Position Paper: Social Environment Design

Edwin Zhang [1 2] **Sadie Zhao** [1] **Tonghan Wang** [1] **Safwan Hossain** [1] **Henry Gasztowtt** [3] **Stephan Zheng** [4]
David C. Parkes [1] **Milind Tambe** [1 5] **Yiling Chen** [1]

Abstract

Artificial Intelligence (AI) holds promise as a technology that can be used to improve government and economic policy-making. This paper proposes a new research agenda towards this end by introducing **Social Environment Design**, a general framework for the use of AI for automated policy-making. The position of this paper is that **Social Environment Design should be further studied as a research agenda by the Reinforcement Learning, EconCS, and Computational Social Choice communities.** The framework extends mechanism design to capture a fully general economic environment, including voting on policy objectives, and gives a direction for the systematic analysis of government and economic policy through AI simulation. We highlight key open problems for future research in AI-based policymaking. By solving these challenges, we hope to achieve various social welfare objectives, thereby promoting more ethical and responsible decision making.

1. Introduction

Macroeconomic policy formulation is a domain fraught with complexity, with traditional economic models providing limited foresight into the outcomes of policy decisions. Policy-makers must not only understand the immediate implications of individual policies but also their aggregate and long-term effects. In addition, human policy-maker incentives are often not aligned with the interests of the general public, and may instead prioritize special interests or reelection (de Figueiredo & Richter, 2014). In light of this, AI-based approaches to policy design that can simulate economies and target different objectives, hold the potential for improved policy understanding and formulation (Zheng et al., 2022; Koster et al., 2022).

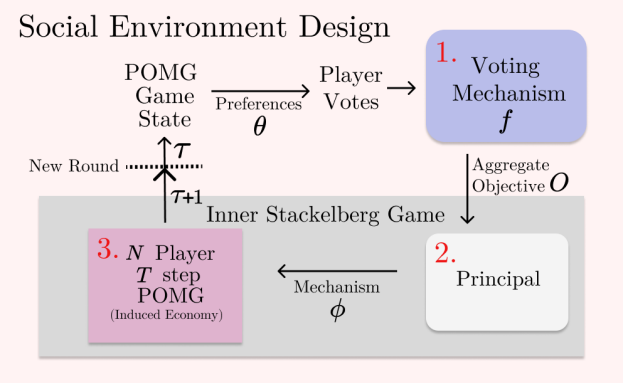

Figure 1: *The proposed framework.* The process begins with voting, where human or AI players report preferences on social welfare objectives to a voting mechanism (1). This defines an objective for the Principal, who designs a parameterized N-player Partially Observable Markov Game (POMG) (2). The players are the same as the voters. This POMG unfolds over several timesteps T (3). Following the POMG, game state information is extracted to initiate a new round of voting, with the last POMG state used as the first game state of the new round. This whole process is repeated for τ timesteps.

In an era where AI is gaining increasing attention across the government apparatus (House, 2023; Engstrom et al., 2020), understanding its potential influence on future policy-making through a rigorous framework has never been more critical. Ideally, such a theoretical framework should have the following desiderata:

1. Ensures **alignment of policy-makers** to the values of its constituents, whilst ensuring fair and equitable representation (Barocas et al., 2023).
2. Holds sufficient **model expressivity** (Patig, 2004) to accurately represent the intricate governance structures found in the real world, capturing the subtleties and variances of socio-economic interactions.
3. Balances expressiveness with **computational tractability**, making it feasible to scale to systems with a large number of agents.

[1]Harvard University [2]Founding AI [3]Oxford University [4]Asari AI [5]Google Research. Correspondence to: Edwin Zhang <ezhang@g.harvard.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

4. Provides **theoretical clarity**, enabling systematic analysis and offering a reductive approach to complex economic models.

In this paper, we propose a new theoretical framework, Social Environment Design, that attempts to make progress towards these desiderata. In our framework, illustrated in Figure 1, we suggest addressing the concern of a misaligned policy-maker with "Voting on Values (Hanson, 2013)", coupled with a Principal policy-maker who seeks to achieve the suggested policy goals. We capture the complexity of a general economic environment whilst maintaining computational tractability by modeling the economy as a Partially Observable Markov Game (POMG), which maintains a fixed observation space for each agent. Finally, we structure our framework as repeatedly finding Stackelberg Equilbria, enabling greater theoretical clarity by allowing reduction to simpler subproblems.

We now restate the position of this paper: **Social Environment Design should be further studied as a research agenda by the RL, EconCS, and Computational Social Choice communities.** Towards this end, we discuss several open problems within the framework of both practical and theoretical interest. By introducing this framework, we open a dialogue on AI's application to macroeconomic policy design, aspiring to someday help leverage AI to assist policymakers in enhancing economic resilience and governance effectiveness.

In summary, we list our core contributions below: 1) We propose the Social Environment Design framework to enable future research in AI-led policymaking in complex economic systems. 2) We release a simple core implementation of our framework as a Sequential Social Dilemma Environment along with code. 3) We provide a characterization of open problems within this area, along with prospective solution concepts and algorithmic approaches to forward the dialogue on AI's application in macroeconomic policy design.

2. Preliminaries

Here we give some preliminaries on several foundational games and solution concepts that we build upon.

Definition 2.1. A $(n+1)$-player **Stackelberg-Nash Game** $\mathcal{S} = (n, m, \mathcal{X}, \mathcal{Y}, \boldsymbol{u})$ comprises one player called the **leader** and $n \in \mathbb{N} \setminus \{0\}$ players called **followers**. In a Stackelberg-Nash game, the leader first commits to an action $\boldsymbol{x} \in \mathcal{X}$ from action space $\mathcal{X} \subset \mathbb{R}^m$. Then, having observed the leader's action, each follower $i \in [n]$, responds with an action y_i in their action space $\mathcal{Y}_i \subset \mathbb{R}^m$. We define the followers' joint action space $\mathcal{Y} = \bigtimes_{i\in[n]} \mathcal{Y}_i$. We refer to a collection of actions $\boldsymbol{y} = (y_1, \cdots, y_n) \in \mathcal{Y}$ as a followers' action profile, and to a collection $(x, y) \in \mathcal{X} \times \mathcal{Y}$ as an action profile.

After all players choose an action, the leader receives payoff $u_o(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}$, while each follower $i \in [n]$ receives payoff $u_i(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}$. Each player $i \in [n]$ aims to maximize her payoff, and the leader aims to maximize her payoff assuming the followers will best respond.

Fixing the leader's action $\boldsymbol{x} \in \mathcal{X}$, a Stackelberg Nash game $\mathcal{S}$ induces a **lower-level Nash game** $\mathcal{G}^{\mathcal{S}} = (n, m, \mathcal{Y}, \boldsymbol{u}_{-0}(\boldsymbol{x}, \cdot))$ among the followers.

Definition 2.2. A **Partially Observable Markov Game** (POMG) $\mathcal{M}$ with n agents is a tuple $(S, A, T, r, \Omega, B, \gamma, \mu_0)$. Here, S is a shared state space for all agents; $A = \bigtimes_{i\in[n]} A_i$ is the joint action space; $T : S \times S \times A \to [0, 1]$ is a stochastic transition function; $r : S \times A \to \mathbb{R}^n$ is the reward function with $r = (r_1, \cdots, r_n)$; $\Omega = \bigtimes_{i\in[n]} \Omega_i$ is the joint observation space; $B : \Omega \times S \times A \to [0, 1]$ is the stochastic observation function; $\gamma \in [0, 1)$ is a discount factor; $\mu_0 \in \Delta(S)$ is the initial state distribution. An agent's behavior in this game is characterized by its policy $\pi_i : \Omega \to A$, which maps observations to actions.

Definition 2.3. A $(n+1)$-player **Stackelberg-Markov Game** $\mathcal{S} = (n, m, \Phi, \Pi, \boldsymbol{u})$ comprises one player called the **leader** and $n \in \mathbb{N} \setminus \{0\}$ players called **followers**. In a Stackelberg-Markov game, the leader first commits to an action $\phi \in \Phi$ from action space $\Phi \subset \mathbb{R}^m$ which induces a n-player **low-level (Partially Observable) Markov Game** $\mathcal{M}^\phi = (S, A^\phi, T^\phi, r^\phi, \Omega^\phi, B^\phi, \gamma^\phi, \mu_0^\phi)$. Then, having observed the leader's action, each follower $i \in [n]$, responds with an policy $\pi_i : \Omega \to A_i$ in their policy space Π_i. We define the followers' joint action space $\Pi = \bigtimes_{i\in[n]} \Pi_i$. We refer to a collection of policies $\pi = (\pi_1, \cdots, \pi_n) \in \Pi$ as a followers' policy profile.

After all players choose an action, the leader receives payoff $u_o(\phi, \pi) \in \mathbb{R}$, while each follower $i \in [n]$ receives payoff $u_i(\phi, \pi) = \mathbb{E}^{\mathcal{M}^\phi, \pi}[\sum_{t=0}^{\infty} (\gamma^\phi)^t r^\phi(s^t, a^t)] \in \mathbb{R}$. Each player $i \in [n]$ aims to maximize her payoff, and the leader aims to maximize her payoff assuming the followers will best respond.

For all followers $i \in [n]$, we define the **δ-best-response correspondence** $\mathcal{BR}_i^\delta(\phi, \pi_{-i}) = \{\pi_i \in \Pi_i \mid u_i(\phi, \pi) \geq \max_{\pi_i \in \Pi_i} u_i(\phi, (\pi_i, \pi_{-i})) - \delta\}$ and the **joint δ-best-response correspondence** $\mathcal{BR}^\delta(\phi, \pi) = \bigtimes_{i\in[n]} \mathcal{BR}_i^\delta(\phi, \pi_{-i})$.

Definition 2.4. A **(ε, δ)-strong Stackelberg-Markov-Nash equilibrium (SSMNE)** in a Stackelberg-Markov game $\mathcal{S} = (n, m, \Phi, \Pi, \boldsymbol{u})$ is an action profile $(\phi^*, \pi^*) \in \Phi \times \Pi$ such that $u_0(\phi^*, \pi^*) \geq \max_{\phi\in\Phi} \max_{\mathcal{BR}^\delta} u_0(\phi, \pi) - \varepsilon$ and $u_i(\phi^*, \pi^*) \geq \max_{\pi_i \in \Pi_i} u_i(\phi^*, (\pi_i, \pi^*_{-i})) - \delta$, for all $i \in [n]$.

Definition 2.5. A **(One-Shot) Mechanism Design** problem $\mathcal{P} = (n, m, \mathcal{T}, S, \boldsymbol{t}, \boldsymbol{u}, u_0, f)$ comprises of n agents, each $i \in [n]$ owns a private type $\boldsymbol{t}_i \in \mathcal{T}_i$ from a set of possible types $\mathcal{T}_i \subset \mathbb{R}^m$.

An agent's preferences over outcomes $\boldsymbol{s} \in S$, for a set S of outcomes, can be expressed in terms of a utility function that is parameterized by the type. Let $u_i(\boldsymbol{s}, \boldsymbol{t}_i)$ denote the utility of agent i for outcome $\boldsymbol{s} \in S$ given type $\boldsymbol{t}_i$. A strategy (policy in RL) $s_i : \mathcal{T}_i \to \mathcal{A}_i$ is a complete decision rule, that defines the action an agent will select in every distinguishable state of the world. Let $a_i = s_i(\boldsymbol{t}_i) \in \mathcal{A}_i$ denote the action of agent i given type $\boldsymbol{t}_i$, where $\mathcal{A}_i$ is the set of all possible actions available to agent i.

A **mechanism** $M = (\{\mathcal{A}_i\}_{i\in[n]}, g)$ defines the set of actions $\mathcal{A}_i$ available to each agent i, and an outcome rule $g : \bigtimes_{i\in[n]} \mathcal{A}_i \to S$, such that $g(\boldsymbol{a})$ is the outcome implemented by the mechanism for action profile $\boldsymbol{a} = (a_1, \cdots, a_n)$. $u_0 : \mathcal{M} \times \bigtimes_{i\in[n]} \mathcal{A}_i \to \mathbb{R}$ is a principal objective function, where $u_0(M, \boldsymbol{a})$ represents the expected utility/revenue of the principal when mechanism designer chooses mechanism M and agents choose action profile $\boldsymbol{a}$. Note that u_0 is an alternative way of defining the social choice function. The goal of the mechanism designer is to design a mechanism $M = (\{\mathcal{A}_i\}_{i\in[n]}, g) \in \mathcal{M}$ that maximizes $u_0(M, \boldsymbol{a}^*)$, where strategy profile $\boldsymbol{a}^* = (a_1^*, \cdots, a_n^*)$ is an (Nash, Bayesian-Nash, dominant-strategy) equilibrium to the game induced by M.

3. Formal Definition of Social Environment Design Game

We define the Social Environment Design Game formally as repeatedly finding a Stackelberg Equilibrium in a Markov Game (Gerstgrasser & Parkes, 2023; Brero et al., 2022), iterated over several rounds of voting.

At a high level, we frame the economic design problem as a Stackelberg game between the policy designer and economic participants. The economic participants first vote for a given objective, or values to optimize for. Subsequently, the Principal (leader) attempts to maximize this objective by designing the rules of an economic system, which induce an environment for the participants. We model this environment as a Partially Observable Markov Game (POMG) with theparticipants as the agents. We refer to this as the **Social Environment Design Game** because it generalizes mechanism design in a number of ways; e.g., it involves voting on goals, and it involves the design of an economic policy for an economic environment in which agents both take actions and report types.

Definition 3.1. A **Social Environment Design Game** $\mathcal{S} = (\Phi, P, \phi_0, D, \delta, \Theta, \mathcal{O}, f)$ is a one-leader-n-follower online[a] Stackelberg-Markov Game, where

- $\Phi \subseteq \mathbb{R}^k$ is the principal action space;
- $P : \Phi \mapsto \mathcal{M}^\phi$ is a policy implementation map that maps from a principal action $\phi \in \Phi$ to a parameterized POMG $\mathcal{M}^\phi = (S, A^\phi, T^\phi, r^\phi, \Omega^\phi, B^\phi, \gamma^\phi, \mu_0^\phi)$;
- $\phi_0 \in \Phi$ is some initial action;
- $D : \Phi \times \Phi \mapsto \mathbb{R}_{\geq 0}$ is a divergence measure on the leader action space;
- $\delta > 0$ is the divergence constraint;
- $\Theta \subseteq \mathbb{R}^{(n+1)\times m}$ is the type space.
- $\mathcal{O} = \{O_i\}_{i\in[m]}$ is some set of predefined social welfare functions, where each O maps $\Phi \times \Pi \mapsto \mathbb{R}$. We give examples of several possible choices of objectives below in Social Welfare Examples. Π here refers to the set of all possible policy profiles in the parameterized POMG;
- $f : \Theta \mapsto \mathcal{O}$ is a social choice function representing the voting mechanism.

[a] Here, online means that the Stackelberg-Markov Game is repeatedly played, with the first state of a new round made equal to the final state of the last round.

Further analysis and breakdown of Definition 3.1.

First we make a note regarding our type space. Since we define the first row of a specific type instantiation $\theta \in \Theta$ to be the type of the principal, Θ has $(n + 1)$ rows. We thus refer to Θ_1 to be the type space of the principal and Θ_{-1} to be the type space of all participants. In addition, Θ can be added to the state space of the POMG, which allows dynamic types that change over time in response to the state of the game. We do not allow the Principal to directly manipulate or observe the state space. Thus, we can embed the type space within the state space to hide it from the Principal. Even with elements of the POMG that the Principal does have control over, such as the state transition function T^ϕ, one can enforce hard constraints on how much power the Principal has to change the function explicitly through the divergence D or implicitly through the implementation map P.

We remark that both the infinite-horizon and finite-horizon version of the Social Environment Design Game can be considered. In contrast to standard Reinforcement Learning (RL), we do not need to introduce a discount factor for the infinite-horizon version, as we consider the Principal only

maximizing for the objective at the current voting round since our model's objective changes at each round. This is perhaps a naive objective, and other continual objectives could instead be considered. Exploring the tradeoffs between the greedy objective and more complex continual objectives is left as an important open problem for future work. In the finite horizon case, we add an additional time horizon $\mathcal{T}$ to our game. We now proceed to a detailed breakdown of our game.

The Social Environment Design Game can be cleanly divided into a **Voting Mechanism** and **Stackelberg Game**, which is played with the Principal's objective determined by the Voting Mechanism.

Definition 3.2. The **Voting Mechanism** is defined as $\mathcal{V} = (\mathcal{O}, f, \Theta)$.

We use the standard axiomatic model (Arrow, 2012), where $\mathcal{O}$ is the set of alternatives, f is the social choice function, and Θ is the type space, or set of all preference profiles. Intuitively, a specific agent i's type θ_i for row i in $\theta \in \Theta$, can be thought of as some latent vector which represents the agent's values. This type contains all information necessary for recovering a partial ordering over alternatives, a more specific way of defining preferences. The goal of the Voting Mechanism is then to define a objective for the Principal to optimize, given these types. To do so, we define the Voting Mechanism f and ask the players for a preference report $\theta_{-1} \in \Theta_{-1}$, which does not necessarily have to be truthful. It remains an open problem whether some notion of approximate incentive compatibility can be achieved by the principal. The Voting Mechanism then computes the objective $O = f(\theta_1, \theta_{-1})$ as a result of the vote. Here, we set $\theta_1 \in \Theta_1$ to be the preferences of the Principal. Importantly, the objective function includes the full θ, which allows expressing preferences of the principal if one wished to encoded a form of "moral objectivity", or other biases. We also make this modeling choice for generality, as it allows our model to express mechanisms such as auctions where the objective of the principal may be entirely selfish and not depend at all on the participant's types.

Social Welfare Examples. Examples of social welfare functions that could be included in the voting set are the Utilitarian objective $O(\phi, \pi) = \sum_i J(\pi_i^\phi)$, where J is the expected discounted return $J = \sum_t (\gamma^\phi)^t r_i^\phi(s_t, a_{i,t}, a_{-i,t})$, π is the tuple of all agents $\pi = (\pi_i)_{i\in[n]}$, and π_i are singular agents that map $\Omega_i^\phi \to A_i^\phi$. Other possible choices include the Nash Welfare objective $O = \left(\Pi_i J(\pi_i^\phi)\right)^{1/n}$, as well as the Egalitarian objective $O = \min_i J(\pi_i^\phi)$. These are perhaps the most commonly discussed objectives, but bespoke or custom welfare functions could also be considered and added to the set of alternatives.

Definition 3.3. The **Stackelberg Game** $I = (\Phi, P, D, \delta, \phi_0)$ is a Stackelberg-Markov Game.

The Stackelberg Game game is played subsequently after the Voting Mechanism, and can be thought of as a single timestep of the full game. The Principal (leader) will choose action $\phi \in \Phi$ which induces a parameterized **Induced Economy** $\mathcal{M}^\phi = (S, A^\phi, T^\phi, r^\phi, \Omega^\phi, B^\phi, \gamma^\phi, \mu_0^\phi)$ through the policy implementation map $P : \phi \mapsto \mathcal{M}^\phi$. Note that if agent preferences change over time, this can be modeled by adding agent types into the state space of the POMG. The transition function T would then be able to express changes in preferences over time. Thus, the objective of the leader in the Stackelberg Game game is to design a POMG, given the objective O decided prior in the Voting Mechanism:

$$\begin{aligned} &\max_\phi O(\phi, \pi) \\ \text{s.t.} \quad &D(\phi_0, \phi) \le \delta \\ &\mu_0^{P(\phi)} = \Delta(s_T). \end{aligned} \tag{1}$$

Again, π here is the tuple of all agents $\pi = (\pi_i)_{i\in[n]}$, and π_i individual agents that map $\Omega_i^\phi \to A_i^\phi$. Our notation $\mu_0^{P(\phi)}$ denotes the μ_0 of the tuple $P(\phi)$, and $\Delta(s_t)$ refers to a Delta Dirac distribution centered on s_T. Therefore, the second constraint $\mu_0^{P(\phi)} = \Delta(s_T)$ forces the ϕ to choose a POMG that has the same initial state as the terminal state of the last round, so that continuity is kept between rounds.

Lastly, we remark that this constrained optimization can also be transformed into an unconstrained problem by using an additional reparameterization $\mathcal{R} : \xi \mapsto \hat{\Phi}$, where $\xi \in \Xi := \mathbb{R}^L$ and $\hat{\Phi} := \{\phi \mid D(\phi_0, \phi) \le \delta\}$. The optimization can then proceed in $\mathbb{R}^L$ with no constraints. In this case, the Stackelberg game would reduce to $I = (\Xi, P')$, where $P' = P \circ \mathcal{R}$.

Definition 3.4. The **Induced Economy** is a Partially Observable Markov Game $\mathcal{M}^\phi = (S, A^\phi, T^\phi, r^\phi, \Omega^\phi, B^\phi, \gamma^\phi, \mu_0^\phi)$.

Finally, the **Induced Economy** is defined as the POMG produced as the output of the principal. Agents within the POMG interact with one another and attempt to maximize their utility according to their true preferences. The n economic participants (followers) will play strategically in the parameterized POMG $\mathcal{M}^\phi$. At each step t of the game, every follower i chooses an action $a_{i,t}$ from their action space A_i, the game state evolves according to the joint action $\boldsymbol{a}_t = (a_{1,t}, \cdots, a_{n,t})$ and the transition function T, and agents receive observations and reward according to B and r. An agent's behavior in the game is characterized by its policy $\pi_i : \Omega_i^\phi \to A_i^\phi$, which maps observations to actions. Each follower in the POMG $\mathcal{M}^\phi$ individually seeks to maximize its own (discounted) total return

$\sum_t (\gamma^\phi)^t r_i^\phi(s_t, a_{i,t}, a_{-i,t})$.

4. Example: Apple Picking Game

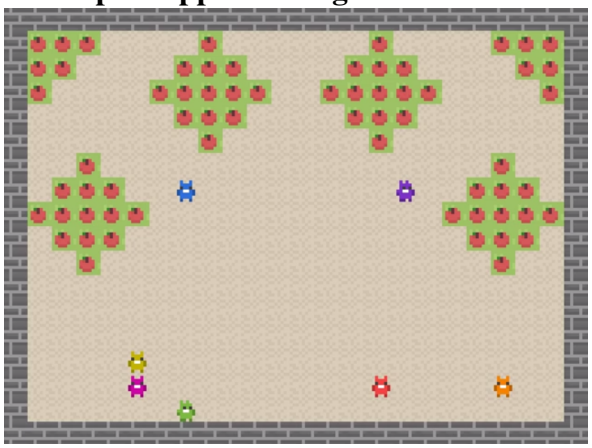

Figure 2: An example of social environment design as an apple picking game, built with Melting Pot 2.0 (Agapiou et al., 2022). Player agents observe a restricted view of their environment, and receive a mixed reward depending on the apples they collect and the apples their observable neighbors collect. The principal observes both an unrestricted view of the environment and the running totals of all the players' cumulative extrinsic rewards, where it collects tax and redistributes wealth on the cumulated rewards at the end of every tax period (50 timesteps), similar to Zheng et al. (2022). Rewards for the Principal are determined by the change in value of the objective it is currently assigned by agent votes. For training details and hyperparameters, please refer to Appendix A.

In order to give a motivating example for how preference elicitation for the principal in Social Environment Design can be used to align policy-maker incentives, we have created a Sequential Social Dilemma Game inspired by the *Harvest Game* proposed in Perolat et al. (2017). The game is illustrated in Figure 2The aim in *Harvest Game* is to collect apples, with each apple yielding a reward. If all apples in an area are harvested, they never grow back. The dilemma arises when individual self-interest drives rapid harvesting, which could permanently deplete resources. Thus, agents must sacrifice personal benefit and cooperate for collective well-being. One potential solution to this dilemma is through the use of a central government that taxes and redistributes apples. Thus, we have created a new game inspired by Zheng et al. (2022) in which a principal designs tax rates on apple collection and players vote on Utilitarian (productivity) vs. Egalitarian (equality) objectives for the principal. As players interact within this evolving environment, the principal faces the challenge of crafting policies that balance immediate economic incentives with sustainability goals. In order to achieve this, the principal must foster cooperation among players, guiding them towards the Pareto-Efficient equilibrium that the players have chosen.

To build intuition for how the Apple Picking Game maps onto the theoretical framework described in section 3, we give a more formal definition of the game here. Recall the definition of a Social Environment Design Game $\mathcal{S} = (\Phi, P, \phi_0, D, \delta, \Theta, \mathcal{O}, f)$. The action space for the environment designer Φ is defined as tax weights $0 \leq \Phi_i \leq 1$, determining the percentage of income to be taxed for each bracket. For simplicity we evenly redistribute the taxes, and set the number of tax brackets to three. In this environment, the Principal can only change the reward function of the induced POMG, so $\mathcal{M}^\phi = (S, A, T, r^\phi, \Omega, B, \gamma, \mu_0)$. The type of each agent is defined as (σ, β), where σ refers to the selfishness of the agent and β refers to the trust the agent holds in the Principal. In some sense this is a rough approximation for the political spectrum (Heywood, 1998). Finally, let a_i be the number of apples collected for a given agent i. We can now define the policy implementation map P, which in this case reduces to the parameterization of the reward function:

$$r_i(a, \phi) = \beta_i r_{\text{extrinsic}}(a, \phi) + (1 - \beta_i) r_{i,\text{intrinsic}}(a). \quad (2)$$

Here, the intrinsic reward is an average between the apples an agent collects and the apples all other agents collect within its field of view weighted by the selfishness of the agent. On the other hand, the extrinsic reward is the amount of apples after tax an agent collects plus an equal share of the redistributed total tax:

$$r_{i,\text{intrinsic}}(a) = \sigma_i a + (1 - \sigma_i)(\sum_j a_j - a),$$

$$r_{\text{extrinsic}}(a, \phi) = (a - T(a, \phi)) + \frac{1}{n} \sum_j T(a_j, \phi),$$

$$\text{where tax } T(a, \phi) = \sum_{b=0}^{B-1} \phi_b \cdot ((\tau_{b+1} - \tau_b)\, \mathbf{1}[a > \tau_{b+1}] + (a - \tau_b)\, \mathbf{1}[\tau_b < a \leq \tau_{b+1}]).$$

Here, $[\tau_b, \tau_{b+1}]$ refer to the tax brackets. Importantly, the principal can only incentivize agents through the extrinsic reward and cannot directly observe the intrinsic reward. The degree to which the principal holds power over the agent depends on the agent's trust or belief in the principal. We sample both selfishness and trust uniformly over [0, 1], and keep them fixed during training. Allowing them to change over time either randomly or in some fashion dependent on the performance of the Principal is left for future work. The objective space of the Principal is defined as $\mathcal{O} = \{\eta \sum_{i,t} a_{i,t} + (1 - \eta) \min_i \sum_t a_(i, t) \mid 0 \leq \eta \leq 1\}$, an interpolation between the Utilitarian and Egalitarian objective. A simple social choice function $f(\sigma) = \eta$ can be defined as the average of agent selfishness: $f(\sigma) = \frac{1}{n} \sum_i \sigma_i$. In this setting, we leave the Principal optimization unconstrained and thus do not need to define D or δ. ϕ_0 is initialized to 0, or no tax.

We run several tax periods per voting round, and at the end of each the principal decides on a new tax rate, for each

bracket, for the next period - as well as calculating, applying and redistributing tax to the players for that entire period, delivered in the players' final extrinsic reward. We release a high quality version of our code designed for fast experimentation and further research in the supplementary material, and include environment hyperparameters and training details in Appendix A. We do not include results as the primary purpose of this paper is to propose a future research agenda and illustrate open problems within it. Preliminary results were promising, but further analysis is warranted.

5. Challenges and Open Problems

Based on the AI-led economic policy-making framework presented, the following key open problems of our framework are proposed for further exploration: **Preference aggregation and democratic representation** in voting mechanisms is a complex challenge that requires advanced algorithms to reflect collective preferences while respecting minority views, as well as ensuring that the simulated population is representative and their preferences correctly modeled. **Modeling human behavior** within the simulator is another key challenge, and points towards possibly incorporating bounded-rationality into MARL (Wen et al., 2019a) or role-based modeling (Wang et al., 2020b; 2021b). To ensure responsible **AI governance and accountability**, responsible oversight mechanisms must be established. Furthermore, exploring socioeconomic interactions within these systems is critical, especially in understanding and deriving the conditions for **convergence to and definition of the Principal's objective**. As our framework is positioned within a continual learning setting, it is important to redefine what an optimal Principal looks like in this context. Finally, **scaling laws** of the framework should be analyzed in order to fully model real-world complexities. Can the framework handle simulating economies with thousands or millions of agents? What is the role of scale? When is simulation useful, and when does it fail? These remain important open problems for future research. In addition, we now give a more detailed outline for several of the above problems for further consideration and thought.

Preference Aggregation and Democratic Representation.

Aggregation algorithms within the Voting Mechanism: The development of sophisticated algorithms that can effectively aggregate disparate and potentially conflicting preferences of diverse agent populations is a significant challenge. These algorithms must ensure that the outcomes represent collective preferences without overwhelming the minority views.

Incorporating diverse decision-making models: The framework must be flexible enough to respect various cultural, ethical, and socioeconomic decision-making paradigms that different groups of agents might exhibit. In addition, such agents should imitate humans well, which we expand on further in the next section.

Modeling Human Behavior.

Bounded Rationality: Human decision-making in economic settings often demonstrates bounded rationality, where decisions are made based on satisficing rather than optimizing behavior. Further research is required to develop AI agents that can capture such nuances in human decision-making.

Cognitive and Behavioral Biases: Human economic behavior is influenced by a variety of cognitive biases. For instance, time-inconsistent preferences can lead to procrastination and problems with self-control, and loss aversion can skew risk preferences. AI agents within this framework need to capture these biases for accurate representation of human economic behavior.

Interaction and Network Effects: Humans do not make economic decisions in isolation; their decisions are profoundly influenced by their interactions with others. This opens another avenue of research in modeling these network effects accurately within the agent behavior models.

AI Governance and Accountability.

Transparent decision-making processes: AI systems involved in policy-making need to have their decision-making processes be fully transparent. The creation of interpretable AI models that can provide explanations for suggested policies is essential for trust and accountability.

Legal and ethical frameworks for AI decisions: There is an urgent need to establish legal and ethical frameworks that delineate the responsibilities and liabilities associated with AI-driven decision-making. These frameworks should set guidelines for what constitutes fair and lawful AI behavior in an economic context.

Oversight and human-AI collaboration: Establishing effective oversight mechanisms that involve both AI and human collaboration is critical. The role of human experts in supervising and guiding AI decisions, and their ability to intervene when AI-driven policies deviate from desired outcomes is still to be determined.

Convergence to Desired Outcomes.

Existence and characterization of forms of convergence or equilbria: Conventional game-theoretic equilibria may not be the right object of study in the first place in such settings, as empirically these economic systems likely will never converge, and instead a continual objective should likely be adopted. In addition, fundamental work is required to characterize the conditions under which a stable equilibrium might exist in such complex socioeconomic interactions. The uniqueness or multiplicity of equilibria and the conditions leading to each scenario need in-depth exploration.

Influence of dynamic changes on convergence points: The complex dynamics of economic systems call for a deep understanding of the sensitivity of equilibria to shocks and changes in the environment and agent behavior from variables that may have been unforeseen by the principal. Ensuring the robustness and stability of the principal to be able to recover from such shocks is also of importance.

Scaling Laws and Computational Efficiency.

Scaling up the model to larger systems: The proposed framework needs to be scaled to simulate economies of increasingly complexities. This comprises accommodating an increasing number of agents and more intricate interactions among them. Scaling laws of the model parameters and the computational resources required need to be examined.

Efficient learning and decision-making algorithms: Efficient algorithms for learning agent behavior and optimizing the policy design are crucial for the practicality of the framework. Particularly, the principal must be extremely sample efficient, as every step it takes induces an entire MARL optimization.

Massive parallelization: To tackle real-world complex systems, embracing the advantage of high-performance computing is necessary. This includes implementing the framework with massively parallel computations for both the learning and the decision-making processes. Techniques for splitting these processes into smaller tasks that can be processed simultaneously, as well as the efficient management of these tasks, represent challenging aspects to be addressed.

6. Related Work

The proposed framework resides between various strands of research, including but not limited to economic policy design, Stackelberg game learning, multi-agent reinforcement learning, mechanism design, and computational social choice. In this section, we delve into a comprehensive exploration of its connections with prior research.

6.1. Economic Policy Design and Simulation

Several approaches to automated economic policy design have been proposed in the past (Liu et al., 2022; Curry et al., 2023; Yang et al., 2021), and how usage of AI may span both participation in and design of economic systems (Parkes & Wellman, 2015). Here we cite several that are most related to our proposed framework and research agenda. Perhaps most related to our approach is the Human Centered Mechanism Design line of work from Koster et al. (2022); Balaguer et al. (2022). They propose learning mechanisms from behavioral models trained on human data, with the mechanism objective attempting to satisfy a majoritarian vote of the human participants. However, their work differs from ours in several key ways; firstly, they do not consider a fully general economic environment and limit their scope only to a generalization of the linear public goods setting. In other words, our framework encompasses Environment Design whilst theirs encompasses only Mechanism Design. Secondly, the voting that is defined within their framework is taken over actual mechanisms proposed by the designer and is by majority, whereas our voting is taken explicitly over Principal objectives and does not specify a majority vote, which allows potentially addressing issues such tyranny of the masses in future work. A more general game environment is illustrated in the AI Economist (Zheng et al., 2022), although they make strong assumptions on the goal of the Principal is and do not allow participants to adjust it.

6.2. Stackelberg Game

From the perspective of the Principal, it plays a Stackelberg game with agents of different types. Stackelberg games model many real-world problems that exhibit a hierarchical order of play by different players, including taxation (Zheng et al., 2022), security games (Jiang et al., 2013; Gan et al., 2020), and commercial decision-making (Naghizadeh & Liu, 2014; Zhang et al., 2016; Aussel et al., 2020). In the simplest case, a Stackelberg game contains one leader and one follower. For these games with discrete action spaces, Conitzer & Sandholm (2006) show that linear programming approaches can obtain Stackelberg equilibria in polynomial time in terms of the pure strategy space of the leader and follower. To find Stackelberg equilibria in continuous action spaces, Jin et al. (2020); Fiez et al. (2020) propose the notion of local Stackelberg equilibria and characterize them using first- and second-order conditions. Moreover, Jin et al. (2020) show that common gradient descent-ascent approaches can converge to local Stackelberg equilibria (except for some degenerate points) if the learning rate of the leader is much smaller than that of the follower. Fiez et al. (2020) give update rules with convergence guarantees. Different from these works, in this paper, we consider Stackelberg games with multiple followers.

More sophisticated than its single-follower counterpart, unless the followers are independent (Calvete & Galé, 2007), computing Stackelberg equilibria with multiple followers becomes NP-hard even when assuming equilibria with a special structure for the followers (Basilico et al., 2017). Recently, Wang et al. (2021a) propose to deal with an arbitrary equilibrium which can be reached by the follower via differentiating though it. Gerstgrasser & Parkes (2023) proposes a meta-learning framework among different policies of followers to enable fast adaption of the principal, which builds upon prior work done by Brero et al. (2022) who first introduced the Stackelberg-POMDP framework.

Multi-agent reinforcement learning holds the promise to

extend Stackelberg learning to more general and realistic problems. Tharakunnel & Bhattacharyya (2007) propose Leader-Follower Semi-Markov Decision Process to model the sequential Stackelberg learning problem. Cheng et al. (2017) propose Stackelberg Q-learning but without any convergence guarantee. Shu & Tian (2019); Shi et al. (2019) study leader-follower problems from an empirical perspective, where the leader learns deep models to predict the followers' behavior.

6.3. Multi-Agent Reinforcement Learning

Another important component of the proposed framework is the followers' behavior learning. Deep multi-agent reinforcement learning algorithms have witnessed significant advances in recent years. COMA (Foerster et al., 2018), MADDPG (Lowe et al., 2017), PR2 (Wen et al., 2019b), and DOP (Wang et al., 2021c) study the problem of policy-based multi-agent reinforcement learning. They use a (decomposed) centralized critic to calculate gradients for decentralized actors. Value-based algorithms decompose the joint value function into individual utility functions in order to enable efficient optimization and decentralized execution. VDN (Sunehag et al., 2018), QMIX (Rashid et al., 2018), and QTRAN (Son et al., 2019) progressively expand the representation capabilities of the mixing network. QPLEX (Wang et al., 2020a) implements the full IGM class (Son et al., 2019) by encoding the IGM principle into a duplex dueling network architecture. Weighted QMIX (Rashid et al., 2020) proposes weighted projection to decompose any joint action-value functions. There are other works that investigate into MARL from the perspective of coordination graphs (Guestrin et al., 2002b;a; Böhmer et al., 2020) and communication (Singh et al., 2019; Mao et al., 2020). These techniques are all relevant for usage in participant learning, although they do not explicitly consider behavior modelling of humans.

For modeling behavior, role-based learning frameworks (Wang et al., 2020b; 2021b) are the most related to our work. They learn the types (or the roles) of different agents from scratch based on the shared reward signal. The objective is to automatically decompose the task and reduce the learning complexity by learning sub-task specific polices. However, these works are majorly studied in the setting of the Decentralized Partially Observable Markov Decision Process (Dec-POMDP), and are thus different from our work by two points: (1) The reward is shared among agents; and (2) The dynamics, including reward and transition dynamics, are static in these models. There likely would exist significant challenges in generalizing these to non-shared reward settings that are essential for many economic applications.

6.4. Computational Social Choice

Computational social choice is an interdisciplinary field combining computer science and social choice theory, focusing on the application of computational techniques to social choice mechanisms (such as voting rules or fair allocation procedures) and the theoretical analysis of these mechanisms with computational tools (Brandt et al., 2016). A fundamental component of the field is the study of manipulative behavior in elections and other collective decision-making processes, as well as the design of systems resistant to manipulation (Elkind et al., 2010; Procaccia, 2010). This area of study will likely inform the development of the Voting Mechanism, and thus merits much consideration in our framework. Additionally, computational social choice attempts to optimize the fair distribution of resources, often involving complex allocation problems (Thomson, 2016; Procaccia, 2016), another area for drawing inspiration from for development of human baselines to compare against the Principal. However, our work also differs significantly in considering individual values within elections. Computational social choice typically assumes a discrete set of alternatives. This requires voters to express their values through support of a candidate that shares similar values. On the other hand, our framework enables voters to directly report their values in a continuous type space θ. This allows the voters to more precisely express values, without having to rely on a discrete set of candidates or policies who may not be exactly aligned with their personal θ.

6.5. Automated Mechanism Design

Automated Mechanism Design has a rich history somewhat similar in motivation to ours, and was first introduced by Sandholm (2003), where search algorithms are used to computationally create specific rule sets (mechanisms) for games that lead to desirable outcomes even when participants act in self-interest. More recently, the work of automated mechanism design has been advanced through deep learning, in the framework known as *differentiable economics*. Dütting et al. (Forthcoming 2023) use deep neural network to learn the allocation and payment rules of auctions. Since then, a line of follow-up work has been introduced, extending the framework to make the architecture more powerful and general (Shen et al., 2021; Ivanov et al., 2022; Duan et al., 2023; Curry et al., 2022; Wang et al., 2023). While these techniques are applied to settings much less general than ours, architectural details may be useful in building a Principal.

7. Conclusion

In this paper, we present a theoretical framework for both policy design and simulation that merges economic policy design with AI to potentially help better inform economic policy-making. It tackles issues such as preference

aggregation and counterfactual testing in complex economic systems. Significant challenges, including democratic representation and accountability in AI-driven systems, are highlighted. We hope to engage interdisciplinary expertise and foster collaborative innovation, and aspire to help create AI systems that not only enhance economic resilience and governance effectiveness but also uphold democratic ideals and ethical standards.

8. Impact Statement

This paper puts forth the position that Social Environment Design should be studied further, and that AI holds promise in improving policy design by proposing a general framework that can simulate general socioeconomic phenomena and scale to large settings. While we strongly believe in this research direction, there are several associated considerations to be aware of. Despite its power, this model is not a panacea, and its effectiveness depends on capturing all pertinent stakeholders within a given scenario. It is also important to ensure agent modeling is consistent with the diverse motivations and incentives that exist in the real world. Lastly, any real-world trial of this initiative should engage vigorously and faithfully with non-technical stakeholders.

References

Agapiou, J. P., Vezhnevets, A. S., Duéñez-Guzmán, E. A., Matyas, J., Mao, Y., Sunehag, P., Köster, R., Madhushani, U., Kopparapu, K., Comanescu, R., et al. Melting pot 2.0. *arXiv preprint arXiv:2211.13746*, 2022.

Arrow, K. J. *Social Choice and Individual Values*. Yale University Press, 2012. ISBN 978-0-300-17931-6. URL https://www.jstor.org/stable/j.ctt1nqb90.

Aussel, D., Brotcorne, L., Lepaul, S., and von Niederhäusern, L. A trilevel model for best response in energy demand-side management. *European Journal of Operational Research*, 281(2):299–315, 2020.

Balaguer, J., Koster, R., Weinstein, A., Campbell-Gillingham, L., Summerfield, C., Botvinick, M., and Tacchetti, A. Hcmd-zero: Learning value aligned mechanisms from data. *arXiv preprint arXiv:2202.10122*, 2022.

Barocas, S., Hardt, M., and Narayanan, A. *Fairness and Machine Learning: Limitations and Opportunities*. MIT Press, 2023.

Basilico, N., Coniglio, S., and Gatti, N. Methods for finding leader–follower equilibria with multiple followers. *arXiv preprint arXiv:1707.02174*, 2017.

Böhmer, W., Kurin, V., and Whiteson, S. Deep coordination graphs. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

Brandt, F., Conitzer, V., Endriss, U., Lang, J., and Procaccia, A. D. *Handbook of computational social choice*. Cambridge University Press, 2016.

Brero, G., Eden, A., Chakrabarti, D., Gerstgrasser, M., Li, V., and Parkes, D. C. Learning stackelberg equilibria and applications to economic design games. *arXiv preprint arXiv:2210.03852*, 2022.

Calvete, H. I. and Galé, C. Linear bilevel multi-follower programming with independent followers. *Journal of Global Optimization*, 39(3):409–417, 2007.

Cheng, C., Zhu, Z., Xin, B., and Chen, C. A multi-agent reinforcement learning algorithm based on stackelberg game. In *2017 6th Data Driven Control and Learning Systems (DDCLS)*, pp. 727–732. IEEE, 2017.

Conitzer, V. and Sandholm, T. Computing the optimal strategy to commit to. In *Proceedings of the 7th ACM conference on Electronic commerce*, pp. 82–90, 2006.

Curry, M., Trott, A., Phade, S., Bai, Y., and Zheng, S. Learning solutions in large economic networks using deep multi-agent reinforcement learning. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '23, pp. 2760–2762, Richland, SC, 2023. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450394321.

Curry, M. J., Lyi, U., Goldstein, T., and Dickerson, J. P. Learning revenue-maximizing auctions with differentiable matching. In *International Conference on Artificial Intelligence and Statistics*, pp. 6062–6073. PMLR, 2022.

de Figueiredo, J. M. and Richter, B. K. Advancing the Empirical Research on Lobbying. *Annual Review of Political Science*, 17(1):163–185, 2014. doi: 10.1146/annurev-polisci-100711-135308. URL https://doi.org/10.1146/annurev-polisci-100711-135308. eprint: https://doi.org/10.1146/annurev-polisci-100711-135308.

Duan, Z., Sun, H., Chen, Y., and Deng, X. A scalable neural network for dsic affine maximizer auction design. *arXiv preprint arXiv:2305.12162*, 2023.

Dütting, P., Feng, Z., Narasimhan, H., Parkes, D. C., and Ravindranath, S. S. Optimal auctions through deep learning: Advances in differentiable economics. *Journal of the ACM*, Forthcoming 2023. First version, ICML 2019, pages 1706–1715. PMLR, 2019.

Elkind, E., Faliszewski, P., and Slinko, A. Good Rationalizations of Voting Rules. In *Proceedings of the National Conference on Artificial Intelligence*, volume 2, September 2010.

Engstrom, D. F., Ho, D. E., Sharkey, C. M., and Cuéllar, M.-F. Government by Algorithm: Artificial Intelligence in Federal Administrative Agencies, February 2020. URL https://papers.ssrn.com/abstract=3551505.

Fiez, T., Chasnov, B., and Ratliff, L. Implicit learning dynamics in stackelberg games: Equilibria characterization, convergence analysis, and empirical study. In *International Conference on Machine Learning*, pp. 3133–3144. PMLR, 2020.

Foerster, J. N., Farquhar, G., Afouras, T., Nardelli, N., and Whiteson, S. Counterfactual multi-agent policy gradients. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Gan, J., Elkind, E., Kraus, S., and Wooldridge, M. Mechanism design for defense coordination in security games. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 402–410, 2020.

Gerstgrasser, M. and Parkes, D. C. Oracles & followers: Stackelberg equilibria in deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 11213–11236. PMLR, 2023.

Guestrin, C., Koller, D., and Parr, R. Multiagent planning with factored mdps. In *Advances in neural information processing systems*, pp. 1523–1530, 2002a.

Guestrin, C., Lagoudakis, M., and Parr, R. Coordinated reinforcement learning. In *ICML*, volume 2, pp. 227–234. Citeseer, 2002b.

Hanson, R. Shall we vote on values, but bet on beliefs? *Journal of Political Philosophy*, 21(2):151–178, 2013.

Heywood, A. *Political Ideologies: An Introduction*. Macmillan, 1998. ISBN 9780333698877. URL https://books.google.com/books?id=slulQgAACAAJ.

House, T. W. Executive Order on the Safe, Secure, and Trustworthy Development and Use of Artificial Intelligence, October 2023.

Ivanov, D., Safiulin, I., Filippov, I., and Balabaeva, K. Optimal-er auctions through attention. *Advances in Neural Information Processing Systems*, 35:34734–34747, 2022.

Jiang, A. X., Procaccia, A. D., Qian, Y., Shah, N., and Tambe, M. Defender (mis) coordination in security games. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

Jin, C., Netrapalli, P., and Jordan, M. What is local optimality in nonconvex-nonconcave minimax optimization? In *International Conference on Machine Learning*, pp. 4880–4889. PMLR, 2020.

Koster, R., Balaguer, J., Tacchetti, A., Weinstein, A., Zhu, T., Hauser, O., Williams, D., Campbell-Gillingham, L., Thacker, P., Botvinick, M., et al. Human-centred Mechanism Design with Democratic AI. *Nature Human Behaviour*, 6(10):1398–1407, 2022.

Liu, Z., Lu, M., Wang, Z., Jordan, M., and Yang, Z. Welfare maximization in competitive equilibrium: Reinforcement learning for Markov exchange economy. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 13870–13911. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/liu22l.html.

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, O. P., and Mordatch, I. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing Systems*, pp. 6379–6390, 2017.

Mao, H., Liu, W., Hao, J., Luo, J., Li, D., Zhang, Z., Wang, J., and Xiao, Z. Neighborhood cognition consistent multi-agent reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7219–7226, 2020.

Naghizadeh, P. and Liu, M. Voluntary participation in cyber-insurance markets. In *Workshop on the Economics of Information Security (WEIS)*, 2014.

Parkes, D. C. and Wellman, M. P. Economic reasoning and artificial intelligence. *Science*, 349(6245): 267–272, 2015. doi: 10.1126/science.aaa8403. URL https://www.science.org/doi/abs/10.1126/science.aaa8403.

Patig, S. Measuring expressiveness in conceptual modeling. In *International Conference on Advanced Information Systems Engineering*, 2004. URL https://api.semanticscholar.org/CorpusID:9715547.

Perolat, J., Leibo, J. Z., Zambaldi, V., Beattie, C., Tuyls, K., and Graepel, T. A multi-agent reinforcement learning model of common-pool resource appropriation, 2017.

Procaccia, A. D. Can approximation circumvent gibbard-satterthwaite? In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI'10, pp. 836–841. AAAI Press, 2010.

Procaccia, A. D. Cake cutting algorithms. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., and Procaccia, A. D. (eds.), *Handbook of Computational Social Choice*, pp. 311–330. Cambridge University Press, 2016. doi: 10.1017/CBO9781107446984.014.

Rashid, T., Samvelyan, M., Witt, C. S., Farquhar, G., Foerster, J., and Whiteson, S. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 4292–4301, 2018.

Rashid, T., Farquhar, G., Peng, B., and Whiteson, S. Weighted qmix: Expanding monotonic value function factorisation for deep multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 33, 2020.

Sandholm, T. Automated Mechanism Design: A New Application Area for Search Algorithms. In Rossi, F. (ed.), *Principles and Practice of Constraint Programming – CP 2003*, Lecture Notes in Computer Science, pp. 19–36, Berlin, Heidelberg, 2003. Springer. ISBN 978-3-540-45193-8. doi: 10.1007/978-3-540-45193-8_2.

Schulman, J., Moritz, P., Levine, S., Jordan, M., and Abbeel, P. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Shen, W., Tang, P., and Zuo, S. Automated Mechanism Design via Neural Networks, May 2021. URL http://arxiv.org/abs/1805.03382. arXiv:1805.03382 [cs].

Shi, Z., Yu, R., Wang, X., Wang, R., Zhang, Y., Lai, H., and An, B. Learning expensive coordination: An event-based deep rl approach. In *International Conference on Learning Representations*, 2019.

Shu, T. and Tian, Y. M^3RL: Mind-aware multi-agent management reinforcement learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

Singh, A., Jain, T., and Sukhbaatar, S. Learning when to communicate at scale in multiagent cooperative and competitive tasks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

Son, K., Kim, D., Kang, W. J., Hostallero, D. E., and Yi, Y. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 5887–5896, 2019.

Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W. M., Zambaldi, V., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J. Z., Tuyls, K., et al. Value-decomposition networks for cooperative multi-agent learning based on team reward.

In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 2085–2087. International Foundation for Autonomous Agents and Multiagent Systems, 2018.

Tharakunnel, K. and Bhattacharyya, S. Leader-follower semi-markov decision problems: theoretical framework and approximate solution. In *2007 IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning*, pp. 111–118. IEEE, 2007.

Thomson, W. *Introduction to the Theory of Fair Allocation*, pp. 261–283. Cambridge University Press, 2016. doi: 10.1017/CBO9781107446984.012.

Wang, J., Ren, Z., Liu, T., Yu, Y., and Zhang, C. Qplex: Duplex dueling multi-agent q-learning. *arXiv preprint arXiv:2008.01062*, 2020a.

Wang, K., Xu, L., Perrault, A., Reiter, M. K., and Tambe, M. Coordinating followers to reach better equilibria: End-to-end gradient descent for stackelberg games. *arXiv preprint arXiv:2106.03278*, 2021a.

Wang, T., Dong, H., Lesser, V., and Zhang, C. ROMA: Multi-agent reinforcement learning with emergent roles. In *Proceedings of the 37th International Conference on Machine Learning*, 2020b.

Wang, T., Gupta, T., Mahajan, A., Peng, B., Whiteson, S., and Zhang, C. RODE: Learning roles to decompose multi-agent tasks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021b.

Wang, T., Dütting, P., Ivanov, D., Talgam-Cohen, I., and Parkes, D. C. Deep contract design via discontinuous piecewise affine neural networks. *arXiv preprint arXiv:2307.02318*, 2023.

Wang, Y., Han, B., Wang, T., Dong, H., and Zhang, C. Dop: Off-policy multi-agent decomposed policy gradients. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021c.

Wen, Y., Yang, Y., Luo, R., and Wang, J. Modelling bounded rationality in multi-agent interactions by generalized recursive reasoning. *arXiv preprint arXiv:1901.09216*, 2019a.

Wen, Y., Yang, Y., Luo, R., Wang, J., and Pan, W. Probabilistic recursive reasoning for multi-agent reinforcement learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019b.

Yang, J., Wang, E., Trivedi, R., Zhao, T., and Zha, H. Adaptive incentive design with multi-agent meta-gradient reinforcement learning. *arXiv preprint arXiv:2112.10859*, 2021.

Zhang, H., Xiao, Y., Cai, L. X., Niyato, D., Song, L., and Han, Z. A multi-leader multi-follower stackelberg game for resource management in lte unlicensed. *IEEE Transactions on Wireless Communications*, 16(1):348–361, 2016.

Zheng, S., Trott, A., Srinivasa, S., Parkes, D. C., and Socher, R. The AI Economist: Taxation policy design via two-level deep multiagent reinforcement learning, 2022. URL https://www.science.org/doi/abs/10.1126/sciadv.abk2607.

A. Environment Hyperparameters and Training Details

Here we give a detailed breakdown of several key hyperparameters and Training Details within our environment in section 4.

We use PPO (Schulman et al., 2017) player agents with parameter sharing and GAE (Schulman et al., 2015), collecting samples at a horizon shorter than the episode length to perform multiple policy update iterations per episode. The principal has separate, discrete, action subspaces for each tax bracket, and is also trained by standard PPO at the same time-scale as the player agents. We follow a two-phase curriculum with tax annealing, as suggested in Zheng et al. (2022). This annealing can be formalized as a constraint in the policy implementation map by simply bounding the maximum tax percentage that can be set. It is worth noting, however, that training the principal in this way is susceptible to issues of non-stationarity, and we refer to Yang et al. (2021) for a discussion on alternatives.

To give a further explanation regarding the Apple Respawn Probabilities, the probability of a respawn per timestep depends on how many neighbors are around it in a circular radius of 2. With four neighbors, the respawn probability is 0.025. With 0, the probability becomes 0.

Hyperparameter	Value
Number of Agents	7
Initial Number of Apples	64
Apple Respawn Probabilities	[0.025, 0.005, 0.0025, 0.0]
Base Reward	1 on apple collection
Social Reward	1 on apple collection of observable agents
Agent Type (σ, β)	Sampled from Uniform $[0, 1]$
Agent Observability (units are grid tiles)	(Forward: 9, Right: 5, Backward: 1, Left: 5)
Principal Tax Brackets (units are in apples)	[(1,10),(11,20),(21,10000)]
Tax Period	50
Episode Length	1000
Sampling Horizon	200

Table 1: Hyperparameters for our methods in section 4.