DeepHAM: A Global Solution Method for Heterogeneous Agent Models with Aggregate Shocks

Jieqiun Han[*], Yucheng Yang[†], and Weinan E[‡]

This version: January 2022
First version: December 2021

Abstract

We propose an efficient, reliable, and interpretable global solution method, *Deep learning-based algorithm for Heterogeneous Agent Models, DeepHAM*, for solving high dimensional heterogeneous agent models with aggregate shocks. The state distribution is approximately represented by a set of optimal generalized moments. Deep neural networks are used to approximate the value and policy functions, and the objective is optimized over directly simulated paths. Besides being an accurate global solver, this method has three additional features. First, it is computationally efficient for solving complex heterogeneous agent models, and it does not suffer from the curse of dimensionality. Second, it provides a general and interpretable representation of the distribution over individual states; and this is important for addressing the classical question of whether and how heterogeneity matters in macroeconomics. Third, it solves the constrained efficiency problem as easily as the competitive equilibrium, and this opens up new possibilities for studying optimal monetary and fiscal policies in heterogeneous agent models with aggregate shocks.

Keywords: Heterogeneous agent models, aggregate shocks, global solution, deep learning, generalized moments, constrained efficiency.

[*]Flatiron Institute and Princeton University. Email: jiequnhan@gmail.com.
[†]Princeton University. Email: yuchengy@princeton.edu.
[‡]Peking University and Princeton University.

We are grateful to Gianluca Violante for numerous discussions and constructive feedback throughout this project. We thank Mark Aguiar, Anmol Bhandari, Nobu Kiyotaki, Moritz Lenel, Ziang Li, Ben Moll, Jonathan Payne, Mikkel Plagborg-Møller, Chris Sims, Jeffrey Sun, Wei Xiong, and many seminar participants for helpful comments. All errors are our own.

1 Introduction

Incorporating both explicit heterogeneity and aggregate fluctuations into quantitative models has been one of the most important developments in macroeconomics in the recent decade. It has become an eminent agenda for a number of reasons. First, uneven dynamics across sectors and population groups after major economic fluctuations and policy shocks suggest that heterogeneity and aggregate shocks are necessary components in studying fundamental macroeconomic problems. Second, the recent development of the heterogeneous agent New Keynesian (HANK) models suggests that heterogeneity provides key channels for transmitting aggregate shocks, which is crucial for obtaining correct aggregate implications of monetary and fiscal policies. Third, the advance of computational methods and computing power allows economists to build more realistic models beyond the representative agent models that have long been dominant in both academic and policy research.

Despite the prominence of this agenda, its workhorse models, heterogeneous agent (HA) models with aggregate shocks, still present severe computational challenges. Ideally, we would like to develop solution methods that fulfill the following basic requirements:

- **Efficiency**: The method should be computationally efficient, especially for complex HA models with multiple state variables. This is necessary in order to use the method for calibration, estimation, and further quantitative analysis.

- **Reliability**: It should produce accurate solutions for all practical situations that HA models are intended for. In particular, it should be applicable beyond the local perturbation regime if nonlinear and nonlocal effects of aggregate shocks are important.

- **Interpretability**: We are not just interested in the numbers that come out of an algorithm, but also in understanding the reasons behind it. For that, the major components of the algorithm should be interpretable. In particular, solutions to HA models with aggregate shocks usually involve mappings from the distribution over all individual states to the agent's welfare or decision outcomes. An ideal solution should provide interpretability of these mappings through an interpretable representation of the state distribution. An interpretable representation of the distribution is also necessary for deriving reduced dynamic models at the aggregate level from the original HA model.

- **Generality**: The method should in principle be applicable to a wide variety of different HA models (simple or complex), and to different notions of equilibrium (competitive equilibrium, or constrained efficiency problem).

Currently, there are two main approaches for solving HA models with aggregate shocks, and they satisfy only a subset of the requirements listed above. The first is the Krusell-Smith (KS) method, a global solution method proposed in Krusell and Smith (1998). The KS method approximates agent distribution with a small number of moments (e.g., the first moment), which are interpretable. It is efficient for solving simple HA models but becomes less effective when solving complex HA models with multiple shocks, or multiple endogenous states, or solving the estimation problem. That is due to the large number of variables (such as the number of moments needed) introduced and the resulting curse of dimensionality problem, i.e., as the number of variables increases, the computational cost goes up exponentially fast. The second approach is the local perturbation method proposed in Reiter (2009). The perturbation method allows us to study or estimate complex HA models, but it is not reliable for models where aggregate shocks bring significant nonlinear or nonlocal effects. Nonlinear effects are common in models with the zero lower bounds (ZLB). Nonlocal effects appear in models with large aggregate shocks, or in models (e.g., macro-finance models) where explicit consideration of aggregate uncertainty plays an important role in shaping agents' behavior, resulting in the deviation of the risky steady state from the deterministic steady state. Table 1 summarizes the advantages and limitations of these two methods.

Model features	KS method	Perturbation method	DeepHAM
Multiple shocks	No	Yes	Yes
Multiple endogenous states	No	Yes	Yes
Large shocks	Yes	No	Yes
Risky steady state	Yes	No	Yes
Nonlinearity (e.g., ZLB)	Yes	No	Yes

Table 1: Model features that different solution methods to HA models with aggregate shocks can handle.

In this paper, we propose a new solution method, *the Deep learning-based algorithm for Heterogeneous Agent Models (DeepHAM)*, which satisfies all the requirements listed above. We formulate an HA model with N agents, where N would be large if we hope to solve a problem with a continuum of agents. To solve HA models, the fundamental objects of interest are each agent's value and policy functions. A complication arises since these functions depend not only on the agent's own state, but also the distribution of all the agents in the economy. To address this issue, we represent the value function and policy function with deep neural networks, and present an algorithm to update the value and policy functions iteratively. Deep neural networks are a class of functions in deep learning, which have strong representational capability for high dimensional functions and can be efficiently optimized with stochastic gradient descent algorithms. Here we take advantage of deep learning not only as an efficient solution method for high dimensional problems, but also a flexible representation of high dimensional distributions. In contrast to existing literature that uses deep learning to represent high dimensional policy and value functions directly, we introduce *generalized moments* to represent the state distribution efficiently, and solve the value and policy outcomes as functions of the generalized moments. Generalized moments extract useful information from the state distribution like classical moments, and are automatically solved from the algorithm. The introduction of the generalized moments also ensures that the agent's optimal policy and value functions are invariant with respect to the permutation of the ordering of the agents. Conceptually, the generalized moments are consistent with the typical interaction form between the agents and market, defined through certain moments of the state distribution, and are readily interpretable. As we will see below, a single generalized moment not only leads to more accurate solutions than using only the first moment, but also extracts key information from the agent distribution that matters most for aggregate welfare and dynamics. Thus, it provides a general and interpretable way to study a key question in macroeconomics: whether, why, and how inequality matters for the macroeconomy.

As we will demonstrate later, DeepHAM meets all the requirements listed above. First, it shows better global accuracy compared with existing methods. In the baseline model we study, DeepHAM with only the first moment in the state vector reduces the Bellman equation error by 37.5% compared to the KS method. DeepHAM with one generalized moment reduces the error by 54.2%. Second, the computational cost of DeepHAM is quite low in solving complex HA models, and it does not suffer from the curse of dimensionality. DeepHAM can efficiently solve HA models with endogenous labor supply, or with a Brunnermeier and Sannikov (2014) type of financial sector. Third, the use of generalized moments allows us to revisit the classical question in the macroeconomic literature on whether and how heterogeneity matters to aggregate welfare and dynamics. Krusell and Smith (1998) famously argued that, in their setup, individual welfare is affected by other agents only through the mean of wealth distribution. With the generalized moments, we find that an unanticipated redistributional policy shock would have a non-zero welfare impact on those households who are not in the policy program, even when the mean of the wealth distribution is not affected. Finally, as a testament to the generality of DeepHAM, we demonstrate that it can be used to solve the constrained efficiency problem in HA models, known to be a challenging problem in the literature, as easily as solving the competitive equilibrium. This allows us to study optimal fiscal and monetary policy in HA models with aggregate shocks.

DeepHAM should be applicable to a large class of economic problems with heterogeneity and aggregate shocks, and here we list some such examples. First, since DeepHAM does not suffer from the curse of dimensionality with more endogenous states or shocks, we can introduce more realistic portfolio options like the housing and mortgage choices (Kaplan, Mitman, and Violante, 2020; Boar, Gorea, and Midrigan, 2021). We can efficiently handle models with *ex ante* heterogeneous agents, like households and financial experts (Brunnermeier and Sannikov, 2014), rational and bounded rational agents (Woodford and Xie, 2021), among others. We can study models with rich firm heterogeneity and aggregate shocks (Khan and Thomas, 2013). We can also study HA models with multiple shocks like those in the DSGE literature (McKay and Reis, 2016). Second, we can use DeepHAM to study models with large shocks like the COVID-19 shock, or large endogenous fluctuations like the ones discussed in Petrosky-Nadeau et al. (2018). We can also use it to study asset pricing and wealth effect of monetary and fiscal policy in a HA model, which has been proved to be important in the empirical literature (Andersen et al., 2021), but has only been studied in models with limited heterogeneity due to computational challenges (Kekre and Lenel, 2020; Caramp and Silva, 2021). We can also study the interaction of asset pricing and wealth inequality (Cioffi, 2021). Third, we can study optimal policy problems with heterogeneous agents using the Ramsey approach (Davila et al., 2012; Nuño and Moll, 2018), like optimal monetary and fiscal policy (Bhandari et al., 2021; Dyrda and Pedroni, 2021), or optimal macroprudential policy (Bianchi and Mendoza, 2018), which has been limited due to computational challenges. Last but not least, methodologically, we can extend DeepHAM to do model calibration, by introducing a calibration target in the objective function, so that we can solve and calibrate the HA models in the same algorithmic framework.

Related Literature. Our work builds on the extensive literature on solving HA models with aggregate shocks. As was discussed earlier, there are two main approaches in the literature (Algan et al., 2014): the global Krusell-Smith (KS) method (Krusell and Smith, 1998; Den Haan, 2010; Fernández-Villaverde et al., 2019; Schaab, 2020), and the local perturbation method (Reiter, 2009; Winberry, 2018; Ahn et al., 2018; Boppart et al., 2018; Auclert et al., 2021). Due to the curse of dimensionality, the KS method cannot handle complex HA models with multiple assets and multiple shocks. The perturbation method has been applied to studying complex HA models for the dynamics around the deterministic stationary equilibrium in the absence of aggregate shocks, but is inapplicable for problems with nonlinear dynamics induced by aggregate shocks, or problems that are not close to the deterministic stationary equilibrium. DeepHAM can handle complex HA models with aggregate shocks and provide a global solution.

This paper proposes a general methodology that extracts the key information of the distribution that matters for the aggregate welfare and dynamics. This is an important issue for studying the role of heterogeneity in macroeconomics (Kaplan et al., 2018; Auclert, 2019). For an overview of this literature, see Kaplan and Violante (2018). Most papers in this literature study the role of heterogeneity with quantitative decomposition after solving the model, while we propose a general numerical method that extracts key moments of the distribution in the process of the numerical solution. Our idea of the permutation invariant *generalized moments* coincides with the independent and contemporaneous work of Kahou et al. (2021), while we further explore the interpretation of the generalized moments and heterogeneity, and use it to study the impact of an unanticipated redistributional policy shock.

This work is also of relevance to the literature on machine learning-based algorithms for solving high dimensional dynamic programming problems in scientific computing (Han and E, 2016; Han et al., 2018; Fernández-Villaverde et al., 2020), and in macroeconomics (Duarte, 2018; Fernández-Villaverde, Hurtado, and Nuno, 2019; Azinovic, Gaegauf, and Scheidegger, 2019; Maliar, Maliar, and Winant, 2021). A notable recent contribution by Maliar et al. (2021) also uses deep learning to solve HA models with aggregate shocks. DeepHAM differs from that work in the following aspects. First, in the recursive formulation, DeepHAM addresses the optimization problem with directly simulated paths, while Maliar et al. (2021) optimizes over an objective function as a weighted sum of the Bellman residual and the first-order condition. Their setup requires a good approximation of the partial derivative of the high dimensional value function beyond just the value function itself. This is known to be a challenging task in deep learning. Second, we introduce generalized moments to make the high dimensional policy and value functions permutation invariant to the ordering of agents, which also improves interpretability. Third, this paper provides thorough discussions on both model performance and computational efficiency analysis, which are absent in their work. In addition, this paper presents the first example on using machine learning-based algorithms to solve the constrained efficiency problems in HA models with aggregate shocks.

The rest of this paper is organized as follows. Section 2 presents the DeepHAM method for solving a general HA model with aggregate shocks. Section 3 illustrates DeepHAM on the classical Krusell-Smith model, and highlights the main features of the current approach. Section 4 and 5 applies DeepHAM to more complex HA models and the constrained efficiency problem in HA models with aggregate shocks. Section 6 concludes the paper with some perspectives.

2 DeepHAM: A New Solution Method

2.1 General Setup of HA Models

In this subsection, we discuss the competitive equilibrium in a general HA model. We extend our setup to the constrained efficiency problem in Section 2.4. Consider a discrete time and infinite horizon economy consisting of N agents. N would be large if we hope to solve a problem with a continuum of agents, or smaller if we aim to solve for the strategic equilibrium with finite agents.[1]

For the agent i, her state dynamics, or law of motion for individual states, which usually comes from the agent's budget constraint, is given by:

$$s_{t+1}^i = f(s_t^i, c_t^i, X_t, \bar{\mathbb{S}}_t; z_{t+1}^i), \quad i = 1, \dots, N. \tag{1}$$

Here $s_t^i \in \mathbb{R}^{d_s}, c_t^i \in \mathbb{R}^{d_c}$ denote the state and decision (control) of agent i at period t. $z_t^i \in \mathbb{R}^{d_z}$ denotes the idiosyncratic shock at period t, and is a subvector of s_t^i. The set $\bar{\mathbb{S}}_t = \{(s_t^1, c_t^1), (s_t^2, c_t^2) \dots, (s_t^N, c_t^N)\}$ denotes the collection of all the agents' state-control pairs. Similarly, we use $\mathbb{S}_t = \{s_t^1, s_t^2, \dots, s_t^N\}$ to denote the collection of agent states. Another notation we will frequently use is $S_t = (s_t^1, s_t^2, \dots, s_t^N)$, whose value depends on the ordering of agents. $X_t \in \mathbb{R}^{d_X}$ denotes the aggregate state variables excluding $\mathbb{S}_t$. Here the law of motion of agent states (1) combines the agent's budget constraint, together with other optimization and market clearing conditions that characterize the aggregate prices in the budget constraint. For example, in Krusell and Smith (1998), the household's wealth state in the next period depends on current wealth and consumption, as well as current prices of labor and capital. The prices are explicit functions of individual wealth across distribution, a subvector of $\bar{\mathbb{S}}_t$, according to the representative firm's optimization condition and market clearing conditions. So the law of motion f for the individual state can be written in the form of (1).

The control variables are subject to some inequality constraints:

$$h_l(s_t^i, X_t, \mathbb{S}_t) \le c_t^i \le h_u(s_t^i, X_t, \mathbb{S}_t), \quad i = 1, \dots, N, \tag{2}$$

where h_l, h_u are vector functions with d^c dimensional output, and the dynamics of X_t is modeled by

$$X_{t+1} = g(X_t, \bar{\mathbb{S}}_t; Z_{t+1}), \tag{3}$$

[1] Although our model assumes a finite number of agents, numerical results suggest that a choice of $N = 50$ can approximate the solution to HA models with a continuum of *ex ante* homogeneous agents like Krusell and Smith (1998) quite well. We have also tested $N = 100, 200, ...$, and the results are similar.

where $Z_t \in \mathbb{R}^{d_Z}$ denotes the aggregate shock, and is usually a subvector of X_t. The aggregate state variable X_t also includes other aggregate quantities that affect the law of motion for individual states (1), but cannot be written only as functions of the collection of state-control pairs $\bar{\mathbb{S}}_t$. An example of X_t beyond Z_t is presented in Section 4.

Here we have indicated that f, g, h_l, h_u depend only on the sets $\bar{\mathbb{S}}_t$ or $\mathbb{S}_t$, not the ordering of the agents, i.e., the state dynamics (1) is invariant to the ordering of the agents in the economy. This is a consequence of the "mean-field" character of the interaction between agents. "Mean-field" is a concept that originated in physics, and describes the situation when the agents, or particles, interact with each other not directly, but through an empirical distribution that all the agents contribute to collectively and equally. A typical interaction form of mean-field type is through endogenous aggregate variables $O_t \in \mathbb{R}^{d_O}$ defined by

$$O_t = \frac{1}{N} \sum_{i=1}^{N} \mathcal{O}(s_t^i, c_t^i).$$

Here O_t is a function of $\bar{\mathbb{S}}_t$ as it is permutation invariant to the ordering of agents. Examples of O_t include the first or other moments of individual states. In this paper, we assume the dependence of f, g, h_l, h_u on $\bar{\mathbb{S}}_t$ or $\mathbb{S}_t$ can be written in explicit function forms. The Krusell-Smith model mentioned above is such an example. This point will be more clear in our presentation of the concrete examples in Sections 3 to 5.

According to the above description, $(X_t, \mathbb{S}_t)$ completely characterizes the state of the whole economy. Mathematically, we are interested in how agents should make decisions $c_t^i = \mathcal{C}(s_t^i, X_t, \mathbb{S}_t)$ through the decision rule $\mathcal{C}$ to achieve optimality.[2]

In the setting of competitive equilibrium, each agent i seeks to maximize her discounted lifetime utility:

$$\mathbb{E}_\mu \sum_{t=0}^{\infty} \beta^t u(c_t^i).$$

Here $u(\cdot)$ is the utility function and $\beta \in (0, 1)$ is the discount factor. μ is the distribution for the full state $(X_0, \mathbb{S}_0)$ at the initial time. We use $\mu(\mathcal{C})$ to denote the stationary distribution of $(X_t, \mathbb{S}_t)$ when every agent employs the decision rule $\mathcal{C}$ and we assume that such a stationary distribution always exists. The expectation is taken with respect to both the idiosyncratic and aggregate shocks over all time.

We say that $\mathcal{C}^*$ is an optimal policy in the competitive equilibrium, if for $\forall i \in \{1, \ldots, N\}$,

[2] Note here $\mathcal{C}$ is common across agents. This naturally applies for HA models with *ex ante* homogeneous agents like Krusell and Smith (1998). Similarly, the value function V defined later is common across agents as well. For models with *ex ante* heterogeneous agents, we may introduce additional individual state variables such that $\mathcal{C}$ is common across agents.

$\mathcal{C}^*$ solves agent i's problem

$$\begin{aligned}
&\max_{\mathcal{C}} \ \mathbb{E}_{\mu(\mathcal{C}^*)} \sum_{t=0}^{\infty} \beta^t u(s_t^i, c_t^i), \\
&\text{s.t. } (1)(2)(3) \text{ hold, } c_t^i = \mathcal{C}(s_t^i, X_t, \mathbb{S}_t), \\
&\text{given } c_t^j = \mathcal{C}^*(s_t^j, X_t, \mathbb{S}_t), j = 1, \ldots, N, j \neq i.
\end{aligned}$$

Note that in contrast to the perturbation method in Reiter (2009), which only computes the solution around the stationary equilibrium in the absence of aggregate shocks, a global solution method seeks to find the solution according to the stationary distribution $\mu(\mathcal{C}^*)$ of the economy, which may significantly differ from the stationary equilibrium without aggregate shocks (Kekre and Lenel, 2020; Bhandari et al., 2021).

Now we make two remarks about the general setup we present above.

Remark 1 (Discrete time setup). Throughout this paper, we formulate HA models in discrete time. Starting from Achdou, Han, Lasry, Lions, and Moll (2022), studying HA models in continuous time has received a lot of interest. Compared to discrete time, continuous time setup allows explicit expectation integral formulas with respect to the idiosyncratic shock in specific forms and efficient numerical algorithms for solving the corresponding PDE systems. However, when there are aggregate shocks, one needs to derive and solve the stochastic PDE systems (Carmona and Delarue, 2018) to obtain the global solution, which is much more challenging. Here we use a discrete time setup so that the problem is easier to solve, and more general forms of shocks can be included.

Remark 2 (Infinite horizon). We restrict the setup to the infinite horizon for the sake of conciseness when introducing the algorithm. As seen in Section 2.3, our method can also handle the finite horizon problems like life cycle models with a very small modification.

2.2 Representation of Agent Distribution and Generalized Moments

There are two important steps in introducing DeepHAM. The first is to introduce generalized moments to replace the distribution over all agents. This can be considered as a model reduction step. The second is to introduce an algorithm to solve the reduced model. We remark that the first step is of independent interest: The reduced model itself is a reliable and interpretable model that can be used as a general starting point for performing economic analysis. We discuss the idea of the first step in this subsection and present the detailed algorithm in the next subsection.

In HA models, a key question is what variables should be used to represent the whole economy. In algorithmic terms, this means what should be fed into the policy and value function approximators as input. Clearly, the individual state s_t^i and the aggregate state variable X_t should be taken into account. The main question is therefore how to represent the empirical distribution $\mathbb{S}_t = \{s_t^1, s_t^2, \ldots, s_t^N\}$, which affects the dynamics of s_t^i and X_t as well. Existing literature mostly adopts the following two approaches regarding this point.

1. The full vector S_t (see, e.g., Maliar et al., 2021). This vector contains the complete information of the distribution to characterize the optimal policy and value functions. However, there are two caveats. First, the dimension of S_t is proportional to N. So it would be extremely expensive to deal with economies with a large agent number. Second, the agent's optimal policy and value function should be invariant to the ordering of other agents' states. Simple function approximation form taking S_t as the direct input cannot enforce it straightforwardly.

2. Finite moments, usually the first moment of $\mathbb{S}_t$. This is the approach that the KS method has adopted. One can see it overcomes the two caveats of using the full vector S_t. However, the chosen moments may not carry the complete information necessary for the agents to evaluate the current environment and make the decision. The solution under this simplification may deviate from the ground truth, especially in complex HA models.

To go beyond the above limitations, we introduce a class of generalized aggregate variables $Q_t \in \mathbb{R}^{d_Q}$ into the state vector:

$$Q_t = \frac{1}{N} \sum_{i=1}^{N} \mathcal{Q}(s_t^i).$$

Here the basis function $\mathcal{Q}$ may be pre-determined function forms we specify, or general basis function with variational parameters. For pre-determined function forms, for example, it could be the identity function, then Q_t is the first moment. $\mathcal{Q}$ might also be an indicator function of whether an agent is at the borrowing constraint, then Q_t would capture the share of hand-to-mouth agents (Kaplan, Violante, and Weidner, 2014) in the economy. Q_t based on pre-determined $\mathcal{Q}$ nests the moment representation we discuss above. For general basis function $\mathcal{Q}$, we can parameterize it with neural networks (see, e.g., Han et al., 2019), and solve for the optimal representation in the algorithm. When $\mathcal{Q}$ is a general basis function, we call the resulting Q_t *"generalized moments"*.[3]

With Q_t as the representation of the distribution, we take (s_t^i, X_t, Q_t) as the state vector of policy and value functions. Conceptually, one can think of this in the following two ways.

[3]A brief mathematical introduction to neural networks is presented in Appendix A.

1. Instead of parameterizing the mapping $(s_t^i, X_t, \mathbb{S}_t) \mapsto$ output with neural network models, we decompose it into $(s_t^i, X_t, \mathbb{S}_t) \mapsto (s_t^i, X_t, Q_t) \mapsto$ output and parameterize the two components by two neural networks, the first step is an encoding network and the second step is a fitting network. In this sense, by specifying the dimension d_Q, we ensure that the complexity of the neural networks does not increase rapidly as N increases. In addition, the final policy functions are automatically permutation invariant. In this way, both shortcomings mentioned above are overcome.

2. Q_t shares the same formula with O_t, which can be interpreted as generalized moments. If $\mathcal{Q}$ is parameterized by a set of specific function forms with empirical knowledge, the generalized moments Q_t are selected from a large set of interpretable aggregate moments. If $\mathcal{Q}$ is directly parameterized by neural networks, optimizing $\mathcal{Q}$ can guide the agent to find the aggregate variables most relevant to their decision making. Compared to the KS method, one has more flexibility to represent the whole economy and can obtain a macroeconomic model closer to the underlying decision rules, without sacrificing interpretability.

In this paper, we use two separate sets of generalized moments Q_t^c and Q_t^V to extract the distribution information for policy and value functions. This choice simplifies the updating rule for them. Algorithmically, when we fed Q_t^c/Q_t^V into the policy/value functions, the variational parameters of the basis and policy/value parameters can be trained jointly end-to-end through the corresponding objective function. Using shared generalized moments for both the policy and value functions will be investigated in future work.

2.3 Solution Method for Competitive Equilibrium

We first describe the algorithm for solving the competitive equilibrium of the economies of the form described in Section 2.1. The algorithm for solving the constrained efficiency problem is quite similar and will be discussed in Section 2.4. To solve the model, we rewrite agents' objective function based on the dynamic programming principle over T periods. In the competitive equilibrium, for $\forall i \in \{1, \ldots, N\}$, the policy function $\mathcal{C}^*$ solves agent i's problem

$$
\begin{aligned}
&\max_{\mathcal{C}} \ \mathbb{E}_{\mu(\mathcal{C}^*)} \left[\sum_{t=0}^{T} \beta^t u\left(s_t^i, c_t^i\right) + \beta^T V(s_T^i, X_T, \mathbb{S}_T) \right], \qquad (4) \\
&\text{s.t. } (1)(2)(3) \text{ hold, } c_t^i = \mathcal{C}(s_t^i, X_t, \mathbb{S}_t), \\
&\text{given } c_t^j = \mathcal{C}^*(s_t^j, X_t, \mathbb{S}_t), j = 1, \ldots, N, j \neq i,
\end{aligned}
$$

where the value function V is defined by

$$V(s_0^i, X_0, \mathbb{S}_0) = \mathbb{E}\left[\sum_{t=0}^{\infty} \beta^t u\left(s_t^i, c_t^i\right) \big| X_0, \mathbb{S}_0\right], \tag{5}$$
$$\text{s.t. (1)(2)(3) hold, } c_t^i = \mathcal{C}^*(s_t^i, X_t, \mathbb{S}_t), i = 1, \dots, N.$$

The overall idea of the DeepHAM is an iterative procedure starting from the initial guess of the policy $\mathcal{C}_0$, as presented in Algorithm 1. Each iteration includes three steps that we will discuss in detail: (a) prepare the stationary distribution $\mu(\mathcal{C})$; (b) update the value function according to (5); (c) optimize the policy function according to (4). We refer to one such high-level iteration step as a *round* and repeat N_k rounds until convergence. It is similar to the conventional value function iteration algorithm (Ljungqvist and Sargent, 2018, Chapter 3), except that:

1. We parameterize both value and policy functions with neural networks, each of which nests two sub-networks with a feedforward architecture: one approximates the basis function $\mathcal{Q}$, the other approximates the mapping from (s_t^i, X_t, Q_t) to policy or value outcomes.[4]

2. We solve for optimal policy to maximize the total utility in (4) over Monte Carlo simulations for T periods, instead of one period, which is typically used in the conventional value function iteration algorithm. When the state vector is high dimensional, it is computationally expensive or even infeasible to update the policy with one period calculation on a grid. Instead, we update the parameters of the policy function neural network to maximize the expected total utility in (4) over simulated paths. We use T periods to reduce the variance in Monte Carlo simulation.

Now we further explain the details of the three main steps of DeepHAM in the k-th round.

Preparing the stationary distribution. This is done by simulating the economy (1)(3) for sufficiently long periods following the policy $\mathcal{C}_{k-1}$.

Updating the value function. Given the policy $\mathcal{C}_{k-1}$, updating the value function can be formulated as a supervised learning problem. Denote the parameters in the value function neural network as $\Theta^V = (\Theta^{VQ}, \Theta^{VO})$, where Θ^{VQ} are the parameters in the general basis

[4] If we choose to use some pre-determined basis to define Q such as the first moment, we will only have the second sub-network in both the policy and value functions.

Algorithm 1 DeepHAM for solving the competitive equilibrium

Require: Input: the initial policy $\mathcal{C}_0$, the initial value and policy neural networks with parameters Θ^V and Θ^C, respectively

for $k = 1, 2, \ldots, N_k$ **do**
  prepare the stationary distribution $\mu(\mathcal{C}_{k-1})$ according to the policy $\mathcal{C}_{k-1}$
  for $m = 1, 2, \ldots, N_{m_1}$ **do** ▷ update the value function
    sample N_{b_1} samples of $(X_0, \mathbb{S}_0)$ from $\mu(\mathcal{C}_{k-1})$
    compute the realized total utility in (8) through a single simulated path
    use the empirical version of (9) to compute the gradient ∇_{Θ_V}
    update Θ^V with ∇_{Θ_V}
  end for
  for $m = 1, 2, \ldots, N_{m_2}$ **do** ▷ optimize the policy function
    sample N_{b_2} samples of $(X_0, \mathbb{S}_0)$ from $\mu(\mathcal{C}_{k-1})$
    use the empirical version of (10) to compute the gradient ∇_{Θ_C}
    update Θ^C with ∇_{Θ_C}
  end for
  define $\mathcal{C}_k$ according to (11)
end for

function defining Q_t^V, Θ^{VO} are the parameters in the function that maps (s_t^i, X_t, Q_t^V) to value outcomes. So our approximation to the value function can be written as

$$V_{\mathrm{NN}}(s_t^i, X_t, \mathbb{S}_t; \Theta^V) := \widetilde{V}_{\mathrm{NN}}(s_t^i, X_t, Q_t^V; \Theta^{VO}) = \widetilde{V}_{\mathrm{NN}}(s_t^i, X_t, \frac{1}{N}\sum_{i=1}^{N} \mathcal{Q}_{\mathrm{NN}}(s_t^i; \Theta^{VQ})); \Theta^{VO}). \tag{6}$$

We want to use V_{NN} to approximate the agents' expected lifetime utility under the policy $\mathcal{C}_{k-1}$, i.e.,

$$V_{\mathrm{NN}}(s_t^i, X_t, \mathbb{S}_t; \Theta^V) \approx \mathbb{E}\left[\sum_{\tau=0}^{\infty} \beta^\tau u(s_{t+\tau}^i, c_{t+\tau}^i) \,\Big|\, s_t^i, X_t, \mathbb{S}_t\right]. \tag{7}$$

However, evaluating the expectation in (7) is still computationally expensive. To reduce the computation cost, given each sample $(s_0^i, X_0, \mathbb{S}_0)$ from the stationary distribution $\mu(\mathcal{C}_{k-1})$, we only simulate a single path with T_{simul} (T_{simul} sufficiently large) periods under the policy $\mathcal{C}_{k-1}$ to get the truncated realized total utility

$$\widehat{V}^i = \sum_{\tau=0}^{T_{\mathrm{simul}}} \beta^\tau u(s_\tau^i, c_\tau^i). \tag{8}$$

Note that $\widehat{V}_t^i$ is a random variable influenced by the realization of the idiosyncratic and aggregate shocks. Still, we know that the true value function minimizes the difference with the realized total utility. So we only need to solve the following regression problem to update the value function:

$$\min_{\Theta^V} \mathbb{E}_{\mu(\mathcal{C}_{k-1})} \left[V_{\text{NN}}(s_0^i, X_0, \mathbb{S}_0; \Theta^V) - \widehat{V}^i \right]^2. \tag{9}$$

We use the stochastic gradient descent algorithm to solve (9). Specifically, in each update step, we sample N_{b_1} samples of $(X_0, \mathbb{S}_0)$ from $\mu(\mathcal{C}_{k-1})$, use the empirical version of (9) to compute the gradient with respect to Θ^V by backpropagation, and update Θ^V accordingly. We repeat N_{m_1} steps to achieve convergence. As $\Theta^V = (\Theta^{VQ}, \Theta^{VO})$, at the end of the update, we also obtain the updated basis function $\mathcal{Q}_{\text{NN}}(\cdot; \Theta^{VQ})$ and the generalized moments Q_t at the same time.[5]

Optimizing the policy function. In the competitive equilibrium, the policy function is iteratively updated following the spirit of fictitious play (Brown, 1951). A similar idea has been used in Han and Hu (2020) and Hu (2021) to solve stochastic differential games based on neural networks. Similar to the value function, we will update the parameters associated with the policy function neural network through stochastic gradient descent. We call each update a "play". In each "play", we fix everyone but agent $i = 1$'s policy as that from the last play, and consider agent $i = 1$'s utility maximization problem to update the neural network parameters, to get the new policy in this "play". Then all the agents adopt the new policy in this "play". We repeat the "plays" until convergence.[6]

For the utility maximization problem of agent $i = 1$, the algorithm basically builds upon the one proposed in Han and E (2016): optimizing the parameters of the policy function neural network over simulated paths. Given the updated value function $V_{\text{NN}}(s_t^i, X_t, \mathbb{S}_t; \Theta^V)$ in the same round and other agents' policy function from the last "play", agent $i = 1$ aims to solve

$$\max_{\Theta^C} \mathbb{E}_{\mu(\mathcal{C}_{k-1})} \left[\sum_{t=0}^{T} \beta^t u(s_t^i, c_t^i) + \beta^T V_{\text{NN}}(s_T^i, X_T, \mathbb{S}_T; \Theta^V) \right], \tag{10}$$

with her policy parameterized by neural networks in the form of

$$\begin{aligned} c_t^i &= \mathcal{C}(s_t^i, X_t, \mathbb{S}_t; \Theta^C) \\ &= \left(h_u(s_t^i, X_t, \mathbb{S}_t) - h_l(s_t^i, X_t, \mathbb{S}_t)\right) \odot c_{\text{NN}}\left(s_t^i, X_t, \mathbb{S}_t; \Theta^C\right) + h_l(s_t^i, X_t, \mathbb{S}_t). \end{aligned} \tag{11}$$

Here the outputs of $h_u(\cdot), h_l(\cdot)$, and $c_{\text{NN}}(\cdot)$ are d^c dimensional, and $\odot$ denotes element-wise multiplication. We use the sigmoid function $\frac{1}{1+e^{-x}} \in [0, 1]$ as the last composed function of $c_{\text{NN}}(\cdot)$, so that the inequality constraints (2) are always satisfied. For a mathematical

[5]Backpropagation is an algorithm allowing an efficient computation of all partial derivatives of the neural network (composition of a series of functions) with respect to its parameters.

[6]In a more general setup, we can also fix the other agents' policies for several "plays" and then update from the agent $i = 1$'s policy.

introduction to the composition structure of neural networks, see Appendix A. Similar to the value function, the parameters in the policy function neural network $\Theta^C = (\Theta^{CQ}, \Theta^{CO})$, where Θ^{CQ} are parameters in the general basis function and Θ^{CO} are parameters in the function that maps (s_t^i, X_t, Q_t) to policy outcomes. So we have

$$c_{\text{NN}}(s_t^i, X_t, \mathbb{S}_t; \Theta^C) = \widetilde{c}_{\text{NN}}(s_t^i, X_t, Q_t^C; \Theta^{CO}) = \widetilde{c}_{\text{NN}}(s_t^i, X_t, \frac{1}{N}\sum_{i=1}^{N} \mathcal{Q}_{\text{NN}}(s_t^i; \Theta^{CQ})); \Theta^{CO}). \quad (12)$$

Again we use the stochastic gradient descent algorithm corresponding to (10). We sample N_{b_2} samples of $(X_0, \mathbb{S}_0)$ from $\mu(\mathcal{C}_{k-1})$ to be the initial conditions, use the empirical version of (9) to compute the gradient with respect to Θ^C, and update Θ^C accordingly. The gradient with respect to Θ^C can be obtained by backpropagation as well since all the component functions in (10)(11)(12) are explicit and differentiable; see Figure 1 for the computational graph corresponding to (10).

Figure 1: Computational Graph to Solve HA Models using DeepHAM

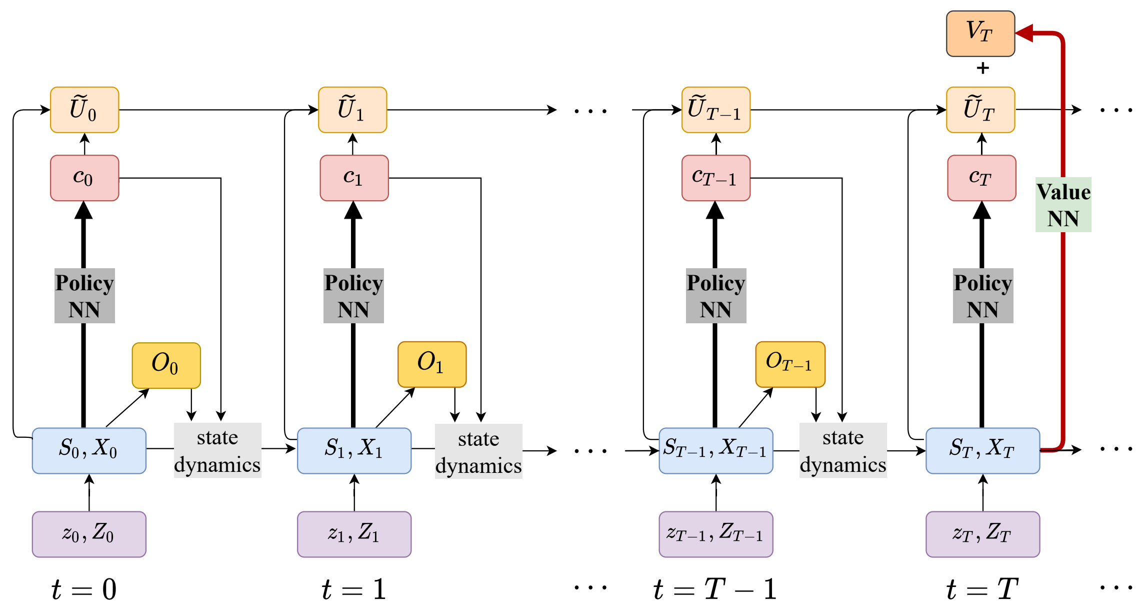


Figure note: S_t, z_t, and c_t denote the collection of all agents' states, idiosyncratic shocks, and decisions at time t, respectively. Z_t denotes aggregate shocks at time t. $\widetilde{U}_t$ denotes the collection of all agents' cumulative utilities up to period t, i.e., $\widetilde{U}_t^i = \sum_{\tau=0}^{t} \beta^\tau u\left(s_\tau^i, c_\tau^i\right)$.

From the above description, we can see one merit of DeepHAM: its readiness to deal with aggregate shocks. The algorithm bears very little difference when solving models with or without aggregate shocks. In contrast, the continuous time PDE approach (Achdou et al., 2022) can solve models without aggregate shocks efficiently (in the low-dimensional case), but face challenges in the presence of aggregate shocks.

2.4 Extension to Constrained Efficiency Problem

In the constrained efficiency problem, a benevolent social planner seeks to find a policy rule $\mathcal{C}$ determining each agent's decision variable c_t^i, to maximize the discounted sum of social welfare $\Omega(\bar{\mathbb{S}}_t)$ that depends on the collection of all the agents' state-control pairs:

$$\begin{aligned} &\max_{\mathcal{C}} \ \mathbb{E}_{\mu(\mathcal{C})} \sum_{t=0}^{\infty} \beta^t \Omega(\bar{\mathbb{S}}_t), \\ &\text{s.t. } (1)(2)(3) \text{ hold, } c_t^i = \mathcal{C}(s_t^i, X_t, \mathbb{S}_t), i = 1, \dots, N. \end{aligned}$$

Here the social welfare function can take the utilitarian form $\Omega(\bar{\mathbb{S}}_t) = \frac{1}{N}\sum_{i=1}^N u(s_t^i, c_t^i)$, or $\Omega(\bar{\mathbb{S}}_t) = \sum_{i=1}^N \omega_i u(s_t^i, c_t^i)$ with Negishi weights $\omega_i = \frac{u_c(s_t^i, c_t^i)}{\sum_{i=1}^N u_c(s_t^i, c_t^i)}$ (Bhandari et al., 2021), or other general forms.

The overall procedure for solving the constrained efficiency problem is the same as that for solving the competitive equilibrium, as presented in Algorithm 2: each round consists of (a) preparing the stationary distribution, (b) updating the value function, and (c) optimizing the policy function. Below we explain the three steps in the k-th round in detail and mainly highlight the differences with those in solving the competitive equilibrium, in terms of the value and policy objectives.

Algorithm 2 DeepHAM for solving the constrained efficiency problem

Require: Input: the initial policy $\mathcal{C}_0$, the initial value and policy neural networks with parameters Θ^V and Θ^C, respectively

for $k = 1, 2, \dots, N_k$ **do**
  prepare the stationary distribution $\mu(\mathcal{C}_{k-1})$ according to the policy $\mathcal{C}_{k-1}$
  for $m = 1, 2, \dots, N_{m_1}$ **do** ▷ update the value function
    sample N_{b_1} samples of $(X_0, \mathbb{S}_0)$ from $\mu(\mathcal{C}_{k-1})$
    compute the realized total social welfare in (14) through a single simulated path
    use the empirical version of (15) to compute the gradient ∇_{Θ_V}
    update Θ^V with ∇_{Θ_V}
  end for
  for $m = 1, 2, \dots, N_{m_2}$ **do** ▷ optimize the policy function
    sample N_{b_2} samples of $(X_0, \mathbb{S}_0)$ from $\mu(\mathcal{C}_{k-1})$
    use the empirical version of (16) to compute the gradient ∇_{Θ_C}
    update Θ^C with ∇_{Θ_C}
  end for
  define $\mathcal{C}_k$ according to (11)
end for

Preparing the stationary distribution. This is done exactly the same as that in the competitive equilibrium problem, by simulating the economy (1)(3) for sufficiently long periods following the policy $\mathcal{C}_{k-1}$.

Updating the value function. Given the policy $\mathcal{C}_{k-1}$, we want to use V_{NN} to approximate the expected total social welfare under the policy $\mathcal{C}_{k-1}$, i.e.,

$$V_{\mathrm{NN}}(X_t, \mathbb{S}_t; \Theta^V) \approx \mathbb{E}\left[\sum_{\tau=0}^{\infty} \beta^\tau \Omega(\bar{\mathbb{S}}_{t+\tau}) \,\middle|\, X_t, \mathbb{S}_t\right], \tag{13}$$

where

$$V_{\mathrm{NN}}(X_t, \mathbb{S}_t; \Theta^V) = V_{\mathrm{NN}}(X_t, Q_t^V; \Theta^{VO}) = V_{\mathrm{NN}}(X_t, \frac{1}{N}\sum_{i=1}^{N} \mathcal{Q}_{\mathrm{NN}}(s_t^i; \Theta^{VQ}); \Theta^{VO}).$$

Similarly, to avoid the computation cost of evaluating the expectation in (13), given each sample $(X_0, \mathbb{S}_0)$ from the stationary distribution $\mu(\mathcal{C}_{k-1})$, we only simulate a single path with T_{simul} (T_{simul} sufficiently large) periods under the policy $\mathcal{C}_{k-1}$ to get the truncated realized total social welfare

$$\widehat{V} = \sum_{\tau=0}^{T_{\mathrm{simul}}} \beta^\tau \Omega(\bar{\mathbb{S}}_\tau). \tag{14}$$

Then we only need to solve the following regression problem to update the value function

$$\min_{\Theta^V} \mathbb{E}_{\mu(\mathcal{C}_{k-1})}\left[V_{\mathrm{NN}}(X_0, \mathbb{S}_0; \Theta^V) - \widehat{V}\right]^2. \tag{15}$$

This can be done by using the stochastic gradient descent in the same way as that for solving the problem (9).

Optimizing the policy function. In order to find the constrained efficiency equilibrium, we need to update the policy function by solving

$$\max_{\Theta^C} \mathbb{E}_{\mu(\mathcal{C}_{k-1})}\left[\sum_{t=0}^{T} \beta^t \Omega(\bar{\mathbb{S}}_t) + \beta^T V_{\mathrm{NN}}(X_T, \mathbb{S}_T; \Theta^V)\right]. \tag{16}$$

Compared to the policy function optimization in the competitive equilibrium problem, here we get rid of the fictitious play step and instead optimize all the agents' policies simultaneously to maximize the total social welfare. The optimization problem (16) can be solved in the same way as that for the problem (10) with the stochastic gradient descent algorithm.

The gradient with respect to Θ^C can be obtained by backpropagation in the same computational graph in Figure 1.

From the above description, we can see that the DeepHAM Algorithm 2 for solving the constrained efficiency problem is identical to Algorithm 1 for the competitive equilibrium, except for the two differences in lines 6 and 11, where we use different objectives for the value function and policy function corresponding to the different settings. Meanwhile, the pipeline of data sampling and optimization methods for those different objectives are exactly the same. In this sense, DeepHAM can solve the constrained efficiency problem as easily as the competitive equilibrium problem.

3 DeepHAM for the Krusell-Smith Model

In this section, we illustrate DeepHAM on the classical Krusell-Smith model, and highlight the advantages of this method.

3.1 Model Setup

The setup follows Den Haan (2010). Household i's state $s_t^i = (a_t^i, z_t^i) \in \mathbb{R}^2$, with beginning-of-period wealth a_t^i, employment status $z_t^i \in \{0, 1\}$. The consumption $c_t^i \in \mathbb{R}$ is the control variable. Households have log utility over consumption. $Z_t \in \{Z^h, Z^l\}$ denotes aggregate productivity. The process z_t^i, Z_t follows a first-order Markov process. The aggregate state variable $X_t = Z_t$, so its dynamics (3) are trivial. The state dynamics of the household comes from the household budget constraint:

$$
\begin{aligned}
&a_{t+1}^i = (1 + r_t - \delta)a_t^i + [(1 - \tau_t)\bar{l}z_t^i + b(1 - z_t^i)]w_t - c_t^i, \\
&a_{t+1}^i \geq 0, \quad c_t^i \geq 0,
\end{aligned}
$$

where the net rate of return of capital is $r_t - \delta$, with depreciation rate δ. The factor prices r_t, w_t are determined by the first-order condition (FOC) of the representative firm, which produces with a Cobb-Douglas technology $Y_t = Z_t K_t^\alpha L_t^{1-\alpha}$, in the competitive factor market:

$$
w_t = Z_t(1 - \alpha)(K_t/L_t)^\alpha, \quad r_t = Z_t \alpha (K_t/L_t)^{\alpha - 1},
$$

with aggregate capital $K_t = \frac{1}{N}\sum_{i=1}^N a_t^i$ and labor supply $L_t = \bar{l}(L^h \mathbb{1}_{Z_t = Z^h} + L^l \mathbb{1}_{Z_t = Z^l})$, in which $\bar{l}$ is the time endowment of each agent. Unemployed agents ($z_t^i = 0$) receive unemployment benefits bw_t where b is unemployment benefit rate. Employed agents ($z_t^i = 1$) earn after-tax labor income $(1-\tau_t)\bar{l}w_t$, where tax rate $\tau_t = b(1-L_t)/\bar{l}L_t$, such that government budget constraint always holds (total tax income equals to unemployment benefits). This completes the specification of (1). The borrowing constraint and non-negative consumption constraint specifies (2). The calibration of the model follows Den Haan (2010) and is presented in Appendix B.1.

3.2 Results

We solve the Krusell-Smith model described above in the case of $N = 50$ using DeepHAM. We have tried other choices of $N = 100, 200, ...,$ and we find $N = 50$ is large enough to approximate the solution to the Krusell-Smith model. The computational graph for this problem is shown in Figure 2.

Figure 2: Computational Graph to Solve Krusell and Smith (1998) using DeepHAM

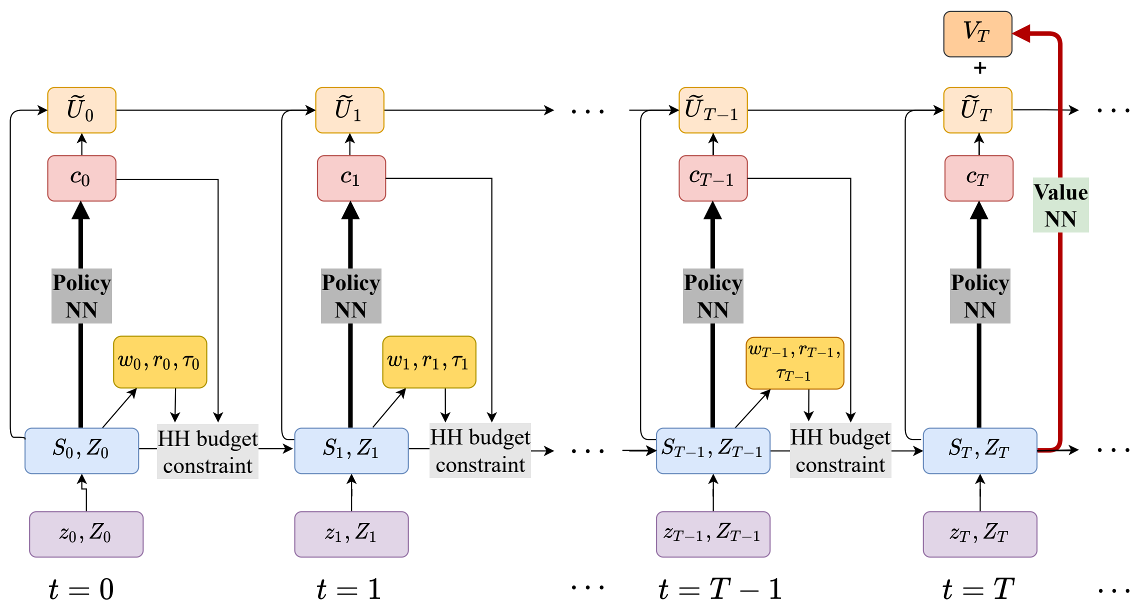


Figure note: S_t, z_t, c_t, and $\widetilde{U}_t$ denote the collection of all agents' states, idiosyncratic shocks, decisions, and cumulative utilities at time t, respectively. Z_t denotes aggregate shocks at time t. Aggregate prices w_t, r_t are determined by FOCs of the representative firm in the competitive factor market. Income tax rate τ_t depends on the aggregate shock Z_t and is pinned down in the government budget constraint.

3.2.1 Solution Accuracy

In Table 2, we compare the Bellman equation errors (defined in Appendix C) of DeepHAM to the same error of the KS method implemented in Maliar et al. (2010). For DeepHAM, we present accuracy measures when we include (1) the first moment of the household wealth distribution and (2) one generalized moment of the household wealth distribution in the state variable. We present the standard deviation of the Bellman errors from multiple runs of the numerical algorithm in the last column of Table 2. We see that all the results are statistically significant.[7]

Method and Moment Choice	Bellman error	Std of error
KS Method (Maliar et al., 2010)	0.0253	0.0002
DeepHAM with 1st moment	0.0184	0.0023
DeepHAM with 1 generalized moment	0.0151	0.0015

Table 2: Comparison of solution accuracy for Krusell-Smith problem

As we see in Table 2, the solutions obtained using DeepHAM are highly accurate. Compared to the KS method, DeepHAM with the first moment in the state vector reduces the Bellman equation error by 27.2%. DeepHAM with one generalized moment reduces the error by 40.3%. Generalized moments play an important role in improving solution accuracy since they provide a more concise representation of the household distribution and extract more information than the first moment. We discuss and interpret the generalized moment we obtain in the Krusell-Smith problem in the next subsection.[8]

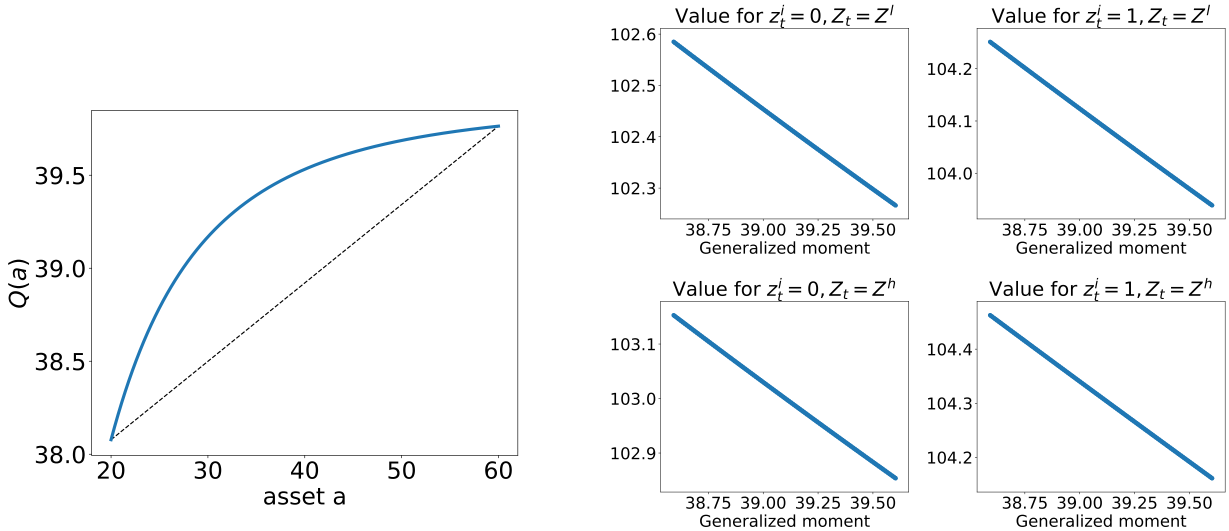
3.2.2 Generalized moments and redistributional effect

Among the three results in Table 2, DeepHAM with one generalized moment yields the most accurate solution. To better understand the improvement, we visualize the mapping from individual asset a_t^i to $\mathcal{Q}(a_t^i)$ provided by the learned basis, and the mapping from the generalized moment $\frac{1}{N}\sum_i \mathcal{Q}(a_t^i)$ to the value function in Figure 3.

We find that the basis function is concave in the individual asset, while the value function is linear with regard to the generalized moment. So households with different levels of wealth will have heterogeneous contributions to the generalized moment: giving an additional unit of assets to poor households increases the generalized moment more than giving the same assets to rich households. This phenomenon means that a redistributional policy would play a role for the aggregate welfare and dynamics even in the simple setup of Krusell and Smith (1998). Consider an unanticipated one-time policy shock (MIT shock): if one

[7]Following the moment construction in Krusell and Smith (1998), here the generalized moments are constructed on the wealth distribution, rather than the joint distribution of wealth and employment status.

[8]As is known in the literature, the KS method with the first moment can solve the Krusell-Smith model reasonably well. This is confirmed by the small Bellman error for the KS method in Table 2. Though DeepHAM can further improve the solution accuracy, the simulated economy based on the DeepHAM solution is highly consistent with that based on the solutions from the KS method. We present such comparison in Appendix D. This further confirms the accuracy of the DeepHAM solution in the Krusell-Smith model.



(a) Plot of $\mathcal{Q}_1(a)$

(b) Map $\frac{1}{N}\sum_i \mathcal{Q}_1(a_i)$ to the value function

Figure 3: Generalized moments for the Krusell-Smith problem. Left panel (solid blue line): a concave mapping from the individual asset to the basis function of the generalized moment. Right panel: mapping from the generalized moment to the value function, assuming household's individual assets fixed at the average level in the stationary equilibrium. Each figure in the right panel corresponds to one realization of idiosyncratic and aggregate shocks.

unit of asset is redistributed from the richest households to the poorest households, the welfare of "middle" households who are not in the redistribution program would decrease on impact since the generalized moment increases. Such an unanticipated policy shock will lead to higher aggregate savings in the future, since there are fewer people on the borrowing constraint. The increase in savings would lead to a higher future wage and a lower future asset return. Under the calibration of this model, the "middle" households who are not in the redistribution program earn more from the capital income than the labor income, so the unanticipated policy shock would make them worse off. This sensible logic is in contrast to the solution of the KS method. According to the KS method, households' welfare only depends on the first moment and individual states. So the redistributional policy shock would have no instantaneous welfare impact on those "middle" households who are not in the redistribution program, since the first moment of individual wealth distribution would not change.

4 DeepHAM for More Complex HA Models

In this section, we use DeepHAM to solve a HA model with a financial sector and aggregate shocks proposed in Fernández-Villaverde, Hurtado, and Nuno (2019). Compared to the

Krusell-Smith model with the two-state aggregate shocks in Section 3, here the aggregate shocks take values in a continuous range, which makes the problem more costly to solve.

4.1 Fernández-Villaverde et al. (2019): Model Setup

The setup is the discrete time version of Fernández-Villaverde et al. (2019). We use the subscript t and $t + \Delta t$ to highlight that it comes from the discretization of a continuous time model, but it should be interpreted as the dynamics between t and $t+1$ in the general setup in Section 2. In this economy, there are N households who save in risk-free bonds and consume. Their labor supply is exogenous and exposed to idiosyncratic shocks. There is a representative financial expert who issues risk-free bonds to households and invests in productive capital. A representative firm produces with capital from the financial expert and with labor supplied by the households. The growth rate of productive capital is exposed to aggregate shocks.

Household's problem. For household i, her state is $s_t^i = (a_t^i, z_t^i) \in \mathbb{R}^2$, with beginning-of-period risk-free asset a_t^i, and the idiosyncratic shocks on labor supply $z_t^i \in \{z_1, z_2\}$ with $0 < z_1 < z_2$. The process z_t^i follows a first-order Markov process with ergodic mean 1 such that the aggregate labor supply $L_t = 1$. Household i has constant relative risk aversion (CRRA) utility from consumption c_t^i with parameter $\gamma > 0$ and discount factor $e^{-\rho \Delta t}$.

The household budget and borrowing constraints determines the state dynamics (1) of household i:

$$
\begin{aligned}
a_{t+\Delta t}^i &= a_t^i + (w_t z_t^i + r_t a_t^i - c_t^i)\Delta t, \qquad (17)\\
a_{t+\Delta t}^i &\geq 0, \quad c_t^i \geq 0,
\end{aligned}
$$

where the aggregate prices are characterized below. Aggregate risk-free asset demand $B_t = \frac{1}{N}\sum_{i=1}^{N} a_t^i$.

Representative firm's problem. The firm produces with a Cobb-Douglas technology $Y_t = K_t^{\alpha} L_t^{1-\alpha}$. She hires labor L_t from households at wage w_t, and rents capital K_t from the financial expert at rental rate rc_t, both in the competitive factor market:

$$
w_t = (1-\alpha)(K_t/L_t)^{\alpha}, \quad rc_t = \alpha (K_t/L_t)^{\alpha-1}. \qquad (18)
$$

Financial expert's problem. The representative financial expert issues risk-free bond B_t at rate r_t to households, and rents capital K_t at rate rc_t to the representative firm. Her net worth $W_t = K_t - B_t$. For the financial expert, the instantaneous return rate on capital is exposed to aggregate shocks Z_t:

$$\frac{K_{t+\Delta t} - K_t}{K_t} = (rc_t - \delta)\Delta t + \sigma Z_t \sqrt{\Delta t},$$

where δ is the depreciation rate of capital, σ is the volatility of aggregate shocks, Z_t follows i.i.d. standard normal distribution.[9]

The financial expert has log utility with discount rate $\widehat{\rho} < \rho$ over consumption $\widehat{C}_t$, so she consumes a constant share of her net worth: $\widehat{C}_t = \widehat{\rho} W_t$, and chooses a leverage ratio proportional to excess return of risky capital $\frac{K_t}{W_t} = \frac{1}{\sigma^2}(rc_t - \delta - r_t)$. So the risk-free return is

$$r_t = \alpha (K_t/L_t)^{\alpha-1} - \delta - \sigma^2 \frac{K_t}{W_t}. \tag{19}$$

The budget constraint of the financial expert $W_{t+\Delta t} = W_t + (rc_t - \delta)K_t \Delta t + \sigma K_t Z_t \sqrt{\Delta t} - B_t r_t \Delta t - \widehat{C}_t \Delta t$ implies the dynamics of net worth W_t:

$$W_{t+\Delta t} = W_t + \left(\alpha K_t^{\alpha-1} - \delta - \widehat{\rho} - \sigma^2 \left(1 - \frac{K_t}{W_t}\right) \frac{K_t}{W_t}\right) W_t \Delta t + \sigma K_t Z_t \sqrt{\Delta t}. \tag{20}$$

Using the general descriptive variables in Section 2, the aggregate state $X_t = W_t$. Since $K_t = B_t + W_t$, the evolution of W_t only depends on $W_t, B_t = \frac{1}{N}\sum_{i=1}^{N} a_t^i, Z_t$. The aggregate state dynamics (3) is specified as (20). Equations (17)(18)(19), together with the stochastic process of z_t^i, complete the specifications of (1) and (2). The calibration of the model follows Fernández-Villaverde et al. (2019) and is presented in Appendix B.2.

4.2 Solution Accuracy and Efficiency

We use DeepHAM to obtain the global solution to the problem described above with $N = 50$. We compare the Bellman equation errors (see the definition in Appendix C) of DeepHAM to the generalized KS method with nonlinear perceived law of motion implemented in Fernández-Villaverde et al. (2019) in Table 3. For DeepHAM, we present accuracy measures when we include (1) only the first moment and (2) one generalized moment of household asset distribution, in the state variable.

As we see in Table 3, the solutions obtained using DeepHAM are highly accurate. Compared to the generalized KS method with nonlinear law of motion implemented by Fernández-Villaverde et al. (2019), DeepHAM with the first moment or a generalized moment can both

[9] In the continuous time model, the aggregate shock is a white noise with volatility σ. There would be a slight numerical difference in the discretized model, but the difference is tiny as we choose a tiny Δt.

Method and Moment Choice	Bellman error	Std of error
KS Method (Fernández-Villaverde et al., 2019)	0.00417	0.00011
DeepHAM with 1st moment	0.00405	0.00059
DeepHAM with 1 generalized moment	0.00422	0.00086

Table 3: Comparison of solution accuracy on a HA model with a financial sector and aggregate shocks. The KS method refers to the solution method implemented by (Fernández-Villaverde et al., 2019).

obtain global solutions with the same level of accuracy. We present the standard deviation of the Bellman errors from multiples runs of the numerical algorithm in the last column of Table 3.[10]

To solve this HA model with a financial sector and aggregate shocks that take values in a continuous range, it only takes the DeepHAM 12% more time, compared to solving the simple Krusell-Smith model in Section 3. This result demonstrates that the efficiency of DeepHAM in studying complex HA models with aggregate shocks: unlike the grid-based method, the computation cost of DeepHAM does not increase quickly when the number of state variables or grid points increases.

5 DeepHAM for Constrained Efficiency Problem in HA Models

In this section, we solve the constrained efficiency problem in HA models using DeepHAM. In contrast to the competitive equilibrium, the constrained optimum of HA models, defined as the allocation decided by a benevolent social planner who maximizes the social welfare, is much harder to solve. Existing literature only handles constrained optimum for HA models without aggregate shocks (Davila, Hong, Krusell, and Ríos-Rull, 2012; Nuño and Moll, 2018), and the computational cost is much higher than for solving the competitive equilibrium of the same model.

In contrast, as presented in Section 2.4, DeepHAM can solve the constrained efficiency problem as easily as the competitive equilibrium. We illustrate this advantage by using DeepHAM to solve the constrained efficiency problem in an Aiyagari model as Davila et al. (2012), and in a HA model with aggregate shocks.

[10] To note, Fernández-Villaverde et al. (2019) builds and solves the model in continuous time. To compare it with DeepHAM, which solves the discrete time version of the model, we evaluate their solution also on the discrete time Bellman equation error defined in Appendix C. Since we choose a small Δt, the continuous time solution should give an approximate meaningful Bellman error in discrete time. The bottom line is that DeepHAM obtains an accurate solution that is comparable to Fernández-Villaverde et al. (2019).

5.1 Model Setup

Baseline setup without aggregate shocks. The baseline setup is an Aiyagari model that follows the "high wealth dispersion" calibration in Davila et al. (2012) to match the US wealth inequality. There are N *ex ante* homogeneous households in this economy. Household i's labor supply is subject to idiosyncratic shocks $z_t^i \in \{e_0, e_1, e_2\}$, which are i.i.d. across agents and follow a Markov process. Household i accumulates asset a_t^i in the form of real capital. Household i's state $s_t^i = (a_t^i, z_t^i)$ follows

$$\begin{aligned} a_{t+1}^i &= (1 + r_t - \delta)a_t^i + w_t z_t^i - c_t^i, \\ a_{t+1}^i &\geq 0, \quad c_t^i \geq 0, \end{aligned} \tag{21}$$

where consumption c_t^i is the only control variable. The representative firm produces with a Cobb-Douglas technology $Y_t = K_t^\alpha L_t^{1-\alpha}$, rents capital and hires labor in a competitive factor market. So the wage w_t and capital rental rate r_t are:

$$w_t = (1 - \alpha)(K_t/L_t)^\alpha, \quad r_t = \alpha(K_t/L_t)^{\alpha-1}, \tag{22}$$

where aggregate saving $K_t = \frac{1}{N}\sum_{i=1}^N a_t^i$ and labor supply $L_t = \bar{L}$ is constant. This completes the specification of (1) and (2). Since there is no aggregate shock in the baseline setup, there is no aggregate state variable X_t nor the dynamics (3).

The benevolent social planner seeks to find a policy rule $\mathcal{C}$ determining c_t^i for all the households $i = 1, ..., N$ and $t = 0, 1, ..., \infty$ to maximize the utilitarian objective:

$$\max_{\mathcal{C}} \frac{1}{N}\mathbb{E}_{\mu(\mathcal{C})} \sum_{i=1}^N \sum_{t=0}^\infty \beta^t u(c_t^i),$$

subject to the constraints in (21).[11]

Setup with aggregate shocks. We also solve the constrained efficiency problem of a HA model with aggregate shocks. On top of the baseline model above, we introduce aggregate productivity shocks $Z_t \in \{Z^l, Z^h\}$ that follows a Markov process, on the production

[11] Given the form of the utilitarian objective, we have an alternative approach to approximate the expected social welfare besides the one presented in Section 2.4. We can learn the individual value function defined in (7) with the approximation form (6) and use that to approximate the expected total social welfare by taking the average of the individual value function. We have used the latter approach in this work.

technology of the representative firm $Y_t = Z_t K_t^{\alpha} L_t^{1-\alpha}$, such that the factor prices are

$$w_t = Z_t(1-\alpha)(K_t/L_t)^{\alpha}, \quad r_t = Z_t\alpha(K_t/L_t)^{\alpha-1}, \tag{23}$$

where aggregate saving $K_t = \frac{1}{N}\sum_{i=1}^{N} a_t^i$ and labor supply $L_t = (L^h \mathbb{1}_{Z_t=Z^h} + L^l \mathbb{1}_{Z_t=Z^l})$. Using the general descriptive variables in Section 2, the aggregate state variable $X_t = Z_t$. We also introduce countercyclical idiosyncratic risk to the model, so that the probability for households to get into low income state $z_t^i = e_0$ becomes larger in the bad aggregate state, and smaller in the good aggregate state. Our setup follows the "integration principle" proposed by Krusell et al. (2009), so that when aggregate shocks are eliminated, the model will exactly reduce to the baseline setup. Equations (21) and (22) (or (23)), together with the stochastic process of z_t^i, complete the specifications of (1) and (2). The calibration of both models are presented in Appendix B.3.

5.2 Results

We solve the constrained planner's problems with $N = 50$ in both the baseline model without aggregate shocks, and in the model with aggregate shocks and countercyclical idiosyncratic shocks. The equilibrium statistics of these problems are presented in Table 4 and 5. In comparison, we also present equilibrium statistics under the competitive equilibrium of the same models.[12]

	No aggregate shock		Aggregate shock	
	Market	Constrained Opt.	Market	Constrained Opt.
Average assets	30.635	119.741	34.296	95.811
Output	10.294	16.816	12.159	17.592
Capital-output ratio	2.976	7.120	2.821	5.446
Interest rate	4.097%	-2.944%	4.678%	-1.433%
Coefficient of variation of wealth	2.621	2.483	2.574	2.924
Wealth Gini	0.864	0.862	0.812	0.878
Coefficient of variation of consumption	1.548	0.710	1.699	0.736
Consumption Gini	0.615	0.386	0.578	0.388

Table 4: Equilibrium statistics in the market outcome (competitive equilibrium) and constrained optimum for models without or with aggregate shocks.

[12]When solving the constrained efficiency problem, we usually find two local maximums of the problem. Since we are solving the constrained planner's problem, we only take the local optimum with higher expected total welfare. We leave the study of the "second best" constrained optimum for future research.

	Positive aggregate shock		Negative aggregate shock	
	Market	Constrained Opt.	Market	Constrained Opt.
Average assets	36.316	99.793	32.260	91.826
Output	13.925	20.038	10.393	15.146
Capital-output ratio	2.608	4.980	3.104	6.063
Interest rate	5.147%	-1.116%	4.208%	-1.750%
Coefficient of variation of wealth	2.533	2.894	2.614	2.953
Wealth Gini	0.815	0.877	0.805	0.877
Coefficient of variation of consumption	1.693	0.756	1.697	0.713
Consumption Gini	0.599	0.407	0.542	0.345

Table 5: Equilibrium statistics in the market outcome (competitive equilibrium) and constrained optimum for the HA model with aggregate shocks, conditional on the realization of aggregate shock.

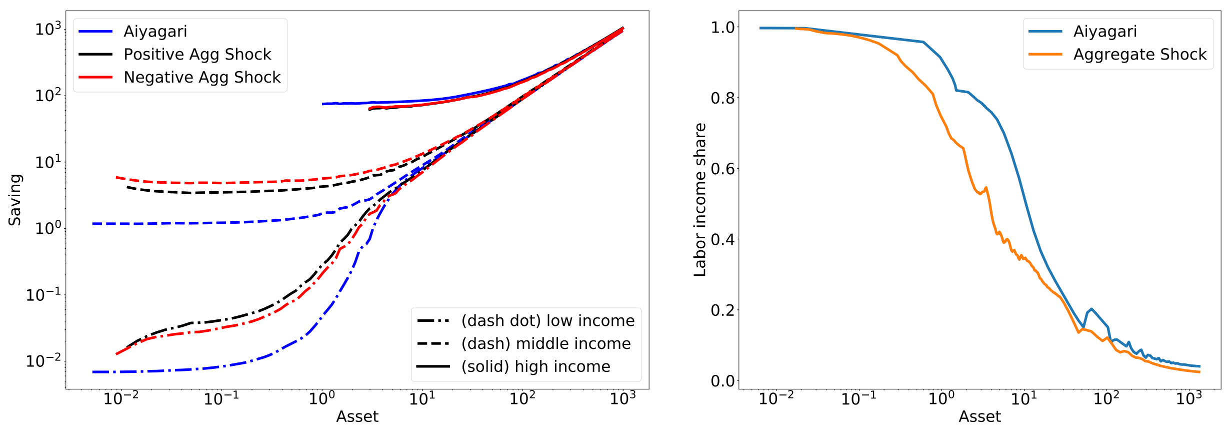
The main findings are as follows. First, in both models with or without aggregate shocks, the constrained optimum requires a much higher level of capital than the competitive equilibrium. In the absence of aggregate shocks, the planner chooses the capital level 3.90 times that in the laissez-faire equilibrium, which is consistent with the finding of Davila et al. (2012). This is because the planner with the utilitarian objective hopes to redistribute from rich households to poor households to improve the social welfare. Since the poor households have a higher labor income share, the planner would raise the aggregate capital level, so that the wage rate would increase, and capital return would decrease according to equations (22) and (23). The poor households would be better off in such a redistribution by raising the aggregate capital level. Meanwhile, in both models, the constrained efficiency problem features a similar level of wealth inequality and a lower level of consumption inequality, compared to the market outcome.

Second, compared to the constrained optimum in the absence of aggregate shocks, the model with aggregate shocks features a lower level of aggregate capital stock. With aggregate shocks and countercyclical unemployment risks, the planner chooses the capital level 2.79 times that in the laissez-faire equilibrium, which is lower than the 3.90 times in the model without aggregate shocks. The reason is that with aggregate shocks, households, especially poor households, have a stronger precautionary saving motive, thus their labor income share is lower than that in the model without aggregate shocks. So the planner would still raise aggregate capital to redistribute through price changes, but not as much as in the economy without aggregate shocks. We provide further validation of this explanation in Section 5.3.

Third, according to Table 5 where we present equilibrium statistics conditional on the realization of aggregate shocks, the planner wants households to increase savings in a higher ratio (2.84) in the bad aggregate state, compared to that (2.74) in the good aggregate state.

5.3 Impact of Aggregate Shocks on Constrained Optimum

To further understand the impact of aggregate shocks on the constrained efficiency problem, we compare household's saving policy and labor share across asset distribution in the constrained optimum with aggregate shocks and without aggregate shocks ("Aiyagari economy") in Figure 4. As is shown in Figure 4a, in the presence of aggregate shocks, households, especially wealth-poor households, save more than they do in the Aiyagari economy due to precautionary motives. As a result, households, especially wealth-poor households, have a lower labor income share compared to the Aiyagari economy, which is shown in Figure 4b. As a result, with the presence of aggregate shocks, the constrained planner does not need to raise the aggregate capital level as much as in the economy without aggregate shocks, to redistribute towards the wealth-poor households.



(a) Saving policy

(b) Labor share

Figure 4: Household's saving policy and labor share along asset distribution in the constrained optimum. In the model with aggregate shocks, households, especially wealth-poor households, have more precautionary saving and lower labor income share, compared to the economy without aggregate shocks.

5.4 Computational Efficiency for Constrained Efficiency Problem

In this section, we highlight the computational efficiency of DeepHAM in solving the constrained planner's problem. In Table 6, we report the computational cost for DeepHAM and the classical method (Davila et al., 2012) for solving the constrained efficiency problem in the baseline Aiyagari model, and in the model with aggregate shocks.

	Aiyagari model	With aggregate shocks
Classical method	15 hours	not solved in the literature
DeepHAM	20 minutes	32 minutes

Table 6: Comparison of the computational speed for the constrained efficiency problem. The conventional method (Davila et al., 2012) is implemented on a laptop with a 2.3Ghz Dual-Core Intel Core i5 processor. DeepHAM is implemented on a small cluster with a NVIDIA Tesla P100 GPU.

According to Table 6, it is very costly to solve the constrained efficiency problem using the classical method, even for the baseline Aiyagari model without aggregate shocks. To our knowledge, the global solution of the constrained efficiency problem in HA models with aggregate shocks has not been presented in the literature. In contrast, DeepHAM can handle this class of problems quite efficiently.

6 Conclusion

We presented DeepHAM, an efficient, reliable, and interpretable deep learning-based method for globally solving HA models with aggregate shocks. DeepHAM achieves highly accurate results, and can be applied to high dimensional HA models without suffering from the curse of dimensionality. The algorithm automatically generates a flexible and interpretable representation of the agent distribution, through generalized moments, that matter the most for the aggregate economy. This feature helps us to understand whether and how heterogeneity matters in macroeconomics. Moreover, DeepHAM can solve the constrained efficiency problem as fast as solving the competitive equilibrium, a significant advantage over existing methods. The results demonstrate that DeepHAM is a powerful tool for studying global patterns of complex HA models with aggregate shocks. This opens up many exciting possibilities for future research.

In this paper, we assumed that the dependence of the aggregate prices and quantities on individual states could be written in explicit function forms. This assumption excludes HA models with aggregate variables that are determined recursively as a function of expected future aggregate variables, for example, the inflation rate in a New Keynesian Philips curve (NKPC), or the indirect utility in the Epstein-Zin preference. Handling forward looking equilibrium conditions in global solution methods is crucial towards solving HANK models with aggregate shocks and requires an additional price function in the algorithm. We leave this for further discussion in a companion paper. Another avenue for future research is to develop an estimation algorithm based on DeepHAM.

References

Achdou, Yves, Jiequn Han, Jean-Michel Lasry, Pierre-Louis Lions, and Benjamin Moll (2022), "Income and wealth distribution in macroeconomics: A continuous-time approach." *The Review of Economic Studies*, 89, 45–86.

Ahn, SeHyoun, Greg Kaplan, Benjamin Moll, Thomas Winberry, and Christian Wolf (2018), "When inequality matters for macro and macro matters for inequality." *NBER Macroeconomics Annual*, 32, 1–75.

Algan, Yann, Olivier Allais, Wouter J Den Haan, and Pontus Rendahl (2014), "Solving and simulating models with heterogeneous agents and aggregate uncertainty." In *Handbook of Computational Economics*, volume 3, 277–324, Elsevier.

Andersen, Asger Lau, Niels Johannesen, Mia Jørgensen, and José-Luis Peydró (2021), "Monetary policy and inequality."

Auclert, Adrien (2019), "Monetary policy and the redistribution channel." *American Economic Review*, 109, 2333–67.

Auclert, Adrien, Bence Bardóczy, Matthew Rognlie, and Ludwig Straub (2021), "Using the sequence-space jacobian to solve and estimate heterogeneous-agent models." *Econometrica*, 89, 2375–2408.

Azinovic, Marlon, Luca Gaegauf, and Simon Scheidegger (2019), "Deep equilibrium nets." *Available at SSRN 3393482*.

Bhandari, Anmol, David Evans, Mikhail Golosov, and Thomas J Sargent (2021), "Inequality, business cycles, and monetary-fiscal policy." *Econometrica*, 89, 2559–2599.

Bianchi, Javier and Enrique G Mendoza (2018), "Optimal time-consistent macroprudential policy." *Journal of Political Economy*, 126, 588–634.

Boar, Corina, Denis Gorea, and Virgiliu Midrigan (2021), "Liquidity constraints in the US housing market." *The Review of Economic Studies*.

Boppart, Timo, Per Krusell, and Kurt Mitman (2018), "Exploiting MIT shocks in heterogeneous-agent economies: the impulse response as a numerical derivative." *Journal of Economic Dynamics and Control*, 89, 68–92.

Brown, George W (1951), "Iterative solution of games by fictitious play." *Activity analysis of production and allocation*, 13, 374–376.

Brunnermeier, Markus K and Yuliy Sannikov (2014), "A macroeconomic model with a financial sector." *American Economic Review*, 104, 379–421.

Caramp, Nicolas and Dejanir H Silva (2021), "Monetary policy and wealth effects: The role of risk and heterogeneity." Technical report.

Carmona, René and François Delarue (2018), *Probabilistic Theory of Mean Field Games with Applications II: Mean Field Games with Common Noise and Master Equations*, volume 84. Springer.

Cioffi, Riccardo A (2021), "Heterogeneous risk exposure and the dynamics of wealth inequality."

Cybenko, George (1989), "Approximation by superpositions of a sigmoidal function." *Mathematics of control, signals and systems*, 2, 303–314.

Davila, Julio, Jay H Hong, Per Krusell, and José-Víctor Ríos-Rull (2012), "Constrained efficiency in the neoclassical growth model with uninsurable idiosyncratic shocks." *Econometrica*, 80, 2431–2467.

Den Haan, Wouter J (2010), "Comparison of solutions to the incomplete markets model with aggregate uncertainty." *Journal of Economic Dynamics and Control*, 34, 4–27.

Duarte, Victor (2018), "Machine learning for continuous-time economics." *Available at SSRN 3012602*.

Dyrda, Sebastian and Marcelo Pedroni (2021), "Optimal fiscal policy in a model with uninsurable idiosyncratic shocks." *Review of Economic Studies*.

Fernández-Villaverde, Jesús, Samuel Hurtado, and Galo Nuno (2019), "Financial frictions and the wealth distribution." Technical report, National Bureau of Economic Research.

Fernández-Villaverde, Jesús, Galo Nuno, George Sorg-Langhans, and Maximilian Vogler (2020), "Solving high-dimensional dynamic programming problems using deep learning." Technical report.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016), *Deep learning*. MIT press.

Han, Jiequn and Weinan E (2016), "Deep learning approximation for stochastic control problems." *arXiv preprint arXiv:1611.07422, Deep Reinforcement Learning Workshop, Conference on Neural Information Processing Systems*.

Han, Jiequn and Ruimeng Hu (2020), "Deep fictitious play for finding markovian nash equilibrium in multi-agent games." In *Mathematical and Scientific Machine Learning*, 221–245, PMLR.

Han, Jiequn, Arnulf Jentzen, and Weinan E (2018), "Solving high-dimensional partial differential equations using deep learning." *Proceedings of the National Academy of Sciences*, 115, 8505–8510.

Han, Jiequn, Chao Ma, Zheng Ma, and Weinan E (2019), "Uniformly accurate machine learning-based hydrodynamic models for kinetic equations." *Proceedings of the National Academy of Sciences*, 201909854.

Hornik, Kurt, Maxwell Stinchcombe, and Halbert White (1989), "Multilayer feedforward networks are universal approximators." *Neural networks*, 2, 359–366.

Hu, Ruimeng (2021), "Deep fictitious play for stochastic differential games." *Communications in Mathematical Sciences*, 19, 325–353.

Kahou, Mahdi Ebrahimi, Jesús Fernández-Villaverde, Jesse Perla, and Arnav Sood (2021), "Exploiting symmetry in high-dimensional dynamic programming." Technical report, National Bureau of Economic Research.

Kaplan, Greg, Kurt Mitman, and Giovanni L Violante (2020), "The housing boom and bust: Model meets evidence." *Journal of Political Economy*, 128, 3285–3345.

Kaplan, Greg, Benjamin Moll, and Giovanni L Violante (2018), "Monetary policy according to HANK." *American Economic Review*, 108, 697–743.

Kaplan, Greg and Giovanni L Violante (2018), "Microeconomic heterogeneity and macroeconomic shocks." *Journal of Economic Perspectives*, 32, 167–94.

Kaplan, Greg, Giovanni L Violante, and Justin Weidner (2014), "The wealthy hand-to-mouth." *Brookings Papers on Economic Activity*, 2014, 77–138.

Kekre, Rohan and Moritz Lenel (2020), "Monetary policy, redistribution, and risk premia." *University of Chicago, Becker Friedman Institute for Economics Working Paper*.

Khan, Aubhik and Julia K Thomas (2013), "Credit shocks and aggregate fluctuations in an economy with production heterogeneity." *Journal of Political Economy*, 121, 1055–1107.

Krusell, Per, Toshihiko Mukoyama, Ayşegül Şahin, and Anthony A Smith Jr (2009), "Revisiting the welfare effects of eliminating business cycles." *Review of Economic Dynamics*,

12, 393–404.
Krusell, Per and Anthony A Smith, Jr (1998), "Income and wealth heterogeneity in the macroeconomy." *Journal of Political Economy*, 106, 867–896.
Ljungqvist, Lars and Thomas J Sargent (2018), *Recursive macroeconomic theory*. MIT press.
Maliar, Lilia, Serguei Maliar, and Fernando Valli (2010), "Solving the incomplete markets model with aggregate uncertainty using the krusell–smith algorithm." *Journal of Economic Dynamics and Control*, 34, 42–49.
Maliar, Lilia, Serguei Maliar, and Pablo Winant (2021), "Deep learning for solving dynamic economic models." *Journal of Monetary Economics*, 122, 76–101.
McKay, Alisdair and Ricardo Reis (2016), "The role of automatic stabilizers in the US business cycle." *Econometrica*, 84, 141–194.
Nuño, Galo and Benjamin Moll (2018), "Social optima in economies with heterogeneous agents." *Review of Economic Dynamics*, 28, 150–180.
Petrosky-Nadeau, Nicolas, Lu Zhang, and Lars-Alexander Kuehn (2018), "Endogenous disasters." *American Economic Review*, 108, 2212–45.
Reiter, Michael (2009), "Solving heterogeneous-agent models by projection and perturbation." *Journal of Economic Dynamics and Control*, 33, 649–665.
Schaab, Andreas (2020), "Micro and macro uncertainty."
Winberry, Thomas (2018), "A method for solving and estimating heterogeneous agent macro models." *Quantitative Economics*, 9, 1123–1151.
Woodford, Michael and Yinxi Xie (2021), "Fiscal and monetary stabilization policy at the zero lower bound: consequences of limited foresight." *Journal of Monetary Economics*.

Appendix

A Neural Networks: a Class of Function Approximator

In this paper, we consider deep, fully connected feedforward neural networks. A network $y = u(x; \Theta)$ with L ($L \geq 1$) hidden layers defines a mapping $\mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$, in which $x \in \mathbb{R}^{d_1}$ is the input variable, $y \in \mathbb{R}^{d_2}$ is the output variable, and $\Theta = (W_1, b_1, \ldots, W_{L+1}, b_{L+1})$ is the collection of network parameters. The network's mapping is defined by a series of composition of linear transformation and nonlinear activation:

$$\begin{aligned}
y &= \sigma_{L+1} \circ (W_{L+1} z_L + b_{L+1}), \\
z_L &= \sigma_{L-1} \circ (W_{L-1} z_{L-1} + b_{L-1}), \\
&\vdots \\
z_2 &= \sigma_2 \circ (W_2 z_1 + b_2), \\
z_1 &= \sigma_1 \circ (W_1 x + b_1).
\end{aligned}$$

Here $W_l \in \mathbb{R}^{m_l \times m_{l-1}}$ is called the weight matrix, and $b_l \in \mathbb{R}^{m_l}$ is called the bias vector, $l = 1, \ldots, L{+}1$. We have $m_0 = d_1, m_{L+1} = d_2$, and $m_1, \ldots, m_l$ are set as network hyperparamters. $\sigma_l : \mathbb{R} \to \mathbb{R}$ is a scalar function called activation function, and $\circ$ denotes element-wise operation. The typical choices of σ_l includes rectified linear unit (ReLU) $\sigma(x) = \max\{(0, x)\}$, sigmoid function $1/(1 + e^{-x})$, and so on. Usually σ_l are all the same for $l = 1, \ldots, L$ and σ_{L+1} is chosen as the identity function to ensure the output is not restricted. In our policy function neural network $c_{\mathrm{NN}}(\cdot)$ in equation (11), we choose σ_{L+1} as the sigmoid function such that the inequality constraints on the decision variable can be satisfied.

It has been proved in Hornik et al. (1989); Cybenko (1989) that neural networks with one hidden layer neural networks are universal approximators, i.e., they can approximate arbitrary well any unknown Borel measurable function over a compact domain. In recent years, it has been extensively studied empirically and theoretically that deep neural networks with multiple hidden layers have better approximation and optimization efficiency (Goodfellow et al., 2016).

B Details of the Model Setup

B.1 Krusell and Smith (1998) Model

Calibration. The parameters follow Den Haan (2010) with each period as one quarter. The capital share $\alpha = 0.36$, depreciation rate of capital $\delta = 0.25$, household discount factor $\beta = 0.99$, labor supply $\bar{l} = 1/0.9$, unemployment benefit rate $b = 0.15$, and and households have log utility. Aggregate productivity $Z_t \in \{1.01, 0.99\}$. The joint transition matrix for idiosynacratic and aggregate shock is $\Pi_Z = \begin{bmatrix} 0.525 & 0.35 & 0.03125 & 0.09375 \\ 0.038889 & 0.836111 & 0.002083 & 0.122917 \\ 0.09375 & 0.03125 & 0.291667 & 0.583333 \\ 0.009115 & 0.115885 & 0.024306 & 0.850694 \end{bmatrix}$, where the four rows and columns correspond to $(Z_t, z_t) = (0.99, 0), (0.99, 1), (1.01, 0), (1.01, 1)$

respectively.

B.2 Fernández-Villaverde et al. (2019) Model

Calibration. We follow Fernández-Villaverde et al. (2019) for parameters. The capital share $\alpha = 0.35$, depreciation rate of capital $\delta = 0.1$, household discount rate $\rho = 0.05$, expert discount rate $\widehat{\rho} = 0.04971$, volatility of aggregate shocks $\sigma = 0.014$, and the risk aversion of the households $\gamma = 2$. To solve the problem in discrete time, we choose $\Delta t = 0.2$, which should be a small number so that the solution is comparable to the continuous time solution. Households' discount factor $\beta = e^{-\rho\Delta t}$. The transition matrix of idiosyncratic shocks is $\Pi_e = \begin{bmatrix} 1-\lambda_1\Delta t & \lambda_1\Delta t \\ \lambda_2\Delta t & 1-\lambda_2\Delta t \end{bmatrix}$ where $\lambda_1 = 0.986$, $\lambda_1 = 0.052$.

B.3 Davila et al. (2012) Model

Calibration. The parameter setting in the baseline model follows Davila et al. (2012). The capital share $\alpha = 0.36$, depreciation rate of capital $\delta = 0.08$, discount factor $\beta = 0.887$, and the risk aversion of the households $\gamma = 2$. Labor endowment $z_t^i \in \{e_0 = 1, e_1 = 5.29, e_2 = 46.55\}$, and $\Pi_e = \begin{bmatrix} 0.992 & 0.008 & 0 \\ 0.009 & 0.980 & 0.011 \\ 0 & 0.083 & 0.917 \end{bmatrix}$ with stationary distribution $\{0.498, 0.443, 0.059\}$. The aggregate labor supply $L_t = \bar{L} = 0.498e_0 + 0.443e_1 + 0.059e_3 = 5.574$.

In the model with aggregate shocks, $Z_t \in \{Z^l, Z^h\} = \{0.95, 1.05\}$ with transition matrix $\Pi_Z = \begin{bmatrix} 0.875 & 0.125 \\ 0.125 & 0.875 \end{bmatrix}$. The idiosyncratic shocks are countercyclical, and the transition matrix across labor endowment $\Pi_{e,t} = \begin{bmatrix} 0.98 & 0.02 & 0 \\ 0.009 & 0.980 & 0.011 \\ 0 & 0.083 & 0.917 \end{bmatrix}$ when $Z_t > 1$, $\Pi_{e,t} = \begin{bmatrix} 0.6512 & 0.3488 & 0 \\ 0.978 & 0.011 & 0.011 \\ 0 & 0.083 & 0.917 \end{bmatrix}$ when $Z_t < 1$. The aggregate labor supply in good and bad aggregate states are $\{L^h, L^l\} = \{7.525, 3.623\}$.

C Accuracy Measures

The main accuracy measure we adopt in this paper is the Bellman equation error defined in this section. We choose it over the Euler equation error since it provides a better measure over the whole state space, especially the region close to the inequality constraints, without the need to introduce the Lagrangian multiplier.

In the general HA model in Section 2, the agent i's optimization problem can be characterized in a recursive form:

$$V(s_t^i, X_t, \mathbb{S}_t) = \max_{c_t^i} \left[u(c_t^i) + \beta \mathbb{E} V(s_{t+1}^i, X_{t+1}, \mathbb{S}_{t+1}) \right].$$

Given the solved value function $V(\cdot)$, we can evaluate the Bellman equation error for each state $(s_t^i, X_t, \mathbb{S}_t)$ in the state space as:

$$\mathrm{err_B}(s_t^i, X_t, \mathbb{S}_t) = \left| V(s_t^i, X_t, \mathbb{S}_t) - \max_{c_t^i} \left[u(c_t^i) + \beta \mathbb{E} V(s_{t+1}^i, X_{t+1}, \mathbb{S}_{t+1}) \right] \right|$$

where the expectation operator is approximated by the Monte Carlo sampling of the aggregate and idiosyncratic shocks, and the consumption choice c_t^i is solved again given the solved value function $V(\cdot)$, rather than directly taken from the optimal policy we have solved.

We then average it with respect to the stationary distribution over $(X_t, \mathbb{S}_t)$ to calculate the Bellmen equation error for the solution we obtain:

$$\mathrm{err_B} = \mathbb{E}_{\mu(\mathcal{C}^*)} \left| V(s_t^i, X_t, \mathbb{S}_t) - \max_{c_t^i} \left[u(c_t^i) + \beta \mathbb{E} V(s_{t+1}^i, X_{t+1}, \mathbb{S}_{t+1}) \right] \right|.$$

D Solution Comparison for Krusell-Smith Model

In this section, we compare the simulated economy based on the DeepHAM solution with that based on the solutions from the KS method. Under the same realization of idiosyncratic and aggregate shocks, the two economies simulated with 1000 agents based on the two solution methods are presented in Figure 5.[13]

As widely validated in the literature, the KS method with the first moment can solve the Krusell-Smith model reasonably well. Though DeepHAM can further improve the solution accuracy as we present in Section 3.2.1, the simulated economy based on the DeepHAM solution is highly consistent with that based on the solutions from the KS method, as shown

[13]The DeepHAM solution comes from a finite agent model with $N = 50$, while we simulate the two economies with 1000 agents to show the two solutions are consistent with each other in simulations with a relatively large agent number.

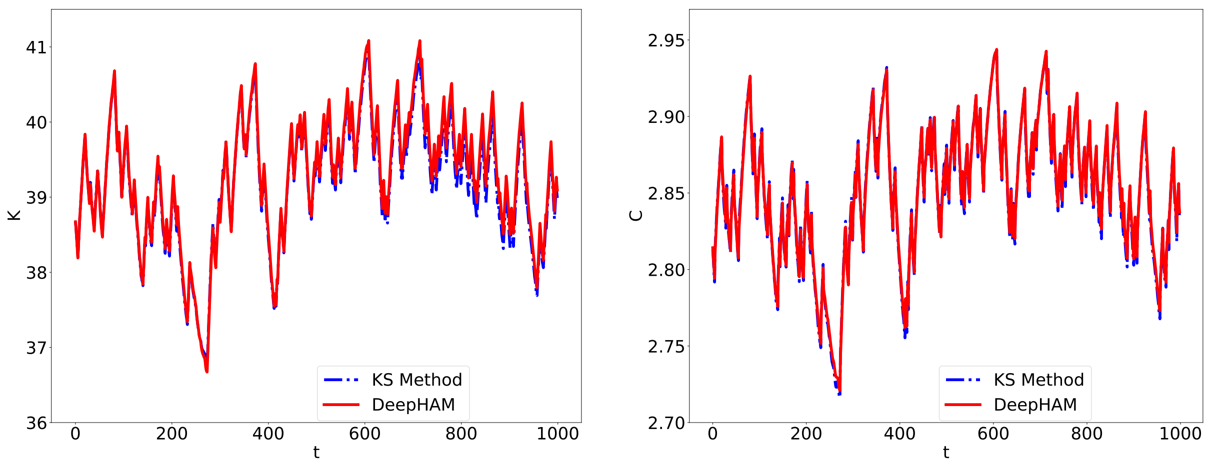


Figure 5: The simulated paths based on solutions obtained from the KS method and DeepHAM, under the same realization of idiosyncratic and aggregate shocks. Left panel: aggregate capital (K_t); right panel: aggregate consumption (C_t).

in Figure 5. This further confirms the accuracy of the DeepHAM solution in the Krusell-Smith model.