Simulating Tax Policy Lifecycles: Agent Utility, Elections, and the Economic Dynamics of Labor and Taxation with LLM Generative Agents

BACKGROUND

Accurately evaluating tax policies historically requires large-scale, real-world experimentation that is politically and financially challenging. Traditional economic models simplify human behavior, limiting their real-world accuracy while machine learning approaches demand significant resources to fine tune models for unique communities. To address these challenges, innovative and scalable methods are needed for simulating societal behavior and optimizing policies. This research explores leveraging large language models (LLMs) to generate synthetic human data, enabling affordable, generalizable policy testing and design. By modeling tax policy decisions as a dynamic game between the government and residents, we aim to optimize social welfare, create a system for tax policy generation and testing, and lay the groundwork for applications in other policy areas.

METHODS

Voting Simulation - Similar to a classic RL loop

1. **Choose** parameters and initialize agents
2. Agents **act** by prompting an LLM with relevant state history and the current prompt to generate their vote for tax planner if it is time for an election, a new tax policy if they are the current tax planner and it is time for a new policy, and their choice of labor
3. **Update** all agents utility

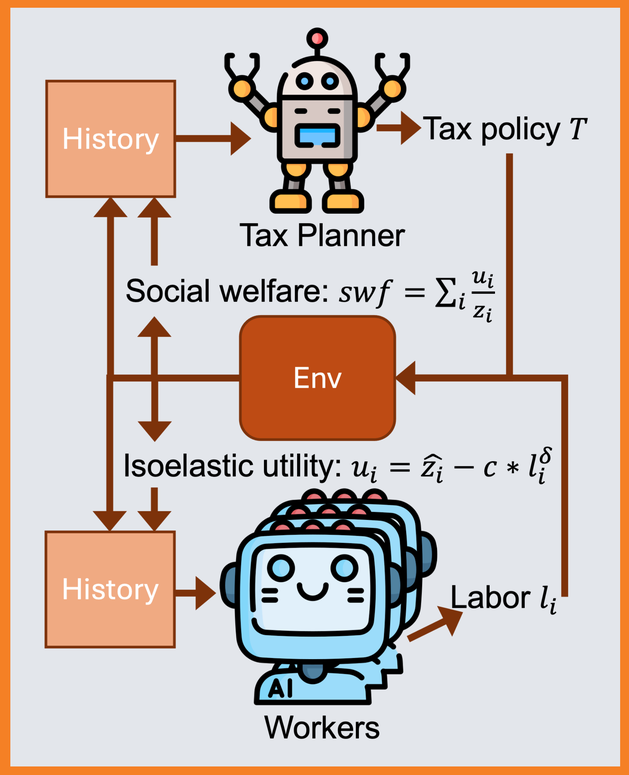
RESULTS

- Worked through ablations to confirm the parameters needed for optimal solution. The parameters tested include timesteps, history context length, and fixing different combinations of agents (shown under Ablations) to see if the LLM could solve a simpler optimization problem.
- Compared results to a tax planner that followed the Saez economic model with the goal of optimizing social welfare
- Programmed a new simulation with a voting feature where all LLM agents work, and the tax planner is elected from among all worker agents. The new problem is a 3 timescale optimization problem. The assumptions necessary for Saez's model to apply are broken

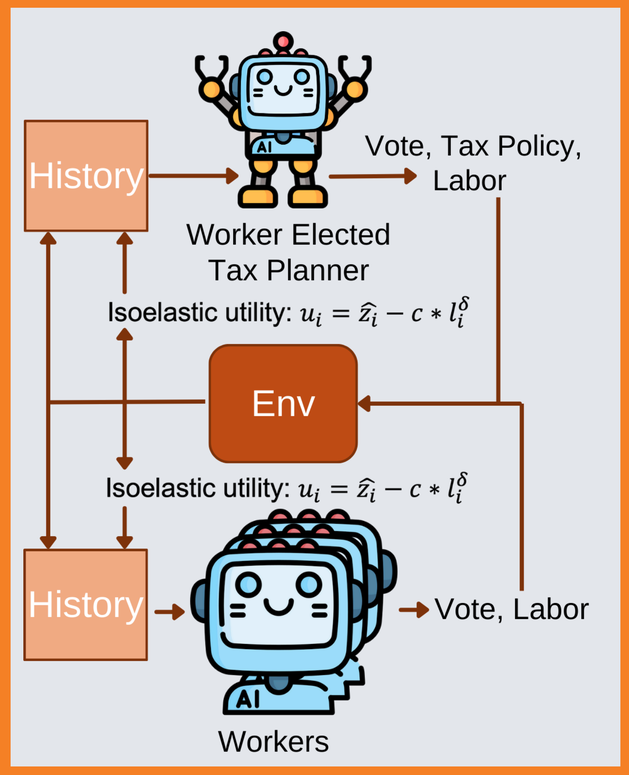
[1] Karten, Seth, et al. The LLM Economist: Optimizing Policy in Multiagent Generative Simulations.

Governments need innovative simulation and modeling techniques to evaluate policy impacts before deployment to make policy experimentation feasible.

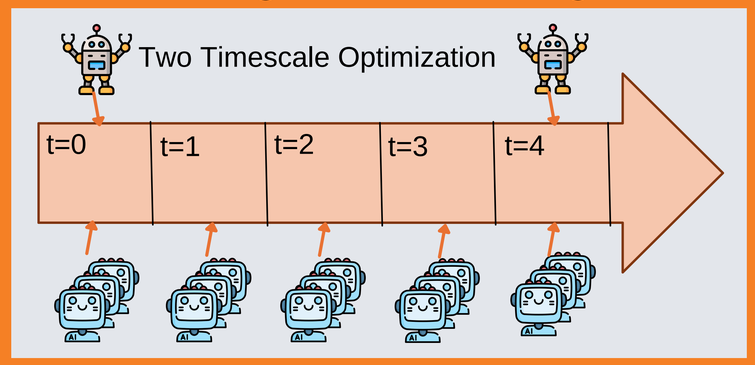
Methodology 1[1]



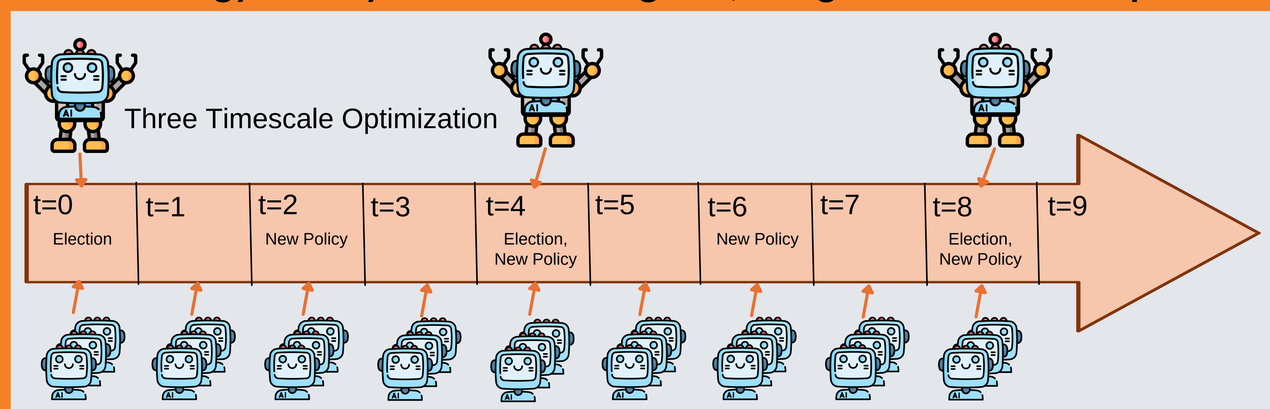
Methodology 2



Methodology 1: Tax Planner agent and Worker agents are different classes



Methodology 2: Only Worker class agents, an agent is elected tax planner

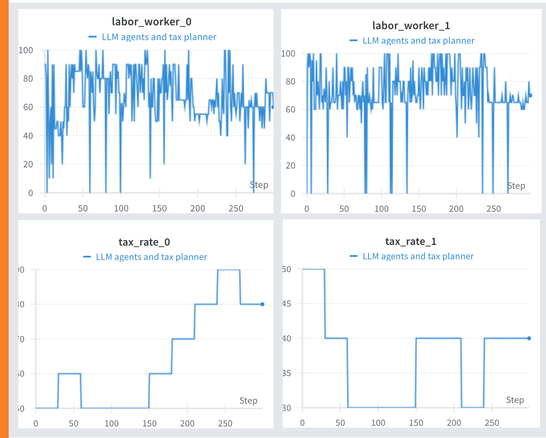


Ablations

- One LLM Worker, Fixed Tax Planner
- One LLM Worker, LLM Tax Planner
- Fixed Workers, LLM Tax Planner
- LLM Workers, Fixed Tax Planner
- LLM Workers, LLM Tax Planner

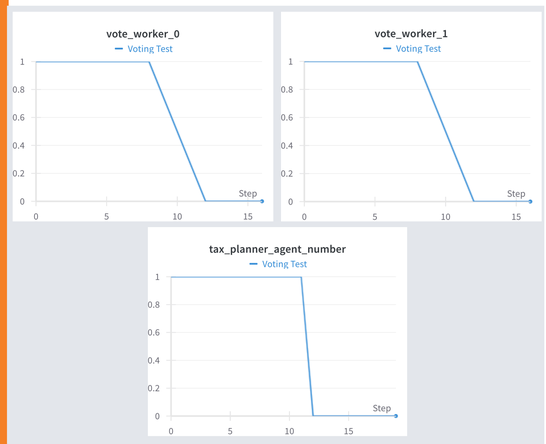
Saez's optimal tax rates are [100, 0], and optimal labor is 60 for both workers.

LLM Workers, LLM Tax Planner Trial



Voting Simulation Proof of Concept

These graphs show the position of tax planner is changing from worker_1 to worker_0 because of the results of an election. More prompt engineering needs to be done to make the LLM output a valid JSON string, so the LLM's outputs were overwritten to test the program's logic.



Coming Soon

- Prompt engineering to make llama output valid JSON strings
- Testing different LLMs

Samuel Kleiner

ECE