The AI Macroeconomy: A New Set of Benchmarks for Multiagent Reinforcement Learning Models

Anonymous Authors[1]

Abstract

We propose a new set of challenging benchmark gym environments for testing single- and multi-agent reinforcement learning environments. Single-agent environments are based on a simple consumption-saving decision problem. In each period, agents face an exogenous positive draw that represents how much income they will have in this period. In response, agents may choose what fraction of that income they would like to consume immediately for a reward, or save and get a return going forward on it. In the full version of the problem, all agents' saving decisions generate a price via market clearing. Agents then must learn what their value will be conditioned on the current state. This environment will provide a challenging, potentially nonstationary environment where agents' actions have critical effects on other agents' actions, albeit via a common observation. This environment will be made publicly available via a Github repository and open-source.

1. Introduction

Understanding how agents arrive at game-theoretic equilibria is a critical part of the current multiagent reinforcement learning landscape. But often these environments are in game-style environments to provide a benchmark against human players. (Bard et al., 2020) (Zhang et al., 2021) Recently, there has been interest in extending these environments to economic and financial environments to understand how the trading behavior of agents in the economy arises, perhaps most notably the AI Economist series of papers and work from Deepmind exploring how bartering economies arise. (Zheng et al., 2020) (Zheng et al., 2021) (Johanson et al., 2022) However, so far these articles have been separated from historically relevant models to the economics community and from any kind of macroeconomic analysis.

We instead propose a classical set of economic problems called consumption-savings problems for the single-agent variant, and Aiyagari-Bewley-Huggett models for the multi-agent variant (traditionally a continuum of agents are present, rather than a discrete number as we have) with varying levels of difficulty to test current multi-agent RL environments. (Aiyagari, 1994). They are in wide use by central banks and academic macroeconomists to model the interaction of income inequality and economic shocks in the economy, with the recent trend started by Kaplan & Violante (2014) These environments have been computationally solved approximately via dynamic programming for the full-information environment. However, they have not been solved for the partially observed setting. We consider a variant in which agents face symmetric problems (in the baseline variant) but can only observe a common price that depends on their trading behavior, as well as any private state and action realizations. Normally, to solve these models nearly exactly with dynamic programming, one of the states to be tracked must be the distribution of all agent states and actions in the economy each period. This is a heavy burden to ask agents if we are thinking of modeling a market-style setting.

2. Description of the Single-Agent Problem

2.1. Baseline Environment

The environment of the single-agent problem is composed of a discrete-time Markov Decision Process which we denote $(\mathbb{R} \times \mathbb{R}^+ \times \mathbb{R}^+, \mathbb{R}, \mathcal{T}, R)$. The state space for agent i is composed of three states: the savings of this period, which we call assets and denote a_t (and which when negative, we think of as borrowing), the current interest rate, which we denote r_t, and the current income realization of the agent, which we denote y_t.

The action space for agents consists of a consumption choice which we denote c_t of consumption goods. The transition operator will be made clear shortly. Finally, the reward function R can be any smooth function of consumption. Typically, for benchmarking, a quadratic is chosen so that we know exactly what the optimal unconstrained decision

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

of the agent would be.

The agent's objective is to maximize a discounted sum of rewards over a horizon T:

$$\max_{\{c_t\}_{t=0}^{T}} \mathbb{E}_0 \sum_{t=0}^{\infty} \gamma^t R(c_t) \tag{1}$$

This is subject to a budget constraint that represents how much an agent can consume each period. This is:

$$\begin{aligned} c_t + a_{t+1} &\leq (1+r_t)a_t + y_t \quad \forall t \in \{1, \ldots, T\} \\ y_t &\sim F, \end{aligned} \tag{2}$$

where F is some distribution function. Typically r_t is taken to be constant, and this is what we do in our baseline problem $r_t = r$.

In our baseline environment, we will always treat R as if it is a quadratic.

Thus our problem in our baseline environment mathematically can be written as the dynamic programming problem:

$$\begin{aligned} Q(a, y, c) &= a_0 + a_1(c - \bar{c})^2 \\ &\quad + \gamma \max_{c'} \mathbb{E}_{y' \sim F}[Q(a', y', c') \mid a, y, c] \\ s.t &\quad c + a' \leq (1+r)a + y \end{aligned} \tag{3}$$

Our agents will not know the distribution F and hence the optimal state action value function $Q(a, y, c)$ or its expected value. This is what they must learn about to solve the problem. If agents had complete knowledge of the problem, the optimal solution would always be to consume $c' = \bar{c}$.

2.2. Modified Environment

Now we introduce a borrowing constraint, $\underline{a} \in \mathbb{R}$, a non-linearity into the problem. Agents cannot borrow more than $\bar{a}$. The new environment is as follows:

$$\begin{aligned} Q(a, y, c) &= a_0 + a_1(c - \bar{c})^2 \\ &\quad + \gamma \max_{c'} \mathbb{E}_{y' \sim F}[Q(a', y', c') \mid a, y, c] \\ s.t &\quad c + a' \leq (1+r)a + y \\ &\quad a' \geq \underline{a} \end{aligned} \tag{4}$$

This new nonlinearity means that the optimal policy is no longer linear $c' = \bar{c}$. If agents did this, they could end up hitting their borrowing limit and, therefore, might have to consume 0 in the future. Agents must now learn this optimal policy.

2.3. Modified Environment 2

We can make this non-linearity more severe by moving away from quadratic rewards. Instead, we consider a reward function $\log(c)$, where if agents ever consume 0 they receive $-\infty$. This introduces the need for caution during exploration to avoid hitting the borrowing constraint. This model is:

$$\begin{aligned} Q(a, y, c) &= \log(c) + \gamma \max_{c'} \mathbb{E}_{y' \sim F}[Q(a', y', c') \mid a, y, c] \\ s.t &\quad c + a' \leq (1+r)a + y \\ &\quad a' \geq \underline{a} \end{aligned} \tag{5}$$

3. Description of the Multi-Agent Environment

We now describe the multi-agent environment. There are N agents in the economy labeled 1 to N. The agents' individual problems are symmetric, conditional on a current state-action pair. However, unlike before the interest rate r is not fixed. Instead, it is endogenously determined. This requires the addition of an **aggregate state** that all agents can observe which will exactly be r itself. Let A denote the average level of assets in the economy across agents in any given time period.

$$A = \frac{1}{N} \sum_{i=1}^{N} a_i \tag{6}$$

Then the interest rate r this period is given by:

$$r = \alpha(A)^{\alpha - 1} - 1, \quad 0 < \alpha < 1 \tag{7}$$

The production function on the right-hand side is a special case of a so-called Cobb-Douglas production function, a well-known class of models for modeling goods producers in economics. This says that for a profit maximizing firm, the rental rate on assets, which is the interest rate, is exactly equal to the return it gets on its marginal unit of capital. Note that the interest rate is increasing (albeit potentially sublinearly) in the aggregate capital stock.

Now, we define three associated relevant information settings that represent different challenges agents can face in this environment.

3.1. Baseline Environment

First suppose the agent only observes their own action and the price. That is, agent i's problem looks like:

$$\begin{aligned} Q(a_i, y_i, r, c_i) &= R(c_i) \\ &\quad + \gamma \max_{c'} \mathbb{E}_{y' \sim F, r'}[Q'(a_i', y_i', r', c_i') \mid a_i, y_i, r, c_i] \\ s.t &\quad c_i + a_i' \leq (1+r)a_i + y_i \\ &\quad a_i' \geq \underline{a} \end{aligned} \tag{8}$$

for all agents i and with associated environment update Equation (7). How the price is generated is common knowledge to all players, although the actual choices of others are

not observed. Note that the appropriate equilibrium concept for this partially observed stochastic game will be a Nash Equilibrium. We define a policy mapping for agent i as a mixed strategy $\pi_i(a_i, y_i, r, c_i) = P(c_i' \mid a_i, y_i, r, c_i)$, where the chosen actions must satisfy the constraints Given an N-tuple of policies π^N, and an initial distribution of agents μ_0 define the true state-action value under the induced policy for agent i as

$$J_i^N(\pi^N) = \mathbb{E}^{\pi^N}\left[\sum_{t=0}^{\infty} \gamma^t R(c_{it})\right]. \tag{9}$$

Then, we say a N-vector of policies π^{N*} constitutes a Nash equilibrium if:

$$J_i^N(\pi^{N*}) = \sup_{\pi^i} J_i^N\left(\pi_{-i}^{N*}, \pi_i\right) \tag{10}$$

for each agent $i \in 1, \dots N$ and where $\pi_{-i}^{(N*)} := \left(\pi^{j*}\right)_{j \neq i}$. Our transition operator $\mathcal{T}$ for this economy is therefore the induced map given by simulating the economy following this N-vector of policies. For further details on the equilibrium, see Saldi et al. (2019). In practice, we have N separate environments, where agents all face identical prices, and choose their action each period. Then across environments, the choices of agents are aggregated up.

This environment is part of a special class of models as $N \to \infty$ called mean-field games.

3.2. Modified Environment

Now we consider an expanded environment where agents may fully observe the environment. Therefore, the actions of all other players become states. The problem of agent i then is to solve the game:

$$\begin{aligned}
&Q(a_i, y_i, r, \mathbf{a}_{-i}', a_i') \\
&= R(y_i + (1+r)a_i - a_i') \\
&+ \gamma \max_{a_i''} \mathbb{E}_{y' \sim F,} \left[Q\left(a_i', y_i', r', \mathbf{a}_{-i}'', a_i''\right) \mid a_i, y_i, r, \mathbf{a}_{-i}', a_i'\right] \\
&\qquad a_i' \geq \underline{a} \\
&\qquad A = \frac{1}{N}(a_i' + \sum_{j \neq i} a_j') \\
&\qquad r' = \alpha A^{\alpha - 1} \\
&\qquad y_i' \sim F,
\end{aligned}$$

where all we did is rewrite the choice from c_i to a_i' by using the budget constraint and assuming it held with equality.

Again, defining the Nash equilibrium exactly as before, except for our new state space, we will again find a collection of π^{N*} and hence $a^{N*'}$ such that given all other players are playing π_{-i}^{N*} and hence a_{-i}^{N*}, our state-action value function is maximized under π_i^{N*}.

3.3. Modified Environment 2

We can also explore what happens under imperfect information. For example, suppose that, in addition to choosing savings for tomorrow or consumption, agents can also choose whether or not to reveal their choices to others. What will happen? What about if they can offer it for a price? These represent economies with communication and information markets. Again, the equilibrium will be a Nash equilibrium, but again more complicated. [1]

4. Implementation

We now discuss the actual implementation of our environment. The entire environment will be implemented as an OpenAI Gym environment. The single agent environments are individual classes, while the multiagent environment will use the single-agent classes as a baseline, aggregate choices, ensure economy-wide constraints are satisfied, and then return prices back to each of the single-agent problems which could be run in a vectorized or parallel manner. The multiagent environment instantiates N copies of the single-agent environment within it.

The timing of the update process is as follows. Each period, a set of actions are fed into the multiagent environment. The multiagent environment then assigns these actions to each copy of the single-agent environment. Within each single agent environment, a unique seed generates a draw of y from F, and a state is recorded from the previous period or initialized. First, the action is checked to ensure that it is valid given the constraint imposed on the problem. If not, it is adjusted to the nearest feasible point on the real interval representing the constraint. Then, the reward is computed. Next, savings are generated that satisfy the agent's borrowing constraint if it is present. Finally, the period comes to a close, and the savings for next period and hence asset holdings for next period are computed. The assets are then updated, and a counter is checked to see if the length of an episode has been exceeded. If it has, then a done boolean is set to true.

Finally, the asset holdings, income draw, and the done Boolean are returned. These pass to the containing multiagent environment. The multi-agent environment computes r and then sets a local variable with this value for each of the single-agent classes for the next period. Finally, idiosyncratic states of income and asset holdings are merged with the interest rate to produce a N-vector of a triplet of states. These are then returned from the multi-agent environment on the conclusion of its step function so that they can be utilized in the learning process for agents in the economy.

[1] For an example of information markets in these settings see Broer et al. (2022)

5. A Discussion on Timing

Note that the timing in these multiagent models is different than in the traditional Aiyagari model. Normally, in equilibrium, it must be the case that agents account for prices when they make their consumption choices today. Here, they do not do that. As a result they are only responding to the forecasted price realization tomorrow given the price realization today. This is computationally easy to implement, but is somewhat removed from the original economics setting.

As an alternative, we can consider implementing prices contemporaneously with agents' choice of consumption. However, this requires some sort of explicit market clearing mechanism and repeatedly generating and offering price schedules for different savings levels to agents.

One possibility is to introduce a bookkeeping mechanism to ensure market clearing, where agents can submit bids for different levels of savings, and these bids determine prices. We leave it to future extensions to puzzle out a computationally feasible mechanism for contemporaneous prices.

6. Future Work

While the single agent models work and have been tested, the multiagent models still need to be verified. When this work is finished, the code will be made available on GitHub as part of a repository. Some examples will also be provided as first-best solutions to problems that arise from complete knowledge of optimal solutions without the presence of strategic deviations.

7. Conclusion

We have described a new set of environments that can be used to benchmark agent performance in single and multiagent settings. These models are widely used in economics, computationally simple, and shed light on interesting market economies. They are also easily able to be utilized as a baseline to extend to more complicated settings. By providing this set of models on GitHub in the future in the OpenAI gym environment, we hope to provide a challenging set of benchmarks and provide a template for how traditional models can be extended and thought about in multiagent settings.

References

Aiyagari, S. R. Uninsured Idiosyncratic Risk and Aggregate Saving. *The Quarterly Journal of Economics*, 109(3):659–684, 1994. ISSN 0033-5533. doi: 10.2307/2118417. URL https://www.jstor.org/stable/2118417. Publisher: Oxford University Press.

Bard, N., Foerster, J. N., Chandar, S., Burch, N., Lanctot, M., Song, H. F., Parisotto, E., Dumoulin, V., Moitra, S., Hughes, E., Dunning, I., Mourad, S., Larochelle, H., Bellemare, M. G., and Bowling, M. The Hanabi Challenge: A New Frontier for AI Research. *Artificial Intelligence*, 280:103216, March 2020. ISSN 00043702. doi: 10.1016/j.artint.2019.103216. URL http://arxiv.org/abs/1902.00506. arXiv:1902.00506 [cs, stat].

Broer, T., Kohlhas, A. N., Mitman, K. E. W., and Schlafmann, K. Information and Wealth Heterogeneity in the Macroeconomy. pp. 34, 2022.

Johanson, M. B., Hughes, E., Timbers, F., and Leibo, J. Z. Emergent Bartering Behaviour in Multi-Agent Reinforcement Learning. Technical Report arXiv:2205.06760, arXiv, May 2022. URL http://arxiv.org/abs/2205.06760. arXiv:2205.06760 [cs] type: article.

Kaplan, G. and Violante, G. L. A Model of the Consumption Response to Fiscal Stimulus Payments. *Econometrica*, 82(4):1199–1239, 2014. ISSN 1468-0262. doi: 10.3982/ECTA10528. URL https://onlinelibrary.wiley.com/doi/abs/10.3982/ECTA10528. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.3982/ECTA10528.

Saldi, N., Başar, T., and Raginsky, M. Approximate Nash Equilibria in Partially Observed Stochastic Games with Mean-Field Interactions. *Mathematics of Operations Research*, 44(3):1006–1033, August 2019. ISSN 0364-765X. doi: 10.1287/moor.2018.0957. URL https://pubsonline.informs.org/doi/10.1287/moor.2018.0957. Publisher: INFORMS.

Zhang, K., Yang, Z., and Başar, T. Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms. Technical Report arXiv:1911.10635, arXiv, April 2021. URL http://arxiv.org/abs/1911.10635. arXiv:1911.10635 [cs, stat] type: article.

Zheng, S., Trott, A., Srinivasa, S., Naik, N., Gruesbeck, M., Parkes, D. C., and Socher, R. The AI Economist: Improving Equality and Productivity with AI-Driven Tax Policies. Technical Report arXiv:2004.13332, arXiv, April 2020. URL http://arxiv.org/abs/2004.13332. arXiv:2004.13332 [cs, econ, q-fin, stat] type: article.

Zheng, S., Trott, A., Srinivasa, S., Parkes, D. C., and Socher, R. The AI Economist: Optimal Economic Policy Design via Two-level Deep Reinforcement Learning. Technical Report arXiv:2108.02755, arXiv, August 2021. URL http://arxiv.org/abs/2108.02755. arXiv:2108.02755 [cs, econ, q-fin] type: article.