

Semi-RL Autonomous Macro System: A Hybrid Approach for Reinforcement Learning in Macroeconomics

1. Introduction

Reinforcement learning (RL) has recently been explored as a tool for macroeconomic modeling, but current approaches fall into two broad categories, each with notable limitations. (1) Single-agent RL with a representative agent: Here a single RL "agent" represents an entire sector (e.g. the representative household or firm) interacting with an abstract macroeconomic environment. This approach simplifies the economy but often requires very restrictive assumptions. It struggles to handle complex lifetime utility maximization with expectations, so researchers are forced to use overly simplified representative agents.

Moreover, because most of the economy is baked into the fixed environment dynamics, these models lack rich agent dynamics – making it difficult to capture how policy changes propagate over time through agent behavior. (2) Multi-agent RL with many agents: In this approach, multiple agents (households, firms, government, etc.) are each modeled as RL agents. While this allows heterogeneity and potentially richer interactions, multi-agent RL systems are notoriously hard to train. They often suffer from instability and have no guarantee of converging to a stable equilibrium as the number of agents grows jmlr.org. This instability severely limits scalability. Yet, to study crucial macro questions like income inequality or optimal taxation, we need to simulate a large population of heterogeneous agents – something that neither single-agent nor naive multi-agent RL handles well under current methods.

To address these challenges, we propose a new framework called the Semi-RL Autonomous Macro System (SAMS). This hybrid approach combines the strengths of traditional economic modeling with modern RL and deep learning techniques. The key idea is to let the majority of agents in the model behave according to established economic principles (following their first-order conditions, or F.O.C., for optimality) while reserving RL for the elements of the system that we want to learn or optimize (such as a policy-making agent). In other words, instead of making every agent an RL learner (which causes learning instability) or reducing the whole economy to one trivial RL agent (which oversimplifies dynamics), we split the work: non-RL agents follow adaptive decision rules derived from economic theory, and a limited number of RL agents learn optimal policies within this environment. The non-RL agents are not static; they behave autonomously by optimizing their objectives (e.g. utility or profit) via their F.O.C., which means they respond realistically to changes in the economic environment. This creates a dynamic and responsive macroeconomic simulator, but without the explosion of learning complexity that plagues full MARL systems.

Technically, our framework leverages a Monte Carlo simulation-based deep learning method to solve these agents' optimization problems. Recent advances in deep learning have shown that we can efficiently solve high-dimensional dynamic models (such as heterogeneous-agent macro models) by casting them as neural-network approximations to Bellman or Euler equations web.stanford.edu. In particular, we build on the method of Maliar et al. (2021), who demonstrated a deep learning approach capable of handling large-scale heterogeneous-agent models (e.g. the Krusell–Smith model with thousands of agents) without resorting to the usual simplifying assumptions web.stanford.edu. By incorporating this deep learning Monte Carlo solver, SAMS frees us from having to assume a single representative agent. We can include a rich

variety of heterogeneous agents whose decision rules are computed through the deep learning framework, ensuring that each agent (or type of agent) optimally responds to its economic incentives and constraints. These agents effectively simulate the rational behavior one would expect in a macroeconomic model, but with computational tractability even in large numbers.

Meanwhile, the elements of the system that do use RL (for example, a policy agent like a government adjusting tax rates) can learn and adapt their strategies over time. Because the bulk of the economy (households, firms, etc.) is handled by the deep-learning-based F.O.C. solvers, the overall simulation remains much more stable. The environment that the RL agent interacts with is grounded in equilibrium behavior (through the non-RL agents), which is less non-stationary than an all-RL multi-agent system. As a result, SAMS can scale to a large number of heterogeneous agents without suffering the divergence or chaos that typically occurs in multi-agent RL. This hybrid design significantly improves stability and realism: policy effects can be observed through the adaptive responses of many simulated agents, yet the training process remains feasible.

In summary, the Semi-RL Autonomous Macro System offers a balanced approach that bridges the gap between existing RL-in-macro methods. It retains the rich dynamics and heterogeneity of agent-based models while constraining the learning problem to a manageable scope. By focusing RL where it is most needed (e.g. learning optimal policy) and using deep learning to handle traditional agent optimization, this framework aims to enable robust macroeconomic simulations for scenarios like inequality, taxation, and beyond. We anticipate that SAMS can unlock more realistic and scalable macroeconomic policy analysis, combining the strengths of reinforcement learning with the rigor of dynamic economic modeling.

2. Model and Methodology

2.1 Households

$$\begin{aligned} & \max_{\{c_t, h_t, a_{t+1}\}_{t=0}^{T_N}} \quad & \mathbb{E}_0 \sum_{t=0}^{T_N} \beta^t \left(\frac{c_t^{1-\theta}}{1-\theta} - \frac{h_t^{1+\gamma}}{1+\gamma} \right) \tag{1} \\ & (1+\tau_s)c_t + a_{t+1} = i_t - T(i_t) + a_t - T^a(a_t), \quad & m_t \geq i_t - T(i_t) + a_t - T^a(a_t) \quad \& \quad i_t = w_{te_th_t+r_{t-1}}a_t \quad \& \quad a_{t+1} \geq 0 \quad \text{iff} \quad (1+\tau_s)c_t \leq m_t \quad \text{iff} \quad c_t \leq \frac{m_t}{1+\tau_s} \\ & \end{aligned} \quad \text{Setting Lagrangian} \quad \begin{aligned} L = & \mathbb{E}_0 \sum \beta^t U(c_t, h_t) + \lambda_t [w_{te_th_t+r_{t-1}}a_t - T(i_t) + a_t - T^a(a_t) - a_{t+1} - (1+\tau_c)c_t] + \\ & \mu_{ta_{t+1}} \end{aligned}$$

$$\end{aligned}$$

$$[c_t]: \mathbb{E}_0[U'(c_t, h_t)] = \lambda_t(1+\tau_s) \tag{2}$$

$$\begin{aligned} [a_{t+1}]: \quad & \mathbb{E}_0[\beta^{t+1}(-\lambda_{t+1} + \mu_t + \beta^{t+1}(\lambda_{t+1}r_t - T'(i_{t+1})r_t + 1 - T^a(a_{t+1})))] = 0 \quad \& \quad \mathbb{E}_0[-\lambda_{t+1} + \mu_t + \beta^{t+1}(\lambda_{t+1}r_t - T'(i_{t+1})r_t + 1 - T^a(a_{t+1})))] = 0 \quad \& \quad \text{implies} \quad \mathbb{E}_0[\mu_t] = \mathbb{E}_0[\lambda_t - \lambda_{t+1}r_t - T'(i_{t+1})r_t + 1 - T^a(a_{t+1})] \end{aligned} \tag{3}$$

$$\begin{aligned} [h_t]: \quad & \mathbb{E}_0[\beta^t U'(c_t, h_t) + \lambda_t[w_{te_t} - T(i_t)]] = 0 \tag{4} \\ & \mathbb{E}_0[-h_t^{-(1+\gamma)} + \lambda_t[w_{te_t} - (w_{te_th_t+r_{t-1}}a_t - (1+\tau_c)c_t) \frac{w_{te_th_t+r_{t-1}}a_t}{1+\tau_c}]] = 0 \end{aligned}$$

Combining (2) and (3), we get the Euler equation for consumption and saving

$$\mu_t = \frac{\mathbb{E}_0[U'(c_t, h_t)]}{1+\tau_s} - \frac{\mathbb{E}_0[U'(c_{t+1}, h_{t+1})]}{1+\tau_s}$$

$$\{1 + \tau_s \{r_t - T'(i_{t+1})r_t + 1 - T'^a(a_{t+1})\}\} \tag{5}$$
 By KKT conditions: $\mu_t > 0$, $a_{t+1} > 0$, and $\mu_t a_{t+1} = 0$, applying the Fischer-Burmeister (FB) function $\psi^{\text{FB}}(x, y) = x + y - \sqrt{x^2 + y^2}$ and turning into unit-free form $x \leq 1 - \frac{\mathbb{E}_0 U'(c_{t+1}, h_{t+1})}{\mathbb{E}_0 U'(c_t, h_t)} \{r_t - T'(i_{t+1})r_t + 1 - T'^a(a_{t+1})\} \tag{6}$ $y \leq a_{t+1} \in [\underline{a}, \underline{m}_t], \quad \underline{a} > 0$

2.2 Government (RL Agent)