Adaptive Mechanism Design using Multi-Agent Revealed Preferences

Luke Snow, Vikram Krishnamurthy

***Abstract*— This paper constructs an algorithmic framework for adaptively achieving the mechanism design objective, finding a mechanism inducing socially optimal Nash equilibria, without knowledge of the utility functions of the agents. We consider a probing scheme where the designer can iteratively enact mechanisms and observe Nash equilibria responses. We first derive necessary and sufficient conditions, taking the form of linear program feasibility, for the existence of utility functions under which the empirical Nash equilibria responses are socially optimal. Then, we utilize this to construct a loss function with respect to the mechanism, and show that its global minimization occurs at mechanisms under which Nash equilibria system responses are also socially optimal. We develop a simulated annealing-based gradient algorithm, and prove that it converges in probability to this set of global minima, thus achieving adaptive mechanism design.**

I. Introduction

Consider the standard non-cooperative game-theoretic setting. Several agents act in their own self-interest, aiming to maximize their individual utility functions which are dependent on the actions of the entire set of agents. The classical concept of Nash equilibria represents the stable joint-action states of the entire group, representing those states in which no agent has any incentive to unilaterally deviate given the actions of all other agents. In many instances, this non-cooperative interaction degrades the overall utility achieved by the group; that is, there may be other joint-actions not arising from the Nash stability criteria under which the entire group of agents does better. We term those joint-actions which maximize the sum of agent utilities as 'socially optimal'. The goal of *mechanism design* is to fashion the game structure, specifically the mapping from joint-action space to agent utilities ("mechanism"), such that *non-cooperative behavior gives rise to socially optimal solutions.*

Literature and Main Result. Mechanism design arises in electronic market design [1], economic policy delegation [2] and dynamic spectrum sharing [3]. Standard solutions, whether analytical [4] or automated [5], pre-suppose knowledge of the agent utility functions in order to devise mechanisms inducing socially optimal Nash equilibria. This paper contributes to the field of algorithmic mechanism design by constructing a framework for attaining socially optimal Nash equilibria *without knowledge of the utility functions of the agents.* This is relevant to problems where the designer does not have a-priori knowledge of each agent's preferences and goals.

We consider a designer who does not know the agent utilities but can repeatedly interact with the system and adapt its mechanism based on the system's Nash equilibrium behavior. We employ microeconomic revealed preference theory to develop an algorithmic formulation which converges to a mechanism inducing socially optimal Nash equilibria behavior. Specifically, our main contributions are:

- We generalize the influential result of Forges and Minelli [6] in revealed preferences to the multi-agent regime. Revealed preferences are a branch of microeconomics that give necessary and sufficient conditions under which it is possible to identify utility maximization behavior from empirical data of a decision maker. Our generalization formulates necessary and sufficient conditions, in the form of linear program feasibility, for the existence of utility functions under which empirically observed Nash equilibrium responses are *socially optimal*.
- We exploit this linear program feasibility to construct a loss function which captures the distance between Nash equilibria to social optima. We show that the global minimizers of this loss function coincide with the socially optimal Nash equilibria. We then prove that this loss function can be *globally minimized* by iteratively observing Nash equilibrium responses of the system and updating the mechanism according to a simulated annealing-based gradient algorithm.

Thus, we provide a novel algorithmic framework for attaining mechanism design in an adaptive fashion without explicit knowledge of the utility functions of the agents.

Section II discusses background on mechanism design, revealed preference theory, and presents our first main result (Theorem 1). Section III exploits Theorem 1 to derive an algorithm for achieving adaptive mechanism design. It contains our second main result, Theorem 2, which proves the algorithm's global convergence in probability. Section IV presents an illustrative numerical example in a non-cooperative game setting.

This research was supported in part by the National Science Foundation grants CCF-2112457 and CCF-2312198, Army Research Office grant W911NF-19-1-0365, and Air Force Office of Scientific Research grant FA9550-22-1-0016

Luke Snow `las474@cornell.edu`, and Vikram Krishnamurthy `vikramk@cornell.edu` are with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853, USA

II. Preliminaries

In this section we first introduce the non-cooperative game-theoretic setting and the mechanism design problem. We then introduce microeconomic revealed preferences theory, and prove a key result which will allow us to achieve adaptive mechanism design.

A. Mechanism Design

For the mechanism design framework we suppose a parametrized finite-player static non-cooperative game, structured by the tuple $G = (M, \{f^i\}_{i\in[M]}, \mathcal{A}, \Theta, O)$ where

- M is the number of participating agents.
- $\mathcal{A}$ is the joint action space. Specifically, each agent i takes an action $a_i \in \mathcal{A}_i \subseteq \mathbb{R}^k$, where $\mathcal{A}_i$ is the space of actions available to agent i. Letting $\mathbf{a} = [a_1, \dots, a_M]'$ denote the *joint-action* (sometimes called joint-strategy) taken by all agents, $\mathcal{A} \subseteq \mathbb{R}^{kM}$ is then the space of joint actions, given by the cartesian product $\mathcal{A} = \otimes_i \mathcal{A}_i$.
- Θ is the mechanism parameter space, and O is the outcome space. Specifically, a *mechanism* $o_\theta : \mathcal{A} \to O$, parametrized by $\theta \in \Theta$, is a mapping from joint action $\mathbf{a}$ to *outcome* $o_\theta(\mathbf{a}) \in O$.
- $f^i : O \to \mathbb{R}$ is the utility function of agent i, assigning to each outcome a real-valued utility.

The real-valued utility gained for agent i, given mechanism o_θ (with parameter θ) and joint action $\mathbf{a}$, is $f^i(o_\theta(\mathbf{a}))$. For notational simplicity we denote $f^i_\theta(\mathbf{a}) = f^i(o_\theta(\mathbf{a}))$, so that the emphasis is on the mapping from joint action $\mathbf{a}$ to utility f^i, with the mechanism o_θ acting as an intermediary such that θ parametrizes this mapping.

$$\mathbf{a} \in \mathcal{A} \xrightarrow{\text{mechanism } \theta} o_\theta(\mathbf{a}) \in O \xrightarrow{\text{utility}} \{f^i_\theta(\mathbf{a})\}_{i=1}^M$$

Fig. 1: Mechanism flowchart. The mechanism modulates the mapping from joint-action to outcome space. The agent utilities are real-valued functions of outcomes. Thus, the mechanism equivalently modulates the utility functions' dependence on joint-actions.

Figure 1 illustrates the information flow in this setup. The mechanism is defined as a parameter which modulates the mapping from joint-action space $\mathcal{A}$ to outcome space O; it thus modulates each agent i's utility gained from joint-actions $\mathbf{a} \in \mathcal{A}$. [1]

The mechanism design problem begins by assuming that each agent i acts in its own self-interest, aiming to maximize its utility function f^i. The standard solution concept for this non-cooperative interaction is the Nash equilibrium:

[1]It is defined in this way, rather than directly parametrizing the utility functions, because oftentimes in practice the mapping from joint-actions to outcomes can be completely specified, while the influence on utility functions may be unknown.

Definition 1 (Nash Equilibrium): Joint strategy $\mathbf{a}$ is a Nash equilibrium of the game G under mechanism o_θ if

$$a_i \in \arg\max_{a\in\mathcal{A}_i} f^i_\theta(a, a_{-i}) \ \forall i \in [M] \tag{1}$$

where $a_{-i} = \mathbf{a}\backslash a_i$ is the set of actions without a_i, so that $(a_i, a_{-i}) = \mathbf{a}$.

Observe that the Nash equilibrium solutions (1) are dependent on the mechanism parameter θ. The *mechanism design* problem is that of designing the mechanism $o_\theta(\cdot)$, through parameter θ, such that a joint action $\mathbf{a}$ at Nash equilibrium also satisfies certain *global welfare conditions*. A standard global condition, known as the utilitarian condition [4], is to maximize the 'social welfare' which is simply the sum of agent utility functions

Definition 2 (Social Optimality): A joint-strategy $\mathbf{a}$ is socially optimal for the game G and mechanism o_θ if it maximizes the sum of agent utilities:

$$\mathbf{a} \in \arg\max_{a\in\mathcal{A}} \sum_{i=1}^M f^i_\theta(a) \tag{2}$$

Then the mechanism design problem is to find a mechanism o_θ, through parameter θ, such that the Nash equilibrium solution $\mathbf{a}$ arising from the conditions (1) simultaneously maximizes the social welfare, i.e.,

Definition 3 (Mechanism Design): The mechanism design problem is to find $\theta \in \Theta$ such that

$$\begin{aligned} &\mathbf{a} \text{ satisfies } a_i \in \arg\max_{a\in\mathcal{A}_i} f^i_\theta(a, a_{-i}) \ \forall i \in [M] \\ &\Rightarrow \mathbf{a} \in \arg\max_{a\in\mathcal{A}} \sum_{i=1}^M f^i_\theta(a) \end{aligned} \tag{3}$$

Notice that social optimality (2) is a special case of Pareto-optimality. In fact, the results presented in this work can easily be extended to the case where general Pareto-optimal solutions are considered. We exclude presentation and discussion of this generalization for brevity.

The mechanism design problem has been well-studied [4], [7], [5] in the case where the agent utility functions $\{f^i_\theta(\cdot)\}_{i=1}^M$ are known or can be designed. We consider the case when the *utility functions are unknown* to the designer. In many realistic scenarios, the designer may need to design the system without explicit knowledge of the preferences and goals of its constituent members. We achieve this by utilizing the methodology of revealed preferences to formulate a simulated annealing-based gradient algorithm which converges to a mechanism parameter θ inducing the relation (3).

B. Multi-Agent Revealed Preferences

The microeconomic field of revealed preferences dates back to the seminal work [8], and establishes conditions under which it is possible to detect utility maximization behavior from empirical consumer data. In particular, Afriat [9] initialized the study of nonparametric utility estimation from microeonomic consumer budget-expenditure data. Specifically, [9] provides necessary and sufficient conditions under which budget-expenditure data is consistent with linearly-constrained

utility maximization. [6] extended this work to incorporate non-linear budget constraints, and in this work we generalize [6] to the multi-agent regime. Specifically, we provide necessary and sufficient conditions for the existence of utility functions under which nonlinearly-constrained *joint*-actions are *socially optimal* (2).

The setting is as follows. We consider a multi-agent setting consisting of M agents. The designer observes the multi-agent system over T time-steps. At each time-step t, agent i's action $a_t^i \in \mathbb{R}^k$ is constrained by

$$g_t^i(a_t^i) \leq 0$$

where $g_t^i : \mathbb{R}^k \to \mathbb{R}$ is a continuous and increasing constraint function. The system then produces joint action $\mathbf{a}_t = [a_t^1, \ldots, a_t^M] \in \mathbb{R}^{kM}$, and the designer observes the dataset $\{\{g_t^i(\cdot)\}_{i=1}^M, \mathbf{a}_t, t \in [T]\}$, where $[x]$ denotes the set $\{1, 2, \ldots, x\}$.

Notice the black-box nature of these observations: the designer does not have any a-priori knowledge of how the actions $\mathbf{a}_t$ are produced. Still, the designer aims to determine if the group responses $\mathbf{a}_t$ are consistent with social-optimality. Specifically, he aims to determine if *there exist* utility functions $\{f^i(\cdot) : \mathbb{R}^{kM} \to \mathbb{R}\}_{i=1}^M$ such that for all $t \in [T]$,

$$\mathbf{a}_t \in \arg\max_a \sum_{i=1}^M f^i(a) \ s.t. \ g_t^i(a^i) \leq 0 \ \forall i \in [M] \qquad (4)$$

where $a = [a^1, \ldots, a^M]$ denotes the joint-action optimization variable. Our first main result generalizes the work of [6], providing necessary and sufficient conditions for this existence in the form of linear program feasibility.

Theorem 1: Consider a dataset of observed system constraints and responses $\mathcal{D} = \{g_t^i(\cdot), \mathbf{a}_t, t \in [T]\}$, with $g_t^i(a_t^i) = 0 \ \forall t \in [T], i \in [M]$[2]. The following are equivalent:

- There exists a set of concave, locally non-satiated utility functions $\{f^i : \mathbb{R}^{kM} \to \mathbb{R}\}_{i=1}^M$ such that the actions $\{\mathbf{a}_t, t \in [T]\}$ are socially optimal, i.e, (4) holds for all $t \in [T]$.
- For each $i \in M$, with $\mathbf{a}_t = [a_t^1, \ldots, a_t^M]$, there exist constants $u_t^i \in \mathbb{R}, \lambda_t^i > 0$ such that the following set of inequalities have a feasible solution

$$u_s^i - u_t^i - \lambda_t^i g_t^i(a_s^i) \leq 0 \quad \forall t, s \in [T] \qquad (5)$$

Proof: See Appendix VI-A ∎

In the following section we provide the adaptive mechanism design setting, which merges the two frameworks discusses in this section and employs Theorem 1 to develop an algorithmic solution. The condition (5) will serve as the key tool in formulating our approach.

[2]This condition is known as 'satiation' of constraint functions, and is widely assumed in the revealed preference literature [6]. Indeed, we will eventually assume concavity of utility functions f^i, which when considered with increasing constraint functions naturally necessitates satiation in utility-maximizing systems

III. Adaptive Mechanism Design

In this section we first outline the interaction dynamics between the designer and the multi-agent system, combining the frameworks of sections II-A and II-B. We utilize Theorem 1 to construct a loss function with respect to mechanism parameter θ, which has global minima corresponding to those mechanisms inducing socially optimal Nash equilibria. In section III-B we provide a simultaneous perturbation stochastic approximation (SPSA) algorithm operating on this loss function, and in section III-C we prove the global convergence of this algorithm under several basic assumptions.

A. Algorithm Derivation

Here we describe the high-level strategy and motivation for our adaptive mechanism design algorithm.

Interaction Procedure: First we outline the interaction procedure between the designer and the system constituents. As in the revealed preference framework, we treat the multi-agent system as a black box. The designer can provide a mechanism o_θ and constraints $\{g_t^i(\cdot)\}_{i=1}^M$. It subsequently observes the *Nash equilibrium* response $\mathbf{a}_t(\theta)$ (1) of the group in game G with mechanism parameter θ and with $\mathcal{A}_i = \{a \in \mathbb{R}^k : g_t^i(a) \leq 0\}$. Here t denotes a time-index as in the setting of Theorem 1, and we denote $\mathbf{a}_t(\theta)$ a function of θ to emphasize the dependence of Nash equilibria on the mechanism o_θ. The assumption that the response $\mathbf{a}_t(\theta)$ is a Nash equilibrium is standard in the mechanism design literature, and is readily achieved from non-cooperative best-response interactions among the agents [10]. Figure 2 displays this interaction procedure.

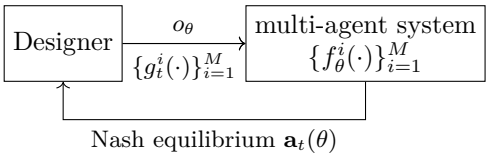

Fig. 2: Interaction procedure. At each time t, the designer can provide mechanism o_θ and constraint functions $\{g_t^i(\cdot)\}_{i=1}^M$, and observes Nash equilibrium response $\mathbf{a}_t$.

Suppose the designer now provides T sets of constraint functions $\{g_t^i(\cdot), t \in [T]\}_{i=1}^M$ and receives T corresponding Nash equilibrium responses $\mathbf{a}_t(\theta)$ under mechanism parameter θ. We denote this dataset as

$$\mathcal{D}_\theta = \{\{g_t^i(\cdot)\}_{i=1}^M, \mathbf{a}_t(\theta), t \in [T]\}$$

We translate the mechanism design goal to this empirical setting: to find a mechanism parameter θ such that empirical Nash equilibrium responses $\mathbf{a}_t(\hat{\theta})$ are *consistent* with social optimality, i.e., (1) of Theorem 1 holds.

Recall that the designer can directly test whether the dataset $\mathcal{D}_\theta$ is consistent with social optimality by testing the feasibility of linear program (5). That is, if (5) is feasible, then the goal is achieved. Our adaptive mechanism design solution is then to iteratively modify

the parameter θ until (5) is feasible for dataset $\mathcal{D}_\theta$. We will next formulate a loss function $L(\theta)$ which is minimized at some $\hat{\theta}$ which induces feasibility of (5).

Formulating the Loss Function: Suppose the LP (5) does not have a feasible solution for the dataset $\mathcal{D}_\theta$. How can we quantify the proximity of θ to a mechanism $\hat{\theta}$ which induces feasibility of (5)? First notice that we can augment the linear program (5) as:

$$\begin{aligned} L(\theta) = \arg\min_{r\in\mathbb{R}} : \exists\{u_t^i, \lambda_t^i, t \in [T], i \in [M]\} : \\ u_s^i - u_t^i - \lambda_t^i g_t^i(\mathbf{a}_s(\theta)) \leq r \quad \forall t, s \in [T], i \in [M] \end{aligned} \tag{6}$$

i.e., $L(\theta)$ is the minimum value of r such that the specified linear program has a feasible solution for the dataset of Nash equilibrium responses $\{\mathbf{a}_s(\theta), s \in [T]\}$. Thus, the optimization variable r can be treated as a metric capturing the proximity of the dataset $\mathcal{D}_\theta$ to being feasible in (5): if $r \leq 0$, then (5) is feasible, else lower r indicates $\mathcal{D}_\theta$ being "closer" to feasible.

Thus, we can adopt the augmented linear program solution $L(\theta)$ as an *objective function* taking parameter θ and outputting a value r representing the proximity of the empirical dataset $\mathcal{D}_\theta$ to feasibility (social optimality). The adaptive mechanism design problem can then be restated as

$$\text{Find } \hat{\theta} \in \Theta \; s.t. \; \hat{\theta} \in \arg\min_\theta L(\theta) \tag{7}$$

We next introduce a simultaneous perturbation stochastic approximation (SPSA) algorithm, and then prove that it converges in probability to the global minima of $L(\theta)$ under basic assumptions that make the problem well-posed. The proposed SPSA algorithm iteratively adjusts the mechanism parameter θ by observing batch datasets $\mathcal{D}_\theta$, evaluating $L(\theta)$, and moving θ in the direction of decreasing $L(\theta)$ by forming a stochastic gradient approximation, until a minima $\hat{\theta}$ making $L(\hat{\theta}) \leq 0$ is obtained. Thus, the SPSA algorithm achieves adaptive mechanism design by adaptively finding a mechanism that induces social optimality for Nash equilibrium responses without knowledge of agent utility functions.

B. Simultaneous Perturbation Stochastic Approximation

Simultaneous perturbation stochastic approximation (SPSA) algorithms are a well-studied class of optimization algorithms that operate without direct access to objective function gradient evaluations.

We use the SPSA algorithm with injected noise, as presented in [11] to globally optimize deterministic nonconvex functions. With objective function $L : \mathbb{R}^p \to \mathbb{R}$, this algorithm iterates via the recursion

$$\theta_{k+1} = \theta_k - a_k G_k(\theta_k) + q_k w_k \tag{8}$$

where $w_k \overset{i.i.d.}{\sim} \mathcal{N}(0, I)$ is purposefully injected Gaussian noise, $a_k = a/k, q_k^2 = q/k \log\log(k), a > 0, q > 0$, and G_k is a term approximating the function gradient,

$$G_k = \frac{L(\theta_k + c_k\Delta_k) - L(\theta_k - c_k\Delta_k)}{2c_k\Delta_k}$$

Here c_k is a scalar, and $\Delta_k = [\Delta_{k1}, \ldots, \Delta_{kp}]' \in \mathbb{R}^p$ is a p-dimensional Rademacher random variable, i.e., for each $i \in [p]$, Δ_{ki} is an independent Bernoulli(1/2) random variable.

Algorithm 1 Adaptive Mechanism Design

Initialize $\theta_0 \in \Theta \subset \mathbb{R}^p, T \in \mathbb{N}$, $a, c, q > 0, \eta \in [\frac{1}{6}, \frac{1}{2}]$
fin = 0; n=0;
for t=1:T **do**
  generate $\gamma_t \in \Gamma$;
end for
while fin==0 **do**
  n = n+1;
  $\Delta_n \sim$ p-dim Rademacher, $w_n \sim \mathcal{N}(0, I)$
  $c_n = c/n^\eta$; $a_n = a/n$; $q_n = \sqrt{q/(n\log(\log(n)))}$;
  $\theta_{n,1} = \theta_n + c_n\Delta_n$;
  $\theta_{n,2} = \theta_n - c_n\Delta_n$;
  for a=1:2 **do**
    for t=1:T **do**
      enact game G with $\theta_{n,a}$, $\{g_t^i(\cdot, \gamma_t)\}_{i\in[M]}$;
      observe Nash equilibrium $\mathbf{a}_t(\theta_{n,a})$;
    end for
    solve: minimize r s.t. $\exists\{u_t^i, \lambda_t^i\}$:
      $u_s^i - u_t^i - \lambda_t^i g_t^i(\mathbf{a}_s, \gamma_t) \leq r \;\; \forall t, s, i$;
    set $r_a = r$;
  end for
  if $\min(r_1, r_2) \leq 0$ **then**
    fin = 1; $i^* = \arg\min_i r_i$;
  else
    set $g_n = [(g_n)_1, \ldots, (g_n)_p]$ with
      $(g_n)_i = (r_1 - r_2)/(2c_n(\Delta_n)_i)$;
    $\theta_{n+1} = \text{Proj}_\Theta(\theta_n - a_n g_n + q_n w_n)$;
  end if
end while
Return $\hat{\theta} = \theta_{n,i^*}$

Observe that this algorithm only evaluates the objective function itself and does not rely on access to its gradients. This is appealing for use with our objective function (6) since it is not immediately obvious how one could access or compute the gradient of (6).

Also notice that the algorithm (8) injects Gaussian noise and thus imitates a simulated annealing [12] or stochastic gradient Langevin dynamics [13] procedure. It turns out that this allows the algorithm to converge to *global* minima of potentially nonconvex functions by escaping local minima [11]. Indeed, we will provide a result guaranteeing the convergence of our algorithm to the global minima of $L(\theta)$.

Implementation: The full implementation of the SPSA algorithm within our adaptive mechanism design framework can be seen in Algorithm 1. In this implementation, the objective function $L(\theta)$ is evaluated by obtaining the empirical dataset $\mathcal{D}_\theta$ and solving the linear program (6). Recall the empirical dataset $\mathcal{D}_\theta$ is obtained by "probing" the multi-agent system with constraint functions

$\{g_t^i(\cdot)\}_{i=1}^M$ over T time points, and observing the Nash equilibrium responses $\{\mathbf{a}_t(\theta), t \in [T]\}$ for mechanism o_θ. There are several subtleties that should be pointed out to fully specify Algorithm 1:

- In order to make this "probing" specification precise, we define for each $i \in [M]$ a parametrized family of constraint functions $\{g^i(\cdot, \gamma)\}$, where γ is a parameter lying in some space Γ. Then, taking T parameters $\{\gamma_t, t \in [T]\}$, e.g., as instantiations of random variables with distributions on Γ, we form the constraints $g_t^i(\cdot) = g^i(\cdot, \gamma_t)$. This probing interaction can be seen on lines 14-15 of Algorithm 1.
- We assume that the mechanism parameters are taken from a compact space $\Theta \subset \mathbb{R}^p$, and so whenever an iterate θ_{k+1} leaves this space, we project it back onto the boundary of Θ by the function

$$\text{Proj}_\Theta(\theta) = \arg\min_{\gamma \in \Theta} \|\gamma - \theta\|$$

Apart from these specifications, the operation of Algorithm 1 follows the recursion (8) exactly, and terminates when a parameter $\hat{\theta}$ for which $L(\hat{\theta}) \leq 0$ is found.

C. Convergence

Here we demonstrate the efficacy of Algorithm 1, by showing its convergence to the global minima of the loss function (6). Before doing so, we must first characterize the conditions under which our problem is well-posed. Recall we assume, as is standard in the mechanism design literature, that for each "probe" constraint set $\{g_t^i(\cdot)\}_{i=1}^M$ and mechanism o_θ, the system outputs a Nash equilibrium response $\mathbf{a}_t(\theta)$ that satisfies (1). Therefore, we must ensure the existence of a Nash equilibrium for every mechanism parameter $\theta \in \Theta$. Then, for a given mechanism parameter set Θ, we must be sure that there exists some $\hat{\theta} \in \Theta$ for which a Nash equilibrium solution is socially optimal, otherwise the Algorithm 1 will never terminate.

In order to address these criteria, we must introduce the concept of 'potential modifiability':

Definition 4 (Potential Modifiable): A game form G is potential modifiable if there exists some mechanism parameter θ such that the utility functions $\{f_\theta^i(\cdot)\}$ admit a 'potential' function $\Phi(\cdot)$ such that for any $i \in [M], \mathbf{a} = (a_i, a_{-i}) \in \mathcal{A}, a_i' \in \mathcal{A}_i$,

$$f^i(a_i', a_{-i}) - f^i(a_i, a_{-i}) = \Phi(a_i', a_{-i}) - \Phi(a_i, a_{-i})$$

A potential modifiable game form G is one for which at least one mechanism parameter exists under which G is a potential game.

Remark: A necessary and sufficient condition for a game G to be a potential game is that each utility function $f_\theta^i(\mathbf{a})$ can be decomposed as a common function $\Phi(\mathbf{a})$ and a term depending only on actions a_{-i}. [14]

Now we introduce two assumptions on the game form G which enforce the well-posedness of our problem, and are also sufficient for global convergence of Algorithm 1.

Assumption 1: The utility functions $\{f_\theta^i(\mathbf{a})\}_{i=1}^M$ are each concave and increasing with respect to a_i, thrice continuously differentiable with bounded n-th order derivatives ($n = 1, 2, 3$) with respect to both a_i and θ, and potential modifiable.

Assumption 2: The constraint functions $g_t^i : \mathcal{A}_i \to \mathbb{R}$ are increasing and thrice continuously differentiable for all t and i.

Discussion of Assumptions: By [15], concavity of utility functions $\{f_\theta^i(\cdot))\}_{i=1}^M$ and convexity of the constraint regions defined by $\{g_t^i(\cdot)\}_{i=1}^M$ is sufficient to ensure existence of a Nash equilibrium solution. Concavity (or quasi-concavity) of utility functions is also widely assumed in the microeconomic and game theory literature [10]. Also observe that in a potential game a Nash equilibrium solution is automatically socially optimal by concavity of the potential function Φ. Therefore, there exists at least one mechanism parameter $\hat{\theta}$ which induces socially optimal Nash equilibrium, namely that parameter under which the game is a potential game. [3]

The additional differentiability structure imposed by Assumptions 1 and 2 is sufficient to guarantee convergence of Algorithm 1 to the set of global minima of $L(\theta)$. This is our second main result, and is stated as follows.

Theorem 2: Let $\hat{\Theta} = \{\theta \in \Theta : L(\theta) \leq 0\}$ and $\|\cdot\|$ denote L^2 norm. Under Assumptions 1 and 2, we have

$$\forall \epsilon > 0, \lim_{k \to \infty} \mathbb{P}(\|\theta_k - \hat{\Theta}\| > \epsilon) = 0,$$
$$\text{where } \|\theta_k - \hat{\Theta}\| = \min_{\hat{\theta} \in \hat{\Theta}} \|\theta_k - \hat{\theta}\|$$

i.e., Algorithm 1 converges in probability to the set of global minimizers of $L(\theta)$

Proof: See Appendix VI-B ∎

Theorem 2 guarantees that Algorithm 1 converges to a parameter $\hat{\theta}$ such that mechanism $o_{\hat{\theta}}$ induces an empirical dataset $\mathcal{D}_{\hat{\theta}}$ which is consistent with social optimality, i.e., satisfies (1) of Theorem 1.

Thus, Algorithm 1 provides a methodology for achieving adaptive mechanism design under basic assumptions on the game form G. In the following section, we implement Algorithm 1 numerically and verify its efficacy in converging to the global minima of $L(\theta)$.

IV. Numerical Implementation

Here we numerically verify the efficacy of Algorithm 1. We implement the following 'river pollution game', as presented in [16]. Three agents $i = 1, 2, 3$ are located along a river. Each agent i is engaged in an economic activity that produces pollution as a level x_i, and the

[3] While potential modifiability is a sufficient condition for existence of a socially optimal Nash equilibrium, it is not necessary. Indeed, we will show in our numerical simulations that a socially optimal Nash equilibrium is found for mechanism parameters not inducing a potential game. Still, potential modifiability is a simple and convenient assumption guaranteeing *at least* one socially optimal Nash equilibrium. A full characterization of the existence of socially optimal Nash equilibria in game structures is an interesting point for future research.

players must meet environmental conditions set by a local authority. Pollutants are expelled into a river, where they disperse. There are two monitoring stations $l = 1, 2$ along the river, at which the local authority has set the maximum pollutant concentration levels. The revenue for agent i is

$$R_i(x) = [d_1\sqrt{x_i} - d_2\sqrt{x_1 + x_2 + x_3}]$$

where d_1, d_2 are parameters that determine an inverse demand law [16]. Agent i has expenditure

$$F_i(x) = (c_{1i}\sqrt{x_i} + c_{2i}x_i)$$

and thus has total profit

$$f^i(x) = [d_1\sqrt{x_i} - d_2\sqrt{x_1 + x_2 + x_3} - c_{1i}\sqrt{x_i} - c_{2i}x_i] \quad (9)$$

The local authority imposes the constraint on total polution levels at each monitoring site as

$$q_l(x_1, x_2, x_3) = \sum_{i=1}^{3} \delta_{il} e_i x_i \leq 100, \ l = 1, 2 \qquad (10)$$

Parameters δ_{il} are transportation-decay coefficients from player i to location l, and e_i is the emission coefficient of player i.

We take the perspective of the designer (local authority), who *does not know* the agent utility functions (9), but can enforce the constraints (10) and can modify the parameters d_2, c_{1i}, c_{2i}. Observe that the parameter vector $[d_2, c_{11}, c_{12}, c_{1,3}, c_{21}, c_{22}, c_{23}] \in \mathbb{R}^7$ can be treated as a mechanism, since it modifies the agent utility functions and can be controlled by the designer. Also, the parameter vector $[e_1, e_2, e_3] \in \mathbb{R}^3$ can be used to modify the constraint functions, and "probe" the system, as discussed in Section III-A.

Furthermore, observe that the utility functions (9) and constraints (10) satisfy assumptions 1 and 2. The utility functions (9) are potential modifiable, as setting $d_1 = c_{1i} = c_{2i} = 0$ yields a trivial potential game. We emphasize that this potential modifiability condition is a sufficient condition for the existence of a mechanism inducing socially optimal Nash equilibria. However, in practice it is unlikely that this is also a *necessary* condition; we verify numerically that the algorithm converges to a non-trivial mechanism which does not correspond to a potential game.

We implement Algorithm 1 in Matlab. The non-cooperative game-enactment in steps 14-15 is achieved by the NIRA-3 Matlab package [16], which employs the relaxation algorithm and Nikaido-Isoda function to compute Nash equilibrium solutions. We then utilize the standard Matlab linear program solver to compute the optimization in lines 17-18. We set the lower bound on $L(\theta)$ to 0, so that the Nash equilibria are consistent with social optimality if and only if $L(\theta) = 0$. We take the constraint functions (10) as generated by random parameters $\gamma_t = [e_1, e_2, e_3] \sim U[0, 1]^3$ for each $t \in [T]$, i.e., each e_i is uniformly distributed on $[0, 1]$. The remainder of the algorithm is implemented with ease: we take $p = 7, c = a = 0.5, \eta = 1/4, q = 0.001, T = 10$.

As mentioned, we take mechanism parameter $\theta = [d_2, c_{11}, c_{12}, c_{1,3}, c_{21}, c_{22}, c_{23}]$ and $\Theta = [0, 1]^7$, that is, each element of θ is restricted to the unit interval $[0, 1]$. We initialize θ_1 as a 7-dimensional uniform random variable in $[0, 0.5]$, and iterate Algorithm 1 until $L(\theta) = 0$. Figure 3 illustrates the iterative value of r (6), corresponding to $L(\theta_n)$, as a function of n.

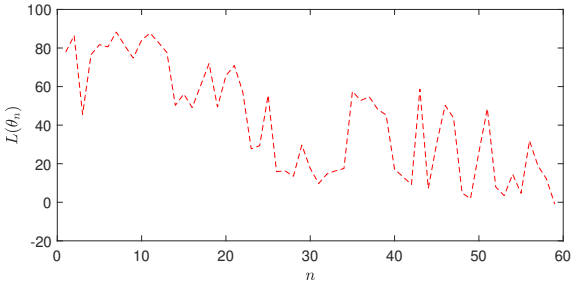

Fig. 3: Value of $L(\theta_n)$ for $n = 1, \ldots, 59$ in Algorithm 1. The algorithm achieves adaptive mechanism design by converging to θ_{59}, given in Table I, which induces empirical data that is consistent with social optimality by minimizing $L(\theta)(6)$.

n	d_2	c_{11}	c_{12}	c_{13}	c_{21}	c_{22}	c_{23}
1	0.198	0.309	0.137	0.258	0.125	0.229	0.228
59	0.018	0.567	0.058	0.167	0.257	0.108	0.424

TABLE I: Value of mechanism parameter θ_n for initial value $n = 1$ and final value $n = 59$. Algorithm 1 terminated at $n = 59$, indicating that θ_{59} induces socially optimal Nash equilibria.

Figure 3 demonstrates that Algorithm 1 converges to a parameter θ_{59} at iteration $n = 59$ such that the empirical dataset $\mathcal{D}_{\theta_{59}}$ produces $L(\theta) = 0$, and is thus consistent with social optimality in the sense of Theorem 1. Table I contains the parameter θ_n values for $n = 1$ and $n = 59$. It can be observed that θ_{59} does not induce a potential game, and thus the social optimality convergence is non-trivial.

V. Conclusion

We have provided a novel methodology for achieving mechanism design when the agent utility functions are unknown to the designer. We first generalized a seminal result in microeconomic revealed preferences, giving necessary and sufficient conditions for Nash equilibria behavior to be consistent with social optimality. Then, we utilized this to construct a loss function which has global minima corresponding to mechanisms inducing socially optimal Nash equilibria. We proved that a simulated annealing-based gradient algorithm will converge in probability to the set of global minima of this loss

function, and thus will allow us to achieve adaptive mechanism design. We also demonstrated the algorithm's efficacy numerically in a standard non-cooperative game setting.

References

[1] C. Yi, J. Cai, C. Yi, and J. Cai, "Fundamentals of Mechanism Design," *Market-Driven Spectrum Sharing in Cognitive Radio*, pp. 17–34, 2016.

[2] D. Mookherjee, "Decentralization, hierarchies, and incentives: A mechanism design perspective," *Journal of Economic Literature*, vol. 44, no. 2, pp. 367–390, 2006.

[3] S. Sengupta and M. Chatterjee, "An economic framework for dynamic spectrum access and service pricing," *IEEE/ACM Transactions on Networking*, vol. 17, no. 4, pp. 1200–1213, 2009.

[4] N. Nisan and A. Ronen, "Algorithmic mechanism design," in *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, 1999, pp. 129–140.

[5] V. Conitzer and T. Sandholm, "Complexity of mechanism design," *arXiv preprint cs/0205075*, 2002.

[6] F. Forges and E. Minelli, "Afriat's theorem for general budget sets," *Journal of Economic Theory*, vol. 144, no. 1, pp. 135–145, 2009.

[7] T. Börgers, *An introduction to the theory of mechanism design*. Oxford University Press, USA, 2015.

[8] P. A. Samuelson, "A note on the pure theory of consumer's behaviour," *Economica*, vol. 5, no. 17, pp. 61–71, 1938.

[9] S. N. Afriat, "The construction of utility functions from expenditure data," *International economic review*, vol. 8, no. 1, pp. 67–77, 1967.

[10] D. Fudenberg and J. Tirole, *Game theory*. MIT press, 1991.

[11] J. L. Maryak and D. C. Chin, "Global random optimization by simultaneous perturbation stochastic approximation," in *Proceedings of the 2001 American control conference.(Cat. No. 01CH37148)*, vol. 2. IEEE, 2001, pp. 756–762.

[12] H. J. Kushner and G. Yin, *Stochastic Approximation Algorithms and Recursive Algorithms and Applications*, 2nd ed. Springer-Verlag, 2003.

[13] M. Welling and Y. W. Teh, "Bayesian learning via stochastic gradient langevin dynamics," in *Proceedings of the 28th international conference on machine learning (ICML-11)*. Citeseer, 2011, pp. 681–688.

[14] X. Guo and Y. Zhang, "Towards an analytical framework for potential games," *arXiv preprint arXiv:2310.02259*, 2023.

[15] J. B. Rosen, "Existence and uniqueness of equilibrium points for concave n-person games," *Econometrica: Journal of the Econometric Society*, pp. 520–534, 1965.

[16] J. Krawczyk and J. Zuccollo, "Nira-3: An improved MATLAB package for finding Nash equilibria in infinite games," 2006.

[17] L. Cherchye, B. De Rock, and F. Vermeulen, "The revealed preference approach to collective consumption behaviour: Testing and sharing rule recovery," *The Review of Economic Studies*, vol. 78, no. 1, pp. 176–198, 2011.

[18] R. M. Freund, "Postoptimal analysis of a linear program under simultaneous changes in matrix coefficients," in *Mathematical Programming Essays in Honor of George B. Dantzig Part I*. Springer, 2009, pp. 1–13.

VI. Appendix

A. Proof of Theorem 1

$(1 \Rightarrow 2)$: Assume (4) holds. We have that $\mathbf{a}_t$ solves:

$$\max_{\mathbf{a}} \sum_{i=1}^{M} \hat{f}^i(\mathbf{a}) \; s.t \; g_t^i(\mathbf{a}) \leq 0 \; \forall i \in [M] \tag{11}$$

By concavity, the functions $\hat{f}^i(\cdot)$ are subdifferentiable, and thus the sum $\sum_{i=1}^{M} \hat{f}(\cdot)$ is subdifferentiable. Then, letting η_t^i be the Lagrange multiplier associated with budget constraint $g_t^i(\cdot)$, an optimal solution $\mathbf{a}_t$ to the maximization problem (11) must satisfy $\sum_{i=1}^{M} \nabla \hat{f}^i(\mathbf{a}_t) = \sum_{i=1}^{M} \eta_t^i \nabla g_t^i(\mathbf{a}_t)$ where $\nabla \hat{f}^i(\mathbf{a}_t)$ and $\nabla g_t^i(\mathbf{a}_t)$ are subgradients of $\hat{f}$ and g_t evaluated at $\mathbf{a}_t$. Let

$$\lambda_t^i := \frac{\left(\sum_{i=1}^{M} \eta_t^i \|\nabla g_t^i(\mathbf{a}_t)\|_\infty + \sum_{j \neq i} \|\nabla \hat{f}^j(\mathbf{a}_t)\|_\infty\right)}{\min_{k \in [n]} |(\nabla g_t^i(\mathbf{a}_t))_k|}$$

$$\Gamma_t^i := \nabla g_t^i(\mathbf{a}_t)$$

and observe that for each $i \in [M]$,

$$\nabla \hat{f}^i(\mathbf{a}_t) \leq \lambda_t^i \Gamma_t^i \tag{12}$$

Next, concavity of the functions $\hat{f}^i$ implies for each i

$$\hat{f}^i(\mathbf{a}_s) - \hat{f}^i(\mathbf{a}_t) \leq \nabla \hat{f}^i(\mathbf{a}_t)'(\mathbf{a}_s - \mathbf{a}_t) \tag{13}$$

Substituting inequality (12) into (13), and setting $v_s^i = \hat{f}^i(\mathbf{a}_s)$, $v_t^i = \hat{f}^i(\mathbf{a}_t)$, we obtain $v_s^i - v_t^i \leq \lambda_t^i(\Gamma_t^i)'(\mathbf{a}_s - \mathbf{a}_t)$ Thus the set $\{\Gamma_t^i, \mathbf{a}_t, t \in [T]\}$ satisfies the linear 'Generalizes Axiom of revealed preference' (GARP) [17] for each $i \in M$. We will show that this implies the dataset $\{g_t^i, x_t, t \in [T]\}$ satisfies *nonlinear* GARP [6]. First, assume $g_t^i(\mathbf{a}_s) < 0$. Since $g_t^i(\mathbf{a}_t) = 0$ we have $g_t^i(\mathbf{a}_s) < g_t^i(\mathbf{a}_t)$, and thus $\nabla g_t^i(\mathbf{a}_t)'(\mathbf{a}_s - \mathbf{a}_t) \leq 0$ by monotonicity of g_t^i. So $(\Gamma_t^i)'\mathbf{a}_s < (\Gamma_t^i)'\mathbf{a}_t$. Then by inverting this implication, we obtain $\Gamma_t^i \mathbf{a}_s \geq \Gamma_t^i \mathbf{a}_t \Rightarrow g_t^i(\mathbf{a}_s) \geq 0$.

Since the linear GARP is satisfied for all $i \in [M]$ we have the relation

$$\mathbf{a}_s R \mathbf{a}_t \Rightarrow \Gamma_t^i \mathbf{a}_t \leq \Gamma_t^i \mathbf{a}_s \Rightarrow g_t^i(\mathbf{a}_s) \geq 0$$

and thus $\{g_t^i, x_t, t \in [T]\}$ satisfies nonlinear GARP for all $i \in [N]$. So, by [6] we have that there for each $i \in [M]$, there exist $u_t^i \in \mathbb{R}, \lambda_t^i > 0$ such that (5) holds.

$(2 \Rightarrow 1)$: Assume (5) holds, take γ such that $g_t^i(\gamma) \leq g_t^i(\mathbf{a}_t) = 0 \; \forall i \in [M]$ and define

$$U^i(\gamma) = \min_{s \in [T]} \left[u_s^i + \lambda_s^i [g_s^i(\gamma)]\right]$$

In [6] it is shown that we can set $u_t^i = U^i(\mathbf{a}_t)$. Then we have that

$$\sum_{i=1}^{M} U^i(\gamma) \leq \sum_{i=1}^{M} \left[u_t^i + \lambda_t^i g_t^i(\gamma)\right] \leq \sum_{i=1}^{M} U^i(\mathbf{a}_t).$$

B. Proof of Theorem 2

Having the constraint sets generated via parameters $\{\gamma_t\}_{t=1}^{T}$, where each γ_t is a random variable from set Γ, the loss function becomes:

$$\begin{aligned} L(\theta) = \arg\min_{r} \; & s.t. \; \exists \; \{u_t^i, \lambda_t^i\} : \\ & u_s^i - u_t^i - \lambda_t^i g_t^i(\hat{\mathbf{a}}_s(\theta), \gamma_t) \leq r \; \forall t, s, i \end{aligned} \tag{14}$$

where $\hat{\mathbf{a}}_s(\theta)$ is a Nash equilibrium solution for the game with mechanism o_θ and feasible sets $\mathcal{A}_i = \{a : g_s^i(a, \gamma_t) \leq 0\}$.

We show that the loss function $L(\theta)$ and algorithm iterates satisfy assumptions H1-H8 of [11].

H1: Δ_n is distributed according to the p-dimensional Rademacher distribution, and thus the conditions are satisfied.

H2: We do not have random measurement noise in this framework. Thus the conditions are satisfied.

H3: Let $\theta = [\theta_1, \ldots, \theta_p]$. By Faá de Bruni's formula, the third derivative of $L(\theta)$ with respect to $\theta_k, k \in [p]$ can be decomposed as the combinatorial form:

$$\frac{\partial^3 L(\theta)}{\partial \theta_k^3} = \sum_{t,i} \sum_{\pi \in \Pi} \frac{\partial^{|\pi|} L(g_t^i(\hat{\mathbf{a}}_s(\theta), \gamma_t))}{\partial g_t^i(\hat{\mathbf{a}}_s(\theta), \gamma_t)^{|\pi|}} \cdot \prod_{B \in \pi} \frac{\partial^{|B|} g_t^i(\hat{\mathbf{a}}_s(\theta), \gamma_t))}{\partial \theta_k^{|B|}} \tag{15}$$

where π iterates through the set of partitions Π of the set $\{1, 2, 3\}$, $B \in \pi$ is an element of partition π, and $|\pi|, |B|$ indicate the cardinality of said partitions and elements, respectively. We need to show that (15) exists and is continuous for all $k \in [p]$.

Now, for notational simplicity re-write $g_t^i(\hat{\mathbf{a}}_s(\theta), \gamma_t))$ as $g_{t,s,i}$ and consider the term $\frac{\partial^n L(g_{t,s,i})}{\partial g_{t,s,i}^n}$ for some $n \in [3]$. This expresses the n-th order derivative of the solution to the linear program (14) with respect to real number $g_{t,s,i}$. Fortunately we can obtain a useful bound on this value by the methodology in [18]. Specifically, we can re-write the linear program (14) as

$$\text{maximize } L(g_{t,s,i}) = c \cdot x \ s.t. \ Ax = 0, \ x \geq 0 \tag{16}$$

where $x = [c, u_1^1, \ldots, u_T^M, \lambda_1^1, \ldots, \lambda_T^M]^T \in \mathbb{R}^{2TM+1}, c = [-1, 0, \ldots, 0] \in \mathbb{R}^{2TM+1}$ and A is a matrix of coefficients that produces the equalities $u_s^i - u_t^i - \lambda_t^i g_{t,s,i} - r = 0 \ \ \forall s, t, i$.

Then, letting $\beta \subset [2TM + 1]$, denote A_β as the submatrix of A with columns corresponding to elements of β, such that A_β is nonsingular. Form $\hat{x}_\beta$ be the basic primal solution to (16) with the $i'th$ element of $\hat{x}_\beta$ equal to zero for $i \in [2TM+1] \backslash \beta$. Also let G_β be the matrix A_β with all elements apart from $g_{t,s,i}$ replaced by zero. Then by Theorem 1 of [18], we have that for $n \in [3]$, $L(g_{t,s,i})$ is n-times differentiable, with derivatives given by $\frac{\partial^n L(g_{t,s,i})}{\partial g_{t,s,i}^n} = (n!)c(-A_\beta^{-1} G_\beta)^n \hat{x}_\beta$ and for $g \in \mathbb{R}$ sufficiently close to $g_{t,s,i}$, $\frac{\partial^n L(g)}{\partial g^n} = \sum_{i=n}^{\infty} \frac{i!}{(i-n)!} c(g - g_{t.s,i})^{(i-n)} (-B^{-1} G_\beta)^i \hat{x}_\beta$

This reveals that the $L(g_{t,s,i})$ is n-times continuously differentiable for $n \in [3]$.

Now, with $\{\hat{a}_{1,s}(\theta), \ldots, \hat{a}_{M,s}(\theta)\} = \hat{\mathbf{a}}_s(\theta)$ expressing the full vector of actions making up the Nash equilibrium $\hat{\mathbf{a}}_s(\theta)$, consider the term $\frac{\partial^n g_t^i(\hat{\mathbf{a}}_s(\theta), \gamma_t))}{\partial \theta_k^n} = \frac{\partial^n g_t^i(\hat{a}_{i,s}(\theta), \gamma_t))}{\partial \theta_k^n}$ for $n \in [3]$, where the equality holds since the constraint function $g_t^i(\cdot)$ only depends on action a_i. This term can again be decomposed using Faá de Bruni's formula as $= \frac{\partial^n g_t^i(\hat{a}_{i,s}(\theta), \gamma_t))}{\partial \theta_k^n} = \sum_{\pi \in \Pi} \sum_j \frac{\partial^{|\pi|} g_t^i(\hat{a}_{i,j,s}(\theta), \gamma_t)}{\partial \hat{a}_{i,j,s}(\theta_k)^{|\pi|}} \cdot \prod_{B \in \pi} \frac{\partial^{|B|} \hat{a}_{i,j,s}(\theta)}{\partial \theta_k^{|B|}}$ with Π being the partitions of $[n]$ here. The constraint function derivatives are bounded by assumption 2. The term $\frac{\partial^{|B|} \hat{a}_{i,j,s}(\theta)}{\partial \theta_k^{|B|}}$expresses the sensitivity of a Nash equilibrium solution to changes in the underlying utility function parameters.

Let $\mathbf{a}(\theta) = \{a_i\}$ be a Nash equilibrium solution for the game [G], where $a_i = \{a_{i,1}, \ldots, a_{i,k}\} \in \mathbb{R}^k$ is the action of player i. Let us begin by decomposing $\frac{\partial^n f_\theta^i(\mathbf{a}(\theta))}{\partial \theta_k^n} = \sum_{\pi \in \Pi} \frac{\partial^{|\pi|} f_\theta^i(\mathbf{a}(\theta))}{\partial \mathbf{a}(\theta)^{|\pi|}} \cdot \prod_{B \in \pi} \frac{\partial^{|B|} \mathbf{a}(\theta)}{\partial \theta_k^{|B|}}$ where again Π is the set of partitions of $[n]$. Begin with the base case $n = 1$. Then, $\frac{\partial f_\theta^i(a_{i,j}(\theta))}{\partial \theta_k} = \frac{\partial f_\theta^i(a_{i,j}(\theta))}{\partial a_{i,j}(\theta)} \frac{\partial a_{i,j}(\theta)}{\partial \theta_k}$ and

$$\frac{\partial a_{i,j}(\theta)}{\partial \theta_k} = \frac{\partial f_\theta^i(a_{i,j}(\theta))}{\partial \theta_k} \left(\frac{\partial f_\theta^i(a_{i,j}(\theta))}{\partial a_{i,j}(\theta)} \right)^{-1} \tag{17}$$

and thus by assumption 1, (17) is continuous. Now, assume $\frac{\partial_{i,j}^n(\theta)}{\partial \theta_k^n}$ is continuous for some n. Then, by decomposition $\frac{\partial^{n+1} f_\theta^i(a_{i,j}(\theta))}{\partial \theta_k^{n+1}} = \sum_{\pi \in \Pi} \frac{\partial^{|\pi|} f_\theta^i(a_{i,j}(\theta))}{\partial a_{i,j}(\theta)^{|\pi|}} \cdot \prod_{B \in \pi} \frac{\partial^{|B|} a_{i,j}(\theta)}{\partial \theta_k^{|B|}}$ we have that $\frac{\partial_{i,j}^{n+1}(\theta)}{\partial \theta_k^{n+1}}$ can be isolated and expressed as a polynomial in terms $\frac{\partial^m f_\theta^i(a_{i,j}(\theta)}{\partial a_{i,j}(\theta)^m}, m \in [n]$, $\frac{\partial^m a_{i,j}(\theta)}{\partial \theta_k^m}, m \in [n-1]$, and thus by the assumption 1, and the induction base, this term is also continuous.

Furthermore, observe that the n-th order derivatives are all bounded by their expressions in the previous derivations and assumption 1.

H4: The algorithm parameters are taken to satisfy these conditions

H5: Our algorithm is isolated to the compact set Θ, eliminating the necessity of asymptotic objective function structure such as this requirement.

H6: This is satisfied.

H7: Let P_η be a parametrized probability measure over the parameter space Θ, defined by the Radon-Nikodym derivative $\frac{dP_\eta(\theta)}{d\theta} = \frac{\exp(-2L(\theta)/\eta^2)}{Z_\eta}$ where $\eta > 0$ and $Z_\eta = \int_\Theta \exp(-2L(\theta)/\eta^2) d\theta < \infty$ is a normalizing term. Then the unique weak limit $P = \lim_{\eta \to 0} P_\eta$ is given by an indicator on points in the set $\hat{\Theta} := \{\theta : L(\theta) = 0\}$, i.e., $P(\theta) = \chi_{\theta \in \hat{\Theta}} / Z_0$ with χ the indicator function and $Z_0 = \int_\Theta \chi_{\theta \in \hat{\Theta}} d\theta$.

H8: The sequence is trivially tight, since $P(\theta_k \in \Theta) = 1 \, \forall k \in \mathbb{N}$