$  with compact support, with probability at least  $1 - \delta$  over draws of the training set  $\mathcal{S} \sim \mathcal{D}^k$  with  $k \in \Omega(1/\varepsilon^2 \log(\delta^{-1} \rho(\mathcal{H}, \varepsilon/\ell_\psi) \rho(\mathcal{F}, \varepsilon/\ell_\psi)))$ :*

$$|\mathbb{E}_{\mathcal{G} \sim \text{unif}(\mathcal{S})} [\psi^{\mathcal{G}}(\mathbf{h}^r(\mathcal{G}), \mathbf{f}^r(\mathcal{G}, \mathbf{h}^r(\mathcal{G})))] - \mathbb{E}_{\mathcal{G} \sim \mathcal{D}} [\psi^{\mathcal{G}}(\mathbf{h}(\mathcal{G}), \mathbf{f}(\mathcal{G}, \mathbf{h}(\mathcal{G})))]| \leq \varepsilon.$$

## 5 EXPERIMENTAL RESULTS

**Arrow-Debreu exchange economies.**<sup>5</sup> Our first set of experiments aim to solve CE in Arrow-Debreu exchange economies (Arrow & Debreu, 1954). The difficulty in solving the pseudo-game associated with Arrow-Debreu exchange economies—hereafter *exchange economies*—arises from the fact that it does not fit into any well-defined categories of pseudo-games, e.g., monotone or jointly convex, for which there are algorithms that converge to GNEs.

An *exchange economy*  $(\mathbf{u}, \mathbf{E})$  consists of a finite set of  $m \in \mathbb{N}_+$  goods and  $n \in \mathbb{N}_+$  consumers (or traders). Every consumer  $i \in [n]$  has a set of possible consumptions  $\mathcal{X}_i \subseteq \mathbb{R}_+^m$ , an endowment of goods  $\mathbf{e}_i = (e_{i1}, \dots, e_{im}) \in \mathbb{R}^m$  and a utility function  $u_i : \mathbb{R}^m \rightarrow \mathbb{R}$ . We denote  $\mathbf{E} = (e_1, \dots, e_n)^T$ . Any exchange economy can be formulated as a pseudo-game whose set of GNE is equal to the set of competitive equilibria (CE)<sup>6</sup> of the original economy (Arrow & Debreu, 1954). This pseudo-game consists of  $n + 1$  agents, who correspond to the  $n$  buyers and a seller. The pseudo-game is given by the following optimization problem, for each buyer  $i \in [n]$ , and the seller, respectively:

$$\max_{\mathbf{x}_i \in \mathcal{X}_i: \mathbf{x}_i \cdot \mathbf{p} \leq \mathbf{e}_i \cdot \mathbf{p}} u_i(\mathbf{x}_i) \quad \left| \quad \max_{\mathbf{p} \in \Delta_m} \mathbf{p} \cdot \left( \sum_{i \in [n]} \mathbf{x}_i - \sum_{i \in [n]} \mathbf{e}_i \right).$$

Let  $\mathbf{v}_i \in \mathbb{R}_+^m$ ,  $\boldsymbol{\rho} \in (-\infty, 1]^n$ , be a vector of parameters for the utility function of buyer  $i \in [n]$ . In our experiments, we consider the following utility function classes: *Linear*:  $u_i(\mathbf{x}_i) = \sum_{j \in [m]} v_{ij} x_{ij}$ , *Cobb-Douglas*:  $u_i(\mathbf{x}_i) = \prod_{j \in [m]} x_{ij}^{v_{ij}}$ , *Leontief*:  $u_i(\mathbf{x}_i) = \min_{j \in [m]} \{x_{ij}/v_{ij}\}$ , and *constant elasticity of substitution (CES)*:  $u_i(\mathbf{x}_i) = (\sum_{j \in [m]} v_{ij} x_{ij}^\rho)^{1/\rho}$ . When we take  $\rho = 1$ ,  $\rho \rightarrow 0$ , and  $\rho \rightarrow -\infty$  for CES utilities, we obtain linear, Cobb-Douglas, and Leontief utilities respectively. We denote  $\mathbf{V} = (\mathbf{v}_1, \dots, \mathbf{v}_n)^T$ . As is standard in the literature (Cheung et al., 2013; Brânzei et al., 2021), we assume that for all buyers  $i \in [n]$ ,  $\mathcal{X}_i = \mathbb{R}_+^m$ . Once a utility function class is fixed, an exchange economy is referred to with the name of the utility function, and can be sampled as a tuple  $(\mathbf{V}, \mathbf{E}) \in \mathbb{R}^{n \times m} \times \mathbb{R}^{n \times m}$  for linear, Cobb-Douglas, and Leontief exchange economies, and as a tuple  $(\mathbf{V}, \boldsymbol{\rho}, \mathbf{E}) \in \mathbb{R}^{n \times m} \times \mathbb{R}^n \times \mathbb{R}^{n \times m}$  for CES exchange economies. For CES exchange economies, we have a *gross substitute (GS)* and *gross complements (GC)* CES economy, either when  $\rho_i \geq 0$  for all buyers or  $\rho_i < 0$  for all buyers, respectively. Otherwise, this is a mixed CES economy.

**Baselines.** For special cases of exchange economies, the computation of CE is well-studied (e.g., Bei et al. (2015)), allowing us to compare the performance of GAES to known specialized methods. We benchmark GAES to *tâtonnement* (Walras, 1896), which is an auction-like algorithm that is guaranteed to converge for CES utilities with  $\boldsymbol{\rho} \in [0, 1]^n$  (Bei et al., 2015), including Cobb-Douglas and excluding Linear utilities. We also benchmark to *exploitability descent* (Goktas & Greenwald, 2022). For each of these baselines, we run an extensive grid search over decreasing learning rates during validation (see Appendix G.2). Each baseline was run to convergence.

We report the results of two experiments. First, we run our algorithms in linear, Cobb-Douglas, Leontief, GS CES, GC CES, and mixed CES exchange economies (we defer the results from the GS

<sup>5</sup>We include experiments on normal-form games, as well as all missing additional implementation details and network architecture diagrams in Appendix G.

<sup>6</sup>We refer the reader to Appendix G.2 for a definition.

CES, and GC CES experiments to the Appendix). We report the distribution of the exploitability on the test set for GAES and the baselines in Figure 10 (additional plots can be found in Appendix G.2). We measure performance w.r.t. the *normalized exploitability*, which is the exploitability of the method divided by the average exploitability over the action space. We observe that in all economies, GAES outputs an action profile that is on average better than at least 99% of the action profiles in terms of exploitability. In all four markets, GAES on average achieves lower exploitability than the baselines, in distribution, in every economy except Cobb-Douglas. This is not surprising since tâtonnement is guaranteed to converge in Cobb-Douglas economies (Bei et al., 2015). That said, tâtonnement does not outperform GAES on average, since tâtonnement’s convergence guarantees hold for different learning rates in each market.

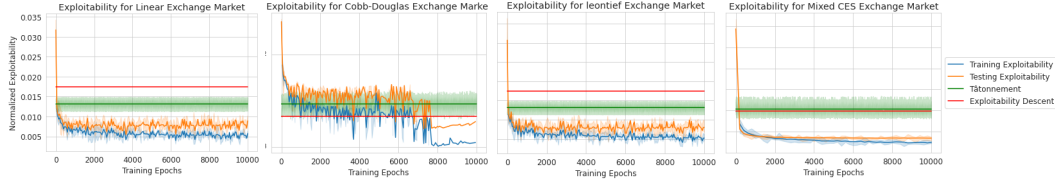


Figure 2: Training and Testing Exploitability of GAES in linear, Cobb-Douglas, Leontief, and mixed CES economies. GAES outperforms all baselines on average in all economies.

Second, we compare GAES with the performance of tâtonnement in a pathological and well-know Leontief exchange economy, the Scarf economy (Scarf, 1960) (Figure 3). Here, we soft start tâtonnement with the output of GAES with some added uniform noise. This additional noise ensures that the starting point of tâtonnement is distinct from the output of GAES). We see on Figure 3 that the prices generated by tâtonnement spiral out and settle into an orbit. This makes sense as, unlike pure Nash equilibria, GNE and CE are not locally stable (Flokas et al., 2020), meaning that soft-starting the algorithm with the output of GAES might be worse than using GAES alone. The success of GAES in Leontief exchange economies as well as the Scarf economy is notable, and suggests that GAES might be smoothing out the loss landscape since for these economies the exploitability is non-differentiable.

**Kyoto joint implementation mechanism.** We solve a pseudo-game model of the joint implementation mechanism proposed in the Kyoto protocols (Protocol, 1997). The *Kyoto Joint Implementation Mechanism* is a *cap-and-trade* mechanism that bounds each country that signed onto the protocol to emit anthropogenic gases below a particular emission cap. Countries bound by the mechanism can also invest in green projects in other countries, which in return increases their emission caps. Breton et al. (2006) introduce a model of the Kyoto Joint Implementation Mechanism, using this to predict the impact of the mechanism. The model that the authors propose is partially solvable analytically, that is, one can characterize equilibria qualitatively using comparative statics (Nachbar, 2002), but cannot obtain closed-form solutions for GNE. Moreover, there is no algorithm that is known to converge to the GNE of this pseudo-game, since it is not monotone.

Formally, the (*Kyoto*) *Joint Implementation Mechanism* (JI) consists of  $n \in \mathbb{N}_+$  countries. Each country  $i \in [n]$ , can make decisions that result in environmentally damaging anthropogenic emissions,  $e_i \in \mathbb{R}_+$ , and can make investments  $x_i \in \mathbb{R}^n$  to offset their emissions. The investments  $x_i \in \mathbb{R}^n$  that each country  $i \in [n]$  makes in another country  $j \in [n]$  offsets that country’s emissions by  $x_{ij}\gamma_j$ , where this is in proportion to an investment return rate  $\gamma_j > 0$  for country  $j$ , with  $\gamma \in \mathbb{R}_+^n$ . Each country  $i$  has a revenue  $r_i : \mathbb{R}_+^n \rightarrow \mathbb{R}$ , which is a function of its emissions  $e_i$ , a cost function  $c_i : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}$  that is a function of all investments, and a negative externality function,  $d_i : \mathbb{R}^n \rightarrow \mathbb{R}$ , that is a function of the net emissions of all countries.

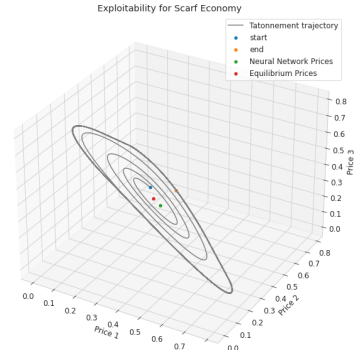


Figure 3: A phase portrait of equilibrium prices in the Scarf Economy. While the output of GAES is close to the equilibrium prices, the final prices outputted by tâtonnement prices are further than the starting prices.



Each country  $i$  aims to maximize their surplus, i.e., their revenue minus costs and negative externalities, constrained by keeping their net emissions under an emission cap,  $\eta_i \in \mathbb{R}_+$ . Additionally, we require the emission transfer balance to hold, i.e.,  $\forall i, e_i - \gamma_i \sum_{j \in [n]} x_{ji} \geq 0$ , that is no country can transfer more emission reduction than they have, giving for each country,  $i$ :

$$\max_{\substack{(e_i, \mathbf{x}_i) \in \mathbb{R}_+ \times \mathbb{R}_+^n : e_i - \sum_{j \in [n]} x_{ij} \gamma_j \leq \eta_i \\ \forall i \in [n], e_i - \gamma_i \sum_{j \in [n]} x_{ji} \geq 0}} r_i(e_i) - c_i(\mathbf{X}) - d_i\left(\left\{e_i - \sum_{j \in [n]} x_{ji} \gamma_j\right\}_i\right) \quad \text{The}$$

literature has traditionally assumed that  $r_i(e_i; \theta_i) = e_i(\theta_i - e_i/2)$ ,  $c_i(\mathbf{X}) = \frac{1}{2} \left( x_{ii}^2 + \sum_{j \neq i} [(x_{ij} + x_{jj})^2 - x_{jj}^2] \right)$ , and  $d_i(e - \sum_{i \in [n]} \mathbf{x}_i, \beta_i) = \beta_i \sum_{i \in [n]} (e_i - \sum_{j \in [n]} x_{ij})$ . Fixing these functional forms, we can sample  $(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\eta}) \in \mathbb{R}_+^m \times \mathbb{R}^m \times \mathbb{R}^{m \times m} \times \mathbb{R}^m$  and obtain a representation of the JI mechanism.

We run two different experiments to replicate and extend the analysis of Breton et al. (2006). We first replicate their qualitative analysis of equilibria (Figure 4). Breton et al. (2006) introduce a comparative static analysis, in which they fix all parameters of JI except for  $\boldsymbol{\theta}$ , and characterize the kinds of GNE as  $\boldsymbol{\theta}$  varies. In Figure 4, the six regions correspond to different kinds of GNE. For instance in Region 1, both countries emit strictly less than their emission cap (see Section 4 of Breton et al. (2006)). The parts of the plot that are not shaded correspond to pseudo-games for which GNE are not unique, and for which equilibria cannot be characterized analytically. We superpose on top of this plot a set of pseudo-games from an unseen test set, and color each pseudo-game by the color of the region whose condition they fulfill.

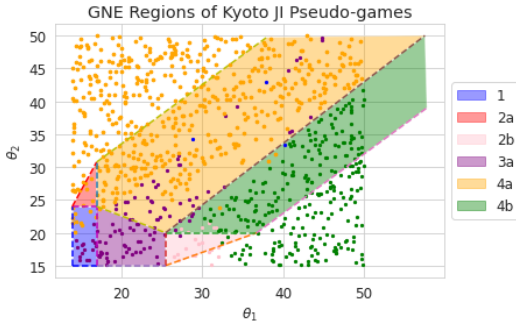


Figure 4: A taxonomy of different equilibrium types for various pseudo-games obtained by fixing all parameters, and varying  $\boldsymbol{\theta}$ . The  $x$  and  $y$  axes represent the revenue parameters  $\theta_1, \theta_2$  of the countries. The colored regions (obtained analytically) correspond to different qualitative types of GNE while the dots correspond to pseudo-games in the test set colored by the GNE types that were predicted for them by GAES.

as the exploitability divided by the average exploitability over the action space, which means that GAES has on average a lower exploitability than  $\approx 99.5\%$  of the feasible action profiles. This confirms our hypothesis that failure to predict Regions 1 and 2a in Figure 4 arises from the proximity between GNE of these types and GNE of types 3a or 4a respectively, since according to exploitability there is very little improvement left for GAES.

## 6 CONCLUSION

We introduced GAES, a GAN that learns mappings from pseudo-games to GNE. GAES outperforms existing methods to compute GNE or CE in exchange economies, and solves even pathological examples, i.e., Scarf economies. Our approach extends the use of exploitability-based learning methods from normal-form games to continuous games, exchange markets, and beyond. GAES adds to the growing list of differentiable economics methods that aim to provide practitioners with computational tools for the study of economic properties. GAES extends the range of models for which we have approximate and reliable solvers.

We observe that with the exception of Regions 1 and 2a, the structure of the GNE generated by GAES lines up well with the type of GNE. We believe that failure to predict Regions 1 and 2a is due to the closeness of the action profiles that fit either of these equilibrium types to other ones. For instance, GAES predicts a GNE of type 4a for pseudo-games in Region 2a but these GNE, although qualitatively different, are very close in action space: the only difference between the two regions is that for the type 2a GNE, one player emits strictly less than its cap and the other emits exactly its cap, while for the type 4a GNE, both players emit exactly their emission cap. A similar conclusion holds for GNE in Region 3a, as predicted in Region 1.

We also solve the JI pseudo-game for a distribution of JI pseudo-games (Figure 14, Appendix G), and these results confirm that the testing normalized exploitability is very low.

We note that *normalized exploitability* is given

## REFERENCES

- S. N. Afriat. Theory of maxima and the method of lagrange. *SIAM Journal on Applied Mathematics*, 20(3):343–357, 1971. ISSN 00361399. URL <http://www.jstor.org/stable/2099955>.
- Jonatha Anselmi, Danilo Ardagna, and Mauro Passacantando. Generalized nash equilibria for saas/paas clouds. *European Journal of Operational Research*, 236(1):326–339, 2014.
- Danilo Ardagna, Sara Casolari, and Barbara Panicucci. Flexible distributed capacity allocation and load redirect algorithms for cloud systems. In *2011 IEEE 4th International Conference on Cloud Computing*, pp. 163–170. IEEE, 2011a.
- Danilo Ardagna, Barbara Panicucci, and Mauro Passacantando. A game theoretic formulation of the service provisioning problem in cloud systems. In *Proceedings of the 20th International Conference on World Wide Web, WWW '11*, pp. 177–186, New York, NY, USA, 2011b. Association for Computing Machinery. ISBN 9781450306324. doi: 10.1145/1963405.1963433. URL <https://doi.org/10.1145/1963405.1963433>.
- Danilo Ardagna, Michele Ciavotta, and Mauro Passacantando. Generalized nash equilibria for the service provisioning problem in multi-cloud systems. *IEEE Transactions on Services Computing*, 10(3):381–395, 2017. doi: 10.1109/TSC.2015.2477836.
- Kenneth Arrow and Gerard Debreu. Existence of an equilibrium for a competitive economy. *Econometrica: Journal of the Econometric Society*, pp. 265–290, 1954.
- Qin Ba and Jong-Shi Pang. Exact penalization of generalized nash equilibrium problems. *Operations Research*, 2020.
- Raef Bassily, Mikhail Belkin, and Siyuan Ma. On exponential convergence of sgd in non-convex over-parametrized learning. *arXiv preprint arXiv:1811.02564*, 2018.
- Xiaohui Bei, Jugal Garg, and Martin Hoefer. Tatonnement for linear and gross substitutes markets. *CoRR abs/1507.04925*, 2015.
- Martin Bichler, Maximilian Fichtl, Stefan Heidekrüger, Nils Kohring, and Paul Sutterer. Learning equilibria in symmetric auction games using artificial neural networks. *Nature machine intelligence*, 3(8):687–695, 2021.
- Jason Bigler. Rolling out first price auctions to google ad manager partners. *Consulted at https://www.blog.google/products/admanager/rolling-out-first-price-auctions-google-ad-manager-partners*, 2019.
- Mathieu Blondel, Quentin Berthet, Marco Cuturi, Roy Frostig, Stephan Hoyer, Felipe Llinares-Lopez, Fabian Pedregosa, and Jean-Philippe Vert. Efficient and modular implicit differentiation. *arXiv preprint arXiv:2105.15183*, 2021.
- James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL <http://github.com/google/jax>.
- Simina Brânzei, Nikhil Devanur, and Yuval Rabani. Proportional dynamics in exchange economies. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pp. 180–201, 2021.
- Michele Breton, Georges Zaccour, and Mehdi Zahaf. A game-theoretic formulation of joint implementation of environmental projects. *European Journal of Operational Research*, 168(1): 221–239, 2006.
- Michael Bruckner and Tobias Scheffer. Nash equilibria of static prediction games. *Advances in neural information processing systems*, 22, 2009.
- Michael Bruckner, Christian Kanzow, and Tobias Scheffer. Static prediction games for adversarial learning problems. *J. Mach. Learn. Res.*, 13:2617–2654, 2012. URL <http://dl.acm.org/citation.cfm?id=2503326>.

- Eric Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6):1061–1103, 2011.
- Luis Felipe Bueno, Gabriel Haeser, and Frank Navarro Rojas. Optimality conditions and constraint qualifications for generalized nash equilibrium problems and their practical implications. *SIAM Journal on Optimization*, 29(1):31–54, 2019. doi: 10.1137/17M1162524. URL <https://doi.org/10.1137/17M1162524>.
- Brendan Burns, Brian Grant, David Oppenheimer, Eric Brewer, and John Wilkes. Borg, omega, and kubernetes. *Communications of the ACM*, 59(5):50–57, 2016.
- Li Chen, Shuhao Liu, Baochun Li, and Bo Li. Scheduling jobs across geo-distributed datacenters with max-min fairness. *IEEE Transactions on Network Science and Engineering*, 6(3):488–500, 2018.
- Mingli Chen, Andreas Joseph, Michael Kumhof, Xinlei Pan, Rui Shi, and Xuan Zhou. Deep reinforcement learning in a monetary model. *arXiv preprint arXiv:2104.09368*, 2021.
- Xi Chen and Xiaotie Deng. Settling the complexity of two-player nash equilibrium. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pp. 261–272. IEEE, 2006.
- Xi Chen and Shang-Hua Teng. Spending is not easier than trading: on the computational equivalence of fisher and arrow-debreu equilibria. In *International Symposium on Algorithms and Computation*, pp. 647–656. Springer, 2009.
- Xi Chen, Decheng Dai, Ye Du, and Shang-Hua Teng. Settling the complexity of arrow-debreu equilibria in markets with additively separable utilities. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pp. 273–282. IEEE, 2009.
- Yun Kuen Cheung, Richard Cole, and Nikhil Devanur. Tatonnement beyond gross substitutes? gradient descent to the rescue. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing, STOC '13*, pp. 191–200, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450320290. doi: 10.1145/2488608.2488633. URL <https://doi.org/10.1145/2488608.2488633>.
- Bruno Codenotti, Amin Saberi, Kasturi Varadarajan, and Yinyu Ye. Leontief economies encode nonzero sum two-player games. In *SODA*, volume 6, pp. 659–667, 2006.
- Vincent Conitzer, Christian Kroer, Debmalya Panigrahi, Okke Schrijvers, Nicolas E Stier-Moses, Eric Sodomka, and Christopher A Wilkens. Pacing equilibrium in first price auction markets. *Management Science*, 2022a.
- Vincent Conitzer, Christian Kroer, Eric Sodomka, and Nicolas E Stier-Moses. Multiplicative pacing equilibria in auction markets. *Operations Research*, 70(2):963–989, 2022b.
- Michael Curry, Tuomas Sandholm, and John Dickerson. Differentiable economics for randomized affine maximizer auctions. *arXiv preprint arXiv:2202.02872*, 2022a.
- Michael Curry, Alexander Trott, Soham Phade, Yu Bai, and Stephan Zheng. Finding general equilibria in many-agent economic simulations using deep reinforcement learning. *arXiv preprint arXiv:2201.01163*, 2022b.
- Michael J Curry, Uro Lyi, Tom Goldstein, and John P Dickerson. Learning revenue-maximizing auctions with differentiable matching. In *International Conference on Artificial Intelligence and Statistics*, pp. 6062–6073. PMLR, 2022c.
- Constantinos Daskalakis, Paul W Goldberg, and Christos H Papadimitriou. The complexity of computing a nash equilibrium. *SIAM Journal on Computing*, 39(1):195–259, 2009.
- Gerard Debreu. *General equilibrium theory*. Edward Elgar Publishing, 1996.
- Nikhil R Devanur, Jugal Garg, and László A Végh. A rational convex program for linear arrow-debreu markets. *ACM Transactions on Economics and Computation (TEAC)*, 5(1):1–13, 2016.

- Peter B Dixon and Brian R Parmenter. Computable general equilibrium modelling for policy analysis and forecasting. *Handbook of computational economics*, 1:3–85, 1996.
- Axel Dreves. Computing all solutions of linear generalized nash equilibrium problems. *Mathematical Methods of Operations Research*, 85(2):207–221, 2017.
- Axel Dreves, Anna Heusinger, Christian Kanzow, and Masao Fukushima. A globalized newton method for the computation of normalized nash equilibria. *J. of Global Optimization*, 56(2):327–340, jun 2013. ISSN 0925-5001. doi: 10.1007/s10898-011-9824-9. URL <https://doi.org/10.1007/s10898-011-9824-9>.
- Zhijian Duan, Dinghuai Zhang, Wenhan Huang, Yali Du, Jun Wang, Yaodong Yang, and Xiaotie Deng. Towards the pac learnability of nash equilibrium. *arXiv preprint arXiv:2108.07472*, 2021a.
- Zhijian Duan, Dinghuai Zhang, Wenhan Huang, Yali Du, Jun Wang, Yaodong Yang, and Xiaotie Deng. Towards the pac learnability of nash equilibrium, 2021b. URL <https://arxiv.org/abs/2108.07472>.
- Zhijian Duan, Jingwu Tang, Yutong Yin, Zhe Feng, Xiang Yan, Manzil Zaheer, and Xiaotie Deng. A context-integrated transformer-based neural network for auction design. In *International Conference on Machine Learning*, pp. 5609–5626. PMLR, 2022.
- Paul Dütting, Zhe Feng, Harikrishna Narasimhan, David Parkes, and Sai Srivatsa Ravindranath. Optimal auctions through deep learning. In *International Conference on Machine Learning*, pp. 1706–1715. PMLR, 2019.
- Francisco Facchinei and Christian Kanzow. Generalized nash equilibrium problems. *Annals of Operations Research*, 175(1):177–211, 2010a.
- Francisco Facchinei and Christian Kanzow. Penalty methods for the solution of generalized nash equilibrium problems. *SIAM Journal on Optimization*, 20(5):2228–2253, 2010b. doi: 10.1137/090749499. URL <https://doi.org/10.1137/090749499>.
- Francisco Facchinei and Lorenzo Lampariello. Partial penalization for the solution of generalized nash equilibrium problems. *Journal of Global Optimization*, 50(1):39–57, 2011.
- Francisco Facchinei, Andreas Fischer, and Veronica Piccialli. On generalized nash games and variational inequalities. *Operations Research Letters*, 35(2):159–164, 2007. ISSN 0167-6377. doi: <https://doi.org/10.1016/j.orl.2006.03.004>. URL <https://www.sciencedirect.com/science/article/pii/S0167637706000484>.
- Francisco Facchinei, Andreas Fischer, and Veronica Piccialli. Generalized nash equilibrium problems and newton methods. *Mathematical Programming*, 117(1):163–194, 2009.
- Andreas Fischer, Markus Herrich, Alexey F Izmailov, and Mikhail V Solodov. A globally convergent lp-newton method. *SIAM Journal on Optimization*, 26(4):2012–2033, 2016.
- Lampros Flokas, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Thanasis Lianas, Panayotis Mertikopoulos, and Georgios Piliouras. No-regret learning and mixed nash equilibria: They do not mix. *arXiv preprint arXiv:2010.09514*, 2020.
- Masao Fukushima. Restricted generalized nash equilibria and controlled penalty algorithm. *Computational Management Science*, 8(3):201–218, 2011.
- Jugal Garg, Ruta Mehta, Vijay V Vazirani, and Sadra Yazdanbod. Settling the complexity of leontief and plc exchange markets under exact and approximate equilibria. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 890–901, 2017.
- Ian Gemp, Thomas Anthony, Janos Kramar, Tom Eccles, Andrea Tacchetti, and Yoram Bachrach. Designing all-pay auctions using deep learning and multi-agent simulation. *Scientific Reports*, 12(1):16937, 2022.
- Ali Ghodsi, Matei Zaharia, Benjamin Hindman, Andy Konwinski, Scott Shenker, and Ion Stoica. Dominant resource fairness: Fair allocation of multiple resource types. In *8th USENIX Symposium on Networked Systems Design and Implementation (NSDI 11)*, 2011.

- Ali Ghodsi, Matei Zaharia, Scott Shenker, and Ion Stoica. Choosy: Max-min fair sharing for data-center jobs with constraints. In *Proceedings of the 8th ACM European Conference on Computer Systems*, pp. 365–378, 2013.
- Denizalp Goktas and Amy Greenwald. Convex-concave min-max stackelberg games. *Advances in Neural Information Processing Systems*, 34, 2021.
- Denizalp Goktas and Amy Greenwald. Exploitability minimization in games and beyond. In *Advances in Neural Information Processing Systems*, 2022.
- Noah Golowich, Sarath Pattathil, Constantinos Daskalakis, and Asuman E. Ozdaglar. Last iterate is slower than averaged iterate in smooth convex-concave saddle point problems. *CoRR*, abs/2002.00057, 2020. URL <https://arxiv.org/abs/2002.00057>.
- Wenshuo Guo, Kirthevasan Kandasamy, Joseph E Gonzalez, Michael I. Jordan, and Ion Stoica. Learning competitive equilibria in exchange economies with bandit feedback, 2021. URL <https://arxiv.org/abs/2106.06616>.
- Avital Gutman and Noam Nisan. Fair allocation without trade. *arXiv preprint arXiv:1204.4286*, 2012.
- Zhu Han, Dusit Niyato, Walid Saad, Tamer Basar, and Are Hjorungnes. *Game theory in wireless and communication networks: Theory, models, and applications*, volume 9780521196963. Cambridge University Press, United Kingdom, January 2011. ISBN 9780521196963. doi: 10.1017/CBO9780511895043. Publisher Copyright: © Cambridge University Press 2012.
- Charles R. Harris, K. Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernandez del Rio, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585:357–362, 2020. doi: 10.1038/s41586-020-2649-2.
- Stefan Heidekrüger, Paul Sutterer, and Martin Bichler. Computing approximate bayes-nash equilibria through neural self-play. In *Workshop on Information Technology and Systems (WITS19)*, 2019.
- Tom Hennigan, Trevor Cai, Tamara Norman, and Igor Babuschkin. Haiku: Sonnet for JAX, 2020. URL <http://github.com/deepmind/dm-haiku>.
- A. Heusinger and C. Kanzow. Relaxation methods for generalized nash equilibrium problems with inexact line search. *Journal of Optimization Theory and Applications*, 143(1):159–183, 2009. URL [https://EconPapers.repec.org/RePEc:spr:joptap:v:143:y:2009:i:1:d:10.1007\\_s10957-009-9553-0](https://EconPapers.repec.org/RePEc:spr:joptap:v:143:y:2009:i:1:d:10.1007_s10957-009-9553-0).
- Edward Hill, Marco Bardoscia, and Arthur Turrell. Solving heterogeneous general equilibrium economic models with deep reinforcement learning. *arXiv preprint arXiv:2103.16977*, 2021.
- Benjamin Hindman, Andy Konwinski, Matei Zaharia, Ali Ghodsi, Anthony D Joseph, Randy Katz, Scott Shenker, and Ion Stoica. Mesos: A platform for {Fine-Grained} resource sharing in the data center. In *8th USENIX Symposium on Networked Systems Design and Implementation (NSDI 11)*, 2011.
- Benjamin F. Hobbs and J. S. Pang. Nash-cournot equilibria in electric power markets with piecewise linear demand functions and joint constraints. *Operations Research*, 55(1):113–127, 2007. doi: 10.1287/opre.1060.0342. URL <https://doi.org/10.1287/opre.1060.0342>.
- J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science and Engineering*, 9(3): 90–95, 2007. doi: 10.1109/MCSE.2007.55.
- Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pp. 448–456. pmlr, 2015.



- Michael Isard, Vijayan Prabhakaran, Jon Currey, Udi Wieder, Kunal Talwar, and Andrew Goldberg. Quincy: fair scheduling for distributed computing clusters. In *Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles*, pp. 261–276, 2009.
- Alexey F Izmailov and Mikhail V Solodov. On error bounds and newton-type methods for generalized nash equilibrium problems. *Computational Optimization and Applications*, 59(1):201–218, 2014.
- Xuegang (Jeff) Ban, Maged Dessouky, Jong-Shi Pang, and Rong Fan. A general equilibrium model for transportation systems with e-hailing services and flow congestion. *Transportation Research Part B: Methodological*, 129:273–304, 2019. ISSN 0191-2615. doi: <https://doi.org/10.1016/j.trb.2019.08.012>. URL <https://www.sciencedirect.com/science/article/pii/S0191261518309044>.
- Wei Jing-Yuan and Yves Smeers. Spatial oligopolistic electricity models with cournot generators and regulated transmission prices. *Operations Research*, 47(1):102–112, 1999. doi: 10.1287/opre.47.1.102. URL <https://pubsonline.informs.org/doi/abs/10.1287/opre.47.1.102>.
- Roger Jones, K. Hennessy, C.M. Page, Albert Pittock, R. Suppiah, Kevin Walsh, and Penny Whetton. *Analysis of the Kyoto Protocol on Pacific Island Countries*. 01 2000.
- Michael I. Jordan, Tianyi Lin, and Manolis Zampetakis. First-order algorithms for nonlinear generalized nash equilibrium problems, 2022. URL <https://arxiv.org/abs/2204.03132>.
- Christian Kanzow. On the multiplier-penalty-approach for quasi-variational inequalities. *Mathematical Programming*, 160(1):33–63, 2016.
- Christian Kanzow and Daniel Steck. Augmented lagrangian and exact penalty methods for quasi-variational inequalities. *Computational Optimization and Applications*, 69(3):801–824, 2018.
- Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-lojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 795–811. Springer, 2016.
- Jacek B. Krawczyk. Coupled constraint nash equilibria in environmental games. *Resource and Energy Economics*, 27(2):157–181, 2005. ISSN 0928-7655. doi: <https://doi.org/10.1016/j.reseneeco.2004.08.001>. URL <https://www.sciencedirect.com/science/article/pii/S0928765504000661>.
- Jacek B. Krawczyk and Stanislav Uryasev. Relaxation algorithms to find nash equilibria with economic applications. *Environmental Modeling and Assessment*, 5(1):63–73, 2000. doi: 10.1023/A:1019097208499. This revised version was published online in July 2006 with corrections to the Cover Date.
- Kevin Lai, Lars Rasmusson, Eytan Adar, Li Zhang, and Bernardo A Huberman. Tycoon: An implementation of a distributed, market-based resource allocation system. *Multiagent and Grid Systems*, 1(3):169–182, 2005.
- Marc Lanctot, Vinicius Zambaldi, Audrunas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Pérolat, David Silver, and Thore Graepel. A unified game-theoretic approach to multiagent reinforcement learning. *Advances in neural information processing systems*, 30, 2017.
- Tianyi Lin, Chi Jin, and Michael Jordan. On gradient descent ascent for nonconvex-concave minimax problems. In *International Conference on Machine Learning*, pp. 6083–6093. PMLR, 2020.
- Zhihan Liu, Miao Lu, Zhaoran Wang, Michael Jordan, and Zhuoran Yang. Welfare maximization in competitive equilibrium: Reinforcement learning for markov exchange economy. In *International Conference on Machine Learning*, pp. 13870–13911. PMLR, 2022.
- Stanislaw Lojasiewicz. A topological property of real analytic subsets. *Coll. du CNRS, Les équations aux dérivées partielles*, 117(87-89):2, 1963.

- Luke Marris, Ian Gemp, Thomas Anthony, Andrea Tacchetti, Siqu Liu, and Karl Tuyls. Turbocharging solution concepts: Solving nes, ces and cces with neural equilibrium solvers, 2022. URL <https://arxiv.org/abs/2210.09257>.
- Koichi Nabetani, Paul Tseng, and Masao Fukushima. Parametrized variational inequality approaches to generalized nash equilibrium problems with shared constraints. *Comput. Optim. Appl.*, 48(3):423–452, apr 2011. ISSN 0926-6003. doi: 10.1007/s10589-009-9256-3. URL <https://doi.org/10.1007/s10589-009-9256-3>.
- John H Nachbar. General equilibrium comparative statics. *Econometrica*, 70(5):2065–2074, 2002.
- John F. Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950. doi: 10.1073/pnas.36.1.48. URL <https://www.pnas.org/doi/abs/10.1073/pnas.36.1.48>.
- Hukukane Nikaido and Kazuo Isoda. Note on non-cooperative convex games. *Pacific Journal of Mathematics*, 5(S1):807–815, 1955.
- Maher Nouiehed, Maziar Sanjabi, Tianjian Huang, Jason D Lee, and Meisam Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. *arXiv preprint arXiv:1902.08297*, 2019.
- Eugene Nudelman, Jennifer Wortman, Yoav Shoham, and Kevin Leyton-Brown. Run the gamut: A comprehensive approach to evaluating game-theoretic algorithms. In *AAMAS*, volume 4, pp. 880–887, 2004.
- Jong-Shi Pang and Masao Fukushima. Quasi-variational inequalities, generalized nash equilibria, and multi-leader-follower games. *Computational Management Science*, 2(1):21–56, 2005.
- Jong-Shi Pang, Gesualdo Scutari, Francisco Facchinei, and Chaoxiong Wang. Distributed power allocation with rate constraints in gaussian parallel interference channels. *IEEE Transactions on Information Theory*, 54(8):3471–3489, 2008.
- David C Parkes, Ariel D Procaccia, and Nisarg Shah. Beyond dominant resource fairness: Extensions, limitations, and indivisibilities. *ACM Transactions on Economics and Computation (TEAC)*, 3(1):1–22, 2015.
- Mark D Partridge and Dan S Rickman. Computable general equilibrium (cge) modelling for regional economic development analysis. *Regional studies*, 44(10):1311–1328, 2010.
- Kyoto Protocol. Kyoto protocol. *UNFCCC Website. Available online: http://unfccc.int/kyoto\_protocol/items/2830.php (accessed on 1 January 2011)*, 1997.
- Jad Rahme, Samy Jelassi, and S Matthew Weinberg. Auction learning as a two-player game. *arXiv preprint arXiv:2006.05684*, 2020.
- Sai Srivatsa Ravindranath, Zhe Feng, Shira Li, Jonathan Ma, Scott D Kominers, and David C Parkes. Deep learning for two-sided matching. *arXiv preprint arXiv:2107.03427*, 2021.
- J. B. Rosen. Existence and uniqueness of equilibrium points for concave n-person games. *Econometrica*, 33(3):520–534, 1965. ISSN 00129682, 14680262. URL <http://www.jstor.org/stable/1911749>.
- Maziar Sanjabi, Jimmy Ba, Meisam Razaviyayn, and Jason D. Lee. On the convergence and robustness of training gans with regularized optimal transport, 2018.
- Herbert Scarf. Some examples of global instability of the competitive equilibrium. *International Economic Review*, 1(3):157–172, 1960. ISSN 00206598, 14682354. URL <http://www.jstor.org/stable/2556215>.
- Dane A. Schiro, Jong-Shi Pang, and Uday V. Shanbhag. On the solution of affine generalized nash equilibrium problems with shared constraints by lemke’s method. *Mathematical Programming*, 142(1-2):1–46, 2013. doi: 10.1007/s10107-012-0558-3. This work was based on research partially supported by the National Science Foundation under grant CMMI-0969600 and the Department of Energy under grant DOE DE-SC0003879.

- Oliver Stein and Nathan Sudermann-Merx. The noncooperative transportation problem and linear generalized Nash games. *European Journal of Operational Research*, 266(2):543–553, 2018. doi: 10.1016/j.ejor.2017.10.00. URL <https://ideas.repec.org/a/eee/ejores/v266y2018i2p543-553.html>.
- Andrea Tacchetti, DJ Strouse, Marta Garnelo, Thore Graepel, and Yoram Bachrach. A neural architecture for designing truthful and efficient auctions. *arXiv preprint arXiv:1907.05181*, 2019.
- Matthias Troffaes. pycddlib a python wrapper for komei fukuda’s cddlib, 2020. URL <https://pycddlib.readthedocs.io/en/latest/license.html>.
- S. Uryas’ev and R.Y. Rubinstein. On relaxation algorithms in computation of noncooperative equilibria, 1994.
- Guido Van Rossum and Fred L Drake Jr. *Python tutorial*. Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands, 1995.
- Hal R Varian. Equity, envy, and efficiency. 1973.
- Vinod Kumar Vavilapalli, Arun C Murthy, Chris Douglas, Sharad Agarwal, Mahadev Konar, Robert Evans, Thomas Graves, Jason Lowe, Hitesh Shah, Siddharth Seth, et al. Apache hadoop yarn: Yet another resource negotiator. In *Proceedings of the 4th annual Symposium on Cloud Computing*, pp. 1–16, 2013.
- Vijay V Vazirani and Mihalis Yannakakis. Market equilibrium under separable, piecewise-linear, concave utilities. *Journal of the ACM (JACM)*, 58(3):1–25, 2011.
- Anna Von Heusinger and Christian Kanzow. Optimization reformulations of the generalized nash equilibrium problem using nikaido-isoda-type functions. *Computational Optimization and Applications*, 43(3):353–377, 2009.
- Anna von Heusinger, Christian Kanzow, and Masao Fukushima. Newton’s method for computing a normalized equilibrium in the generalized nash game through fixed point formulation. *Mathematical Programming*, 132(1):99–123, 2012.
- Leon Walras. *Elements de l’economie politique pure, ou, Theorie de la richesse sociale*. F. Rouge, 1896.
- Michael L. Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021. doi: 10.21105/joss.03021. URL <https://doi.org/10.21105/joss.03021>.
- Seyed Majid Zahedi, Qiuyun Llull, and Benjamin C Lee. Amdahl’s law in the datacenter era: A market for fair processor allocation. In *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pp. 1–14. IEEE, 2018.