Deep Learning for Search and Matching Models*

Jonathan Payne† Yucheng Yang‡

November 29, 2024

First draft: February 15, 2024

Abstract

We develop a new method to globally solve and calibrate search and matching models with aggregate shocks and heterogeneous agents. We characterize general equilibrium as a high dimensional partial differential equation with the distribution as a state variable. We then use deep learning to solve and calibrate the model using the simulated method of moments. This allows us to study search markets that are not "block recursive" and compute variables (e.g. wages and prices) that were previously unattainable. In applications to labor search models, we show that distribution feedback plays a more important role when aggregate shocks have an asymmetric impact across agents. Positive assortative matching weakens more in prolonged expansions, disproportionately benefiting low-wage workers. In applications to over-the-counter markets, we show how financial crises impact bond yields across different maturities.

Keywords: Search and Matching, Distribution Feedback, Two-sided Heterogeneity, Business Cycles, Sorting, Over-the-Counter Financial Markets, Deep learning.

*We thank Adam Rebei for his invaluable contributions during the early stage of the project. We thank Fernando Alvarez, Isaac Baley, Jaroslav Borovička, Katarína Borovičková, Niklas Engbom, Jesús Fernández-Villaverde, Lukas Freund, Pablo Guerron, Lars Hansen, Ji Huang, Zhen Huo, Greg Kaplan, Felix Kübler, Ricardo Lagos, Jeremy Lise, Guido Menzio, Giuseppe Moscarini, Jesse Perla, Xincheng Qiu, Jean-Marc Robin, Tom Sargent, Edouard Schaal, Venky Venkateswaran, Gianluca Violante, and seminar participants at ASU, Atlanta Fed, Chicago, CKGSB, EIEF, Minnesota, Norges Bank, NYU, PKU, PSE, Rice, Tsinghua, Yale; CEF, DSE, ESIF, OzMac, Sargent Alumni Reading Group, SED, Swiss Winter Macro Finance Workshop, Zurich Workshop on the Frontier of Quantitative Macro for helpful comments and suggestions. We thank Nikolai Kurlovich and Runhuan Wang for outstanding research assistance. We thank the Julis-Rabinowitz Center for Public Policy and Finance and the Swiss National Science Foundation for generous financial support.

†Princeton University, Department of Economics. Email: jepayne@princeton.edu

‡University of Zurich, Department of Finance and SFI. Email: yucheng.yang@uzh.ch

1 Introduction

Many important questions in macroeconomics and finance involve aggregate shocks and agent heterogeneity in markets with search and matching frictions. However, modeling these features has proven technically challenging. The need for tractability leads many researchers to simplify the problem by limiting how the cross-sectional distribution impacts agent decision making (e.g. the "block-recursive" equilibria in Lagos and Rocheteau (2009), Menzio and Shi (2011), Lise and Robin (2017)[1]). Developments in the deep learning literature have opened up the possibility to relax these restrictions. In this paper, we present a general formulation of heterogeneous agent search and matching models as a collection of high dimensional partial differential equations (PDEs) with the agent distribution explicitly as a state variable. We then develop the first deep learning method for solving these models and calibrating parameters using simulated method of moments. This allows us to study crisis and business cycle dynamics in a wide class of search models with complex agent heterogeneity.

We focus on models with the following features. The economy is populated by heterogeneous agents (e.g. workers or investors) and heterogeneous institutions (e.g. firms or financial intermediaries) that can be matched or unmatched. Matches generate utility that depends upon the idiosyncratic agent and institution types and an exogenous aggregate variable that follows a continuous time Markov chain. Unmatched agents and institutions engage in random search to meet each other. Upon meeting, they choose whether to accept the match and then bargain over the division of the matching surplus. We show that the equilibrium for this economy can be characterized recursively with a state space consisting of the exogenous aggregate variable and the distribution of matches across types in the economy. The match distribution impacts agent decisions because the opportunity cost of accepting a match depends upon which other agents are looking for matches and which institutions are available to match. The equilibrium is then characterized by a "master" PDE for the surplus function, which includes high dimensional derivatives capturing the impact of distributional changes on surplus.

We propose a deep learning approach for solving this class of PDEs and calibrat-

[1]Following Menzio and Shi (2011) and Lise and Robin (2017), we call an equilibrium "block-recursive" if the agents' value and policy functions are independent of the endogenous distribution of agents across their idiosyncratic states.

ing parameters using simulated method of moments, which we refer to as DeepSAM. To solve the model, we approximate the surplus function with a neural network and then train the neural network parameters to minimize the average loss in the PDE on a random collection of sample points. To our knowledge, we are the first to apply deep learning to study search and matching models. In doing so, we face a number of technical challenges. First, unlike for some competitive market models (e.g. Krusell and Smith (1998)), the distribution does not enter the agent optimization problem exclusively through aggregate prices. Instead, agents need to forecast the distribution of which other agents they will meet and so the Master equation necessarily involves integrating over the equilibrium surplus function weighted by the population match density and the acceptance function. This makes the problem more difficult. Second, search and matching models typically involve a free entry condition, which requires solving a non-linear fixed point problem during each step of the PDE training. Third, we need to stabilize the training algorithm for unusually shaped surplus functions and for neural networks that include economic parameters as input variables. For such cases, we use a "homotopy" approach combined with sampling in the economically relevant part of the state space. This involves training the neural network parameters that give low curvature and then gradually retraining the model with updated parameters.

We extend our solution algorithm to calibrate the model using the simulated method of moments by building on the state-of-the-art practices in the deep learning literature. Instead of solving the model repeatedly across different economic structural parameter values, we introduce these parameters as pseudo-state variables and solve the resulting Extended Master Equations using deep learning. This provides an explicit solution to the problem over a large range of structural parameter values. We then use this solution to build a surrogate model mapping structural parameters to simulated moments that we utilize to undertake for internal calibration.

In Section 3, we deploy our method to solve a "canonical" labor market search model with two-sided heterogeneity that has been extended to include aggregate shocks. This model can be thought of as either the Shimer and Smith (2000) model with two-sided heterogeneity and aggregate shocks, or as the Mortensen and Pissarides (1994) model with worker and firm heterogeneity. We test our solution in a number of ways. We start by examining the neural network approximation to a model without aggregate shocks since the steady state of this model can be solved

with existing fixed point solution techniques. We show that the average squared difference between our solution and the fixed point solution is in the order of magnitude of 10^{-6}. For the model with aggregate shocks, there are no existing solutions to which we can compare so we instead study the training error and stability of the solution. We show that the average numerical error on the differential equation over the full state space is in the order of 10^{-7}. We interpret these results as strong evidence that our neural network training algorithm can find an accurate solution.

We use our solution to the labor search model to study the role of distributional feedback during the COVID-19 pandemic. We calibrate the model to include a crisis state that generates the heterogeneous employment decline across worker skill groups and firm industries that Cajner, Crane, Decker, Grigsby, Hamins-Puertolas, Hurst, Kurz, and Yildirmaz (2020) estimate occurred during COVID. We then compute the impulse response following a COVID shock and decompose the time path by comparing the results to the "restricted" dynamics when agents make matching decisions under the "myopic" belief that they are always at the ergodic employment distribution. We find that in the full model unemployment falls approximately 30% faster during the recovery than in the restricted model. This is because, in the full model, firms understand that COVID disproportionately increases unemployment among low-skill workers, which leads them to forecast a higher opportunity cost of waiting for a high-type match and so offer jobs to a wider range of workers. We also consider a counterfactual crisis where unemployment increases symmetrically across all agent types and show that for this case the restricted dynamics closely approximate the full dynamics. This illustrates that solving the model globally across the distribution is particularly important for understanding large, asymmetric shocks such as COVID or the Great Depression.

In Section 4, we introduce on-the-job search and endogenous separation into our baseline labor search model. This builds on the framework used in Lise and Robin (2017) but extends it to allow workers to possess positive bargaining power during both initial and on-the-job search meetings. We use our deep learning approach to calibrate the model to match empirical moments of the US labor market. The entire solution and calibration process takes 5 hours and 5 minutes, where the model is solved over the economic parameter space and simulated across 10,000 parameter combinations to build the surrogate model deployed for the simulated method of moments. Our calibrated model finds support for the Okun (1973) hypothesis that

low-wage workers benefit disproportionately from longer expansions. This occurs because there is counter-cyclical sorting in our model that is more pronounced as expansions and recessions are prolonged. During an expansion (recession) high-type workers become increasingly scarce (plentiful) so high-type firms become less (more) picky in their job offers. Consequently, longer expansions lead to greater weakening of positive assortative matching, which benefits low-skilled workers.

From a technical perspective, our method allows us to study two features that the labor literature has previously been unable to examine: business cycle dynamics in two-sided heterogeneity models when workers have positive bargaining power and the dynamics of the wage distribution. This is because Lise and Robin (2017) and subsequent papers impose zero worker bargaining power in initial meetings and Bertrand competition in on-the-job meetings, which makes the surplus function block recursive but does not make the division of the surplus block recursive. We show that unemployment and vacancies are more responsive to business cycle shocks when worker bargaining power is small, indicating that the assumptions required to generate block recursivity are quantitatively important. We also show that the wages of low-type workers are more procyclical over the business cycle.

In Section 5, we use DeepSAM to solve an over-the-counter bond market model with heterogeneous investors, different bond maturities, and aggregate default risk. This can be thought of as an extension to Duffie, Gârleanu, and Pedersen (2005) and Weill (2008) that expands investor and asset heterogeneity and allows for aggregate risk. From a technical point of view, relative to the labor models in the earlier sections, this model introduces type switching and asset trade. We use our model to study how liquidity and institutional frictions impact bond prices at different maturities. We show that a financial crisis shock that increases the liquidity needs of hedge funds and increases default risk has more impact on long-maturity bonds. This offers a search-theoretic rationale for the volatility of the term structure.

Literature Review. Over the past three decades, there have been major advances in solving search and matching models with heterogeneity and aggregate risk. One advance is the Bertrand competition model of wage setting introduced in Postel-Vinay and Robin (2002) and deployed in many papers (e.g. Cahuc, Postel-Vinay, and Robin (2006); Lise and Robin (2017)). Another is the directed search block recursive structure introduced by Moen (1997); Menzio and Shi (2010, 2011). Both these

approaches impose contracting and entry assumptions that ensure agent decisions are independent of the distribution of matches. Our paper relaxes these constraints and solves a general class of models where the distribution may impact agents' decision making. Besides solving the search models with two-sided heterogeneity, which is known to be very challenging, our approach can also be applied to solve complex models with one-sided heterogeneity, for example models with endogenous separation, among other features.

Our paper is also connected to recent papers studying business cycle dynamics in heterogeneous agent labor search models (e.g. Krusell, Mukoyama, Rogerson, and Şahin (2017); Schaal (2017); Moscarini and Postel-Vinay (2018); Engbom (2021); Fukui (2020); Baley, Figueiredo, and Ulbricht (2022); Alves (2022); Qiu (2023); Moscarini and Postel-Vinay (2023); Birinci, Karahan, Mercan, and See (2024)). Our DeepSAM approach offers a way to expand the range of models used in this literature by enriching agent heterogeneity, relaxing block recursivity, departing from perfect foresight, studying non-linear crisis dynamics, and potentially other extensions. We demonstrate this by studying the labor market impact of the COVID-19 pandemic and analyzing feedback mechanisms generated by alterations in the distribution, which in turn influence agent decisions and aggregate dynamics.

We are part of a growing computational economics literature using deep learning methods to solve economic models and overcome the limitations of traditional solution techniques. These papers have focused on solving heterogeneous agent macroeconomic models with incomplete but competitive markets (e.g. Azinovic, Gaegauf, and Scheidegger (2022), Maliar, Maliar, and Winant (2021), Han, Yang, and E (2021), Kahou, Fernández-Villaverde, Perla, and Sood (2021), Fernández-Villaverde, Hurtado, and Nuno (2023), Gopalakrishna (2021), Sauzet (2021), Huang (2022), Gu, Lauriere, Merkel, and Payne (2023), Azinovic and Žemlička (2023), Duarte, Duarte, and Silva (2024), Huang (2024), among others, see the recent review by Fernández-Villaverde, Nuno, and Perla (2024)). Our contribution is to show how to undertake deep learning to solve search and matching models, which are workhorse models for a large literature in macroeconomics and finance. What makes them difficult to solve compared to many competitive incomplete market models (e.g. Krusell and Smith (1998)) is that the shape of the distribution matters directly for the equilibrium. This is because, as summarized in Table 1, the distribution impacts agents' decisions via the matching probability with other types rather than through aggregate prices. This

imposes greater challenges on how we develop our numerical and sampling schemes to get an accurate solution. Another reason why search and matching models are difficult is that we have to impose the free entry condition, which typically involves another fixed point iteration.

	Distribution	How distribution affect agents' decisions
HAM	Asset wealth and income	Via aggregate prices
SAM	Type (productivity) of agents in two sides of matching	Via matching process with other types

Table 1: How distribution matters in heterogeneous agent models (HAM) vs search and matching (SAM) models.

Our calibration approach builds on the ideas of introducing structural parameters as pseudo-state variables, first proposed by Norets (2012) and recently extended by Chen, Didisheim, and Scheidegger (2023); Kase, Melosi, and Rottner (2024); Friedl, Kübler, Scheidegger, and Usui (2023); Duarte and Fonseca (2024). Our PDE formulation is related to the master equations in Bilal (2023); Alvarez, Lippi, and Souganidis (2023). Finally, our training approach draws on the "Physics-informed neural networks" (PINN) literature (e.g. Raissi, Perdikaris, and Karniadakis (2019)) in computational science.

The paper is structured as follows. Section 2 describes our DeepSAM methodology for solving and calibrating a general class of search and matching models. Section 3 applies DeepSAM to solve a canonical labor market search model with two-sided heterogeneity and aggregate shocks, and studies the impact of the COVID-19 shock. Section 4 applies DeepSAM to calibrate a richer model with on-the-job search and endogenous separation. Section 5 applies DeepSAM to a search model of over-the-counter bond markets with heterogeneous investors, different bond maturities, and aggregate default risks. Section 6 concludes.

2 Methodology

In this section, we outline a general environment that nests search and matching models from various strains of the search and matching literature. We then introduce our deep learning algorithm, DeepSAM, to solve the model. Finally, we outline a

method for using neural networks to undertake efficient calibration to match simulated moments (a deep learning based "simulated method of moments").

2.1 Environment

Setting: The economy is in continuous time with an infinite horizon. The economy is populated by a continuum of infinitely lived agents (e.g. workers or investors) indexed by type $x \in \mathcal{X}$, and a continuum of institutions (e.g. firms or financial dealers) indexed by type $y \in \mathcal{Y}$. Agents are either employed in a match (e) or unmatched (u). Institutions are either producing in a match (p) or vacant (v). The distribution of matches between agents and institutions is endogenous, and determined by agent and institution decisions. Agents and institutions have a discount rate ρ. The exogenous aggregate state of the economy is indexed by $z_t \in \mathcal{Z}$, which follows a continuous time Markov chain with transition matrix Σ.

Match utility: If an agent is unmatched (unemployed), they get flow utility b. Agents match with institutions but not with each other. If an agent of type x is matched with an institution of type y, then they generate transferable utility $F(x, y, z)$, where F is increasing in each variable and twice differentiable with uniformly bounded first partial derivatives on $\mathcal{Z} \times \mathcal{X} \times \mathcal{Y}$. Matches are destroyed at exogenous rate $\delta(x, y, z)$ that potentially depends upon the match and the aggregate state.

Distributions: Let $g_t^w(x)$ denote the population function of agents. Let $g_t^f(y)$ denote the population function of institutions. For expositional simplicity, here we focus on the case with an exogenous and time invariant g^w and an entry condition that determines g_t^f.[2] Formally, following Hagedorn et al. (2017), new firms can enter the economy with a draw of y from the uniform distribution $U(0, 1)$. They pay a flow cost c per period while operating in the economy. Let $g_t(x, y)$ denote the function of matched workers. Let $g_t^e(x)$ denote the function of employed agents. Let $g_t^u(x)$ denote the function of unemployed agents. Let $g_t^p(y)$ denote the function of producing institutions. Let $g_t^v(y)$ denote the function of vacant institutions. The relationships between the densities are given in Table 2 below. We define the aggregate agent employment by $\mathcal{E}_t := \int g_t^e(x)dx$, aggregate agent unemployment by $\mathcal{U}_t := \int g_t^u(x)dx$,

[2]Our method can also handle endogenous g_t^w. For example, in Section 5, we solve an OTC model with investor type switching.

aggregate producing institutions by $\mathcal{P}_t := \int g_t^p(y)dy$, and aggregate vacant institutions by $\mathcal{V}_t := \int g_t^v(y)dy$. Later we will show we can calculate all densities from g_t and so (z_t, g_t) is a sufficient aggregate state space for the economy.

Description	Function	Conditional Density
Matches	$g_t(x,y)$	
Employed workers	$g_t^e(x) = \int g_t(x,y)dy$	$g_t^e(x)/\mathcal{E}_t$
Unemployed workers	$g_t^u(x) = g_t^w(x) - g_t^e(x)$	$g_t^u(x)/\mathcal{U}_t$
Producing firms	$g_t^p(y) = \int g_t(x,y)dx$	$g_t^p(y)/\mathcal{P}_t$
Vacant firms	$g_t^v(y) = g_t^f(y) - g_t^p(y)$	$g_t^v(y)/\mathcal{V}_t$

Table 2: Summary of distributions

Search and Matching Technology: Only and all unmatched agents engage in random search. We generalize to include "on-the-job" search in Section 4. A function $m : \mathbb{R}^+ \times \mathbb{R}^+ \to \mathbb{R}^+, (\mathcal{U}_t, \mathcal{V}_t) \mapsto m(\mathcal{U}_t, \mathcal{V}_t)$ takes the current level of unemployment and vacancies and generates meetings. The rate at which a worker meets a potential institution is given by $\mathcal{M}_t^u := m(\mathcal{U}_t, \mathcal{V}_t)/\mathcal{U}_t$, while the rate at which a vacant firm meeting a potential hire is $\mathcal{M}_t^v := m(\mathcal{U}_t, \mathcal{V}_t)/\mathcal{V}_t$. The rate at which that an agent meets any institution $y \in Y \subset \mathcal{Y}$ equals $\mathcal{M}_t^u(\int_Y (g_t^v(y)/\mathcal{V}_t)dy)$, where $g_t^v(y)/\mathcal{V}_t$ is the density conditional on being vacant. The rate at which an institution meets any worker $x \in X \subset \mathcal{X}$ equals $\mathcal{M}_t^v(\int_X (g_t^u(x)/\mathcal{U}_t)dx$, where $g_t^u(y)/\mathcal{U}_t$ is the density conditional on being unemployed.

Surplus division: We impose that agents negotiate according to a generalized Nash Bargaining protocol so that agents get a fraction β of surplus and institutions get the remaining fraction $1-\beta$. The contract is implemented by providing $w(x,y,z,g)$ flow utility to the agent and $f(x,y,z) - w(x,y,z,g)$ flow utility to the institution.

2.2 Recursive Characterization of Equilibrium

We now define and characterize a recursive equilibrium. The aggregate states are (z, g), where z is the aggregate productivity and g is the distribution of matches.[3]

[3]The mean field game literature has studied the mathematical difficulties involved in defining a recursive equilibrium with an infinite dimensional state (e.g. Cardaliaguet, Delarue, Lasry, and Lions (2015)). However, there is debate about whether these characterizations are appropriate for

We guess (and later verify) that the law of motion for g takes the form:

$$dg_t(x,y) = \mu^g(x,y,z,g)dt.$$

2.2.1 Surplus Division

Let $V^u(x,z,g)$ denote the value of unemployment for a worker of type x, and $V^e(x,y,z,g)$ denotes the value of worker of type x employed at an institution of type y. Let $V^v(y,z,g)$ denote the value of a vacancy for firm y, and $V^p(x,y,z,g)$ denotes the value of firm y employing a worker of type x. The surplus of a match is defined as:

$$S(x,y,z,g) := V^p(x,y,z,g) - V^v(y,z,g) + V^e(x,y,z,g) - V^u(x,z,g)$$

The Nash Bargaining protocol implies that the division of surplus is given by:

$$\begin{aligned} \beta S(x,y,z,g) &= V^e(x,y,z,g) - V^u(x,z,g) \\ (1-\beta)S(x,y,z,g) &= V^p(x,y,z,g) - V^v(y,z,g) \end{aligned} \tag{2.1}$$

These equations must implicitly determine the transfer or "wage" $w(x,y,z,g)$. As in other papers, we assume that contract terms are indexed to the contracts of new hires so that $V^e(x,y,z,g) - V^u(x,z,g)$ is the same for all agents with a particular (x,y), so is $V^p(x,y,z,g) - V^v(y,z,g)$ for all institutions.

2.2.2 Agent Hamilton-Jacobi-Bellman Equations (HJBEs)

Suppose that agents believe that the evolution of the match distribution g is characterized by function $\tilde{\mu}^g(x,y,z,g)$. Given beliefs, the value functions V^u, V^e, V^v, and V^p satisfy the Hamilton Jacobi Bellman Equations (HJBEs):

$$\begin{aligned} \rho V^u(x,z,g) = b + \mathcal{M}^u \int \alpha(x,\tilde{y},z,g)(V^e(x,\tilde{y},z,g) - V^u(x,z,g))\frac{g^v(\tilde{y})}{\mathcal{V}}d\tilde{y} \\ + \sum_{\check{z}\neq z} \lambda(z,\check{z})(V^u(x,\check{z},g) - V^u(x,z,g)) + \langle D_g V^u, \tilde{\mu}^g \rangle \end{aligned} \tag{2.2}$$

economic models. Numerically, we are always working with a finite type space so the only relevant mathematical question is whether a limit exists as the type space becomes continuous.

$$\rho V^e(x,y,z,g) = w(x,y,z,g) - \delta(x,y,z)(V^u(x,z,g) - V^e(x,y,z,g)) + \sum_{\check{z}\neq z} \lambda(z,\check{z})(V^e(x,\check{z},g) - V^e(x,y,z,g)) + \langle D_g V^e, \tilde{\mu}^g \rangle \tag{2.3}$$

$$\rho V^v(y,z,g) = -c + \mathcal{M}^v \int \alpha(\tilde{x},y,z,g)(V^p(\tilde{x},y,z,g) - V^v(y,z,g))\frac{g^u(\tilde{x})}{\mathcal{U}} d\tilde{x} + \sum_{\check{z}\neq z} \lambda(z,\check{z})(V^v(y,\check{z},g) - V^v(y,z,g)) + \langle D_g V^v, \tilde{\mu}^g \rangle \tag{2.4}$$

$$\rho V^p(x,y,z,g) = F(x,y,z) - w(x,y,z,g) - \delta(V^v(y,z,g) - V^p(x,y,z,g)) + \sum_{\check{z}\neq z} \lambda(z,\check{z})(V^p(x,\check{z},g) - V^p(x,z,g)) + \langle D_g V^p, \tilde{\mu}^g \rangle \tag{2.5}$$

where $D_g V^j$ is the Frechet derivative of V^j with respect to the distribution g, $\langle f(y), h(y) \rangle = \int f(y)h(y)dy$ is the inner product, and α is an indicator function for the optimal match acceptance decision. In equilibrium, α is given by:

$$\alpha(x,y,z,g) := \begin{cases} 1, & \text{if } S(x,y,z,g) > 0 \\ 0, & \text{otherwise} \end{cases} \tag{2.6}$$

because under generalized Nash bargaining both agents and institutions accept the match when the surplus is positive.

The HJBE for unemployed agents can be interpreted in the following way. The left-hand side is the flow value of being unemployed. On the right-hand-side, the first term is the flow utility benefit, the second term is the meeting rate multiplied by the expected gain in a meeting, the third term is the value function shift when the exogenous aggregate state changes, and the final term governs how the value function is impacted by distribution changes. The other HJBEs have a similar interpretation.

2.2.3 Match Distribution Evolution

Given the agent matching decisions, the measure of matches evolves according to:

$$dg_t(x,y) = -\delta(x,y,z)g_t(x,y)dt + \mathcal{M}_t^u g_t^u(x)\alpha(x,y,z,g)\frac{g_t^v(y)}{\mathcal{V}_t}dt$$

Given the state, g, and the firm distribution, g^f, we can recover the other features of the distribution from Table 2. So, the Kolmogorov Forward Equation (KFE) for the

match distribution can be expressed as:

$$\begin{aligned} dg_t(x,y) = & -\delta(x,y,z)g_t(x,y)dt + \frac{m(\mathcal{U}_t,\mathcal{V}_t)}{\mathcal{U}_t\mathcal{V}_t}\alpha_t(x,y)\left(g^w(x) - \int g_t(x,y)dy\right) \\ & \times \left(g^f(y) - \int g_t(x,y)dx\right)dt \\ =: & \ \mu^g(x,y,z,g)dt \end{aligned} \tag{2.7}$$

where the first term is the outflow from the breakup of matches and the second term is the inflow from the create of new matches.

2.2.4 Free Entry and the Firm Distribution

Like in Hagedorn et al. (2017), the firm distribution g_t^f is determined by the "free-entry" condition[4]:

$$0 = \mathbb{E}[V_t^v] = \int V^v(\tilde{y},z,g)d\tilde{y}. \tag{2.8}$$

Combining the free-entry condition with the HJB equations gives:

$$\frac{m(\mathcal{U}_t,\mathcal{V}_t)}{\mathcal{V}_t} = \frac{c}{\int\int \alpha(\tilde{x},\tilde{y},z_t,g_t)\frac{g_t^u(\tilde{x})}{\mathcal{U}_t}(1-\beta)S(\tilde{x},\tilde{y},z_t,g_t)d\tilde{x}d\tilde{y}}$$

where $g_t^u = g_t^w - \int g_t(x,y)dy$ and so the RHS can be computed from g_t and S_t. Conceptually, the matching rate depends upon the average surplus because new institutions enter the model until it is no longer profitable to do so. If the matching function is homothetic in $\mathcal{U}_t$ and $\mathcal{V}_t$ (as is common in the literature), then we have that $\frac{m(z_t,g_t)}{\mathcal{V}_t} = \widehat{m}\left(\frac{\mathcal{V}_t}{\mathcal{U}_t}\right)$ and so we can solve for $\mathcal{V}_t$ explicitly. Otherwise, we can deploy a non-linear solver. Since firm y draws are uniformly distributed, g_t^f is then given by:

$$g_t^f = \mathcal{V}_t + \mathcal{P}_t. \tag{2.9}$$

2.2.5 Equilibrium and Master Equation

Definition 1. A **(recursive) equilibrium** is a collection of functions $\{V^u, V^e, V^v, V^p, w, \alpha, g^f\}$ of the state variables (z,g) such that: (i) given beliefs about the evolution of

[4] As is common in the labor search literature, this makes g_t^f a jump variable. Our approach could be extended to consider alternative entry arrangements for which g_t^f is a state variable.

g_t, $(V^u, V^e, V^v, V^p, \alpha)$ solve the HJB equations (2.2)-(2.5), (ii) the division of surplus satisfies (2.1), (iii) g^f satisfies the free entry condition (2.8), and (iv) agent beliefs about the evolution of g_t are consistent in the sense that $\tilde{\mu}^g = \mu^g$, where μ^g is given by equation (2.7).

After combining the HJB equations and imposing belief consistency, the equilibrium can be characterized by the "master equation" for the surplus:

$$\begin{aligned} 0 = \mathcal{L}^S S =: & -\rho S(x,y,z,g) + F(x,y,z) - \delta(x,y,z)S(x,y,z,g) \\ & - (1-\beta)\frac{m(z,g)}{\mathcal{V}(z,g)} \int \alpha(\tilde{x},y,z,g) S(\tilde{x},y,z,g) \frac{g^u(\tilde{x})}{\mathcal{U}(z,g)} d\tilde{x} \\ & - b - \beta \frac{m(z,g)}{\mathcal{U}(z,g)} \int \alpha(x,\tilde{y},z,g) S(x,\tilde{y},z,g) \frac{g^v(\tilde{y})}{\mathcal{V}(z,g)} d\tilde{y} \\ & + \sum_{\check{z} \neq z} \lambda(z)(S(x,y,\check{z},g) - S(x,y,z,g)) + \langle D_g S(x,y,z,g), \mu^g(x,y,z,g) \rangle \end{aligned} \tag{2.10}$$

where μ^g is given by (2.7), $(\mathcal{U}, \mathcal{V}, g^u, g^v)$ can be calculated by Table 2, g^f comes from equation (2.9), and α is given by equation (2.6). Once we obtain $S(x,y,z,g)$ and $\alpha(x,y,z,g)$ by solving Equation (2.10), we can obtain $\{V^u, V^e, V^v, V^p, w\}$ by solving equations (2.2), (2.3), (2.4), (2.5).

2.2.6 Relation to Environments in Other Papers

Block recursivity: We can compare this setup to well-known papers in the search literature with block recursive equilibria in which agents' decisions do not depend upon the distribution of matches. Lise and Robin (2017) sets $\beta = 0$, introduces a vacancy creation condition at each y, and assumes that all unmatched vacancies will be destroyed. They show this implies that α and S do not depend upon g:[5]

$$\alpha(x,y,z,g) = \alpha(x,y,z), \quad S(x,y,z,g) = S(x,y,z)$$

which means the Lise and Robin (2017) model is block recursive in total surplus and acceptance decisions. In Appendix E.2, we show how our setup can nest the Lise and Robin (2017) model as a special case with additional assumptions on the bargaining process and free entry condition. Menzio and Shi (2011) has one-sided heterogeneity, competitive search, and free firm entry. They show this implies that Surplus does not

[5]They also introduce on-the-job search, which we compare to in Section 4.

depend upon g:

$$S(x, y, z, g) = S(x, y, z)$$

and so their model also has "block-recursivity". The goal of our paper is remove the assumptions that lead to block recursivity and solve for α and S explicitly as a function of g.

Dimension reduction: For models with incomplete but competitive markets, Krusell and Smith (1998) suggests replacing the law of motion for the distribution by the law of motion of its mean (and potentially the law of motion of other low dimensional moments). This is an appealing approach for some competitive market models because the distribution impacts agents' decisions by changing aggregate prices and aggregate prices may primarily depend upon the mean of the distribution. By contrast, in a search and matching model, the distribution impacts agents' decisions by changing the probability distribution over which type of agent they meet. This ultimately enters the master equation on lines 3 and 4 of equation (2.10). There are no ex-ante obvious low-dimensional moments of the distribution that are sufficient for evaluating these terms. Instead, we need to integrate across the surplus function, weighted by the acceptance decision and the density of searching agents.

2.2.7 Model Extensions

In Section 4, we extend the model to incorporate endogenous separation and on-the-job search with positive worker bargaining power. In Section 5, we extend the model to incorporate type switching and asset trade in an over-the-counter market. Our solution approach offers a foundation that could be used to study other important models in the search and matching literature such as within-firm heterogeneity for large firms, multi-dimensional sorting, non-transferable utility, and consumption–saving decisions.

2.3 Approximation with Finite Types

Our goal is to solve the master equation (2.10) numerically to obtain $S(x, y, z, g)$ and $\alpha(x, y, z, g)$. Then we could solve for the value and wage functions using Equations

(2.2) to (2.5). The difficulty of solving Equation (2.10) is that the state space contains an infinite dimensional distribution, g, and so the master equation contains Frechet derivatives with respect to the distribution. To make progress on this problem, we discretize the type space so that equation (2.10) becomes a high, but finite dimensional partial differential equation that can be solved using deep learning.

Discrete type space and KFE: We restrict the possible types to finite collections: $x \in \mathcal{X} = \{x_1, \ldots, x_{n_x}\}$ and $y \in \mathcal{Y} = \{y_1, \ldots, y_{n_y}\}$. With some abuse of notation, we let $\underline{\boldsymbol{g}}_t$ denote the vector of measures of matched agents at the points $(\mathcal{X}, \mathcal{Y})$, where $g_{t,ij} = g_t(x_i, y_j)$ is the function at type (x_i, y_j). We also let $g^w_{t,i} := g^w_t(x_i)$ and $g^f_{t,j} := g^f_t(y_j)$. The aggregate state variables are now: $\{z, \underline{\boldsymbol{g}}_t\}$, the aggregate productivity and the density vector of matched agents. Under this discretization, the Riemann approximation to the KFE is given by:

$$\begin{aligned} dg_{t,ij}/dt &= \mu^g(x_i, y_j, z_t, \underline{\boldsymbol{g}}_t) \\ &= -\delta(x_i, y_j, z) g_{t,ij} + \frac{m(z_t, \underline{\boldsymbol{g}}_t)}{\mathcal{U}(z_t, \underline{\boldsymbol{g}}_t)\mathcal{V}(z_t, \underline{\boldsymbol{g}}_t)} \alpha(x_i, y_j, z_t, \underline{\boldsymbol{g}}_t) \\ &\quad \times \left(g^w_{t,i} - \frac{1}{n_y} \sum_{k=1}^{n_y} g_{t,ik} \right) \left(g^f_{t,j} - \frac{1}{n_x} \sum_{l=1}^{n_x} g_{t,lj} \right), \qquad \forall i \leq n_x, j \leq n_y \end{aligned} \tag{2.11}$$

Master equation: The discretized Master equation for the Surplus is given by:

$$\begin{aligned} 0 = \mathcal{L}^S S &= -(\rho + \delta(x_i, y_j, z)) S(x, y, z, \underline{\boldsymbol{g}}) + F(x_i, y_j, z) - b \\ &\quad - (1-\beta) \frac{m(z_t, \underline{\boldsymbol{g}})}{\mathcal{U}(z, \underline{\boldsymbol{g}})\mathcal{V}(z, \underline{\boldsymbol{g}})} \frac{1}{n_x} \sum_{k=1}^{n_x} \alpha(x_k, y_j, z, \underline{\boldsymbol{g}}) S(x_k, y_j, z, \underline{\boldsymbol{g}}) g^u_k \\ &\quad - \beta \frac{m(z_t, \underline{\boldsymbol{g}})}{\mathcal{U}(z, \underline{\boldsymbol{g}})\mathcal{V}(z, \underline{\boldsymbol{g}})} \frac{1}{n_y} \sum_{l=1}^{n_y} \alpha(x_i, y_l, z, \underline{\boldsymbol{g}}) S(x_i, y_l, z, \underline{\boldsymbol{g}}) g^v_l \\ &\quad + \sum_{k=1}^{n_x} \sum_{l=1}^{n_y} \partial_{g_{kl}} S(x_k, y_l, z, \underline{\boldsymbol{g}}) \mu^g(x_k, y_l, z, \underline{\boldsymbol{g}}) \\ &\quad + \sum_{\check{z} \neq z} \lambda(z) (S(x_i, y_j, \check{z}, \underline{\boldsymbol{g}}) - S(x_i, y_j, z, \underline{\boldsymbol{g}})), \qquad \forall\ i \leq n_x, j \leq n_y \end{aligned} \tag{2.12}$$

where μ^g is given by the discretized KFE (2.11), $(\mathcal{U}, \mathcal{V}, g^u, g^v)$ can be calculated by Table 2 (after appropriate discretization), g^f comes from equation (2.9), and we

approximate $\alpha(x, y, z, \underline{\boldsymbol{g}})$ by:

$$\alpha(x, y, z, \underline{\boldsymbol{g}}) = \left(1 + e^{-\xi S(x,y,z,\underline{\boldsymbol{g}})}\right)^{-1}$$

to ensure differentiability of the value function when there is a finite number of types.[6]

2.4 The DeepSAM Method

In this section, we present the DeepSAM method for using deep learning to solve and calibrate the general class of search and matching models outlined above. We first present, for given structural parameters, how to solve the high dimensional master equation (2.12). We then show how to calibrate the model by introducing structural parameters as pseudo-state variables in the neural network and using a surrogate model to find the parameters that match simulated moments with the data.

2.4.1 Solution Algorithm

We start by outlining our algorithm for solving the discretized master equation (2.12) when the structural parameters are given. Let $\boldsymbol{\omega} = (x, y, z, \underline{\boldsymbol{g}}) \in \Omega^{\omega}$ denote the state space. We approximate the surplus function, S, by a neural network, $\hat{S}$, and use $\boldsymbol{\Theta} \in \Omega^{\Theta}$ to denote the parameters of the neural network:

$$\hat{S}: \ \Omega^{\omega} \times \Omega^{\Theta} \to \mathbb{R}, \quad (\boldsymbol{\omega}, \boldsymbol{\Theta}) \mapsto \hat{S}(\boldsymbol{\omega}; \boldsymbol{\Theta})$$

with the form:

$$\begin{aligned}
\boldsymbol{h}^{(1)} &= \phi^{(1)}(W^{(1)}\boldsymbol{\omega} + \boldsymbol{b}^{(1)}) && \ldots \text{Hidden layer 1} \\
\boldsymbol{h}^{(2)} &= \phi^{(2)}(W^{(2)}\boldsymbol{h}^{(1)} + \boldsymbol{b}^{(2)}) && \ldots \text{Hidden layer 2} \\
&\ldots \\
\boldsymbol{h}^{(H)} &= \phi^{(H)}(W^{(H)}\boldsymbol{h}^{(H-1)} + \boldsymbol{b}^{(H)}) && \ldots \text{Hidden layer H} \\
\hat{S} &= \sigma(\boldsymbol{h}^{(H)}) && \ldots \text{Surplus}
\end{aligned}$$

where, using the terminology of the deep learning literature, H is referred the number of hidden layers, the length of vector $\boldsymbol{h}^{(i)}$ is referred to as the number of neurons in

[6]The "softened" α function can be interpreted as a logit choice model where utility shocks come from an extreme value distribution with parameter ξ.

hidden layer i, $\phi^{(i)}$ is referred to as the activation function for hidden layer i, and the collection $\mathbf{\Theta} = (W^1, \dots W^{(H)}, b^{(1)}, \dots, b^{(H)})$ are the parameters for the neural network.

Algorithm 1: Generic Solution Algorithm

1. Approximate the surplus function by the neural network $S(x, y, z, g) \approx \widehat{S}(x, y, z, g; \mathbf{\Theta})$.
2. Start with initial parameter guess $\mathbf{\Theta}^0$.
3. At iteration n with $\mathbf{\Theta}^n$:
   (a) Generate K sample points, $Q^n = \left\{\left(x_k, y_k, z_k, \{g_{ij,k}\}_{i \le n_x, j \le n_y}\right)\right\}_{k \le K}$.
   (b) Calculate the average mean squared error of the surplus master equation (2.12) on the sample points:
   $$L\left(\mathbf{\Theta}^n, Q^n\right) := \frac{1}{K} \sum_{k \le K} \left| \mathcal{L}^S \widehat{S}\left(x_k, y_k, z_k, \{g_{ij,k}\}_{i \le n_x, j \le n_y}\right)\right|^2$$
   (c) Update NN parameters using stochastic gradient descent (SGD) or a variant. For example by $\mathbf{\Theta}^{n+1} = \mathbf{\Theta}^n - \zeta^n \nabla_{\mathbf{\Theta}} L\left(\mathbf{\Theta}^n, Q^n\right)$.
   (d) Repeat until $L\left(\mathbf{\Theta}^n, Q^n\right) \le \epsilon$ with precision threshold ϵ.
4. Once S is solved, we have α and can solve for worker and firm value functions.

Our goal is to train the parameters of the neural network to approximately solve equation (2.12) globally across the state space. Our approach is summarized in Algorithm 1. Essentially, we use the stochastic gradient descent algorithms or their variants to train the neural network to minimize the average loss in the master equation on a random collection of sample points. As with other neural network approaches, there are many implementation details involved with these generic steps.

Sampling procedure: We first solve the model at the steady state for the different fixed z. We then draw distributions that are perturbed random combinations of the steady state distributions for the different z. This sampling approach uses economic knowledge to increase the solution accuracy in the regions of the state space that are of most interest for the economic question. If required, once the error is small, we can then move to sampling from the ergodic distribution generated by the current

solution. We can also increase sampling in regions of the state space (x, y) where errors are high.

Algorithm stability It is most difficult to stabilize the algorithm when $\widehat{S}(x, y, z, g; \mathbf{\Theta})$ has sharp curvature. In this case, we use a "homotopy" approach (Azinovic and Žemličkа, 2023). Step (1): Train NN for parameters that give low curvature in $\widehat{S}^1$. Step (2): Change parameters closer and retrain NN starting from previous $\widehat{S}^2 = \widehat{S}^1$. Step (3)+: keep changing parameters and retraining until at desired parameters.

2.4.2 DeepSAM for Solution and Calibration

A key goal of quantitative macroeconomics is to calibrate or estimate structural model parameters so that the model generates output that is consistent with observed data moments. Traditionally, this requires solving the model repeatedly for different parameter values, which can be a computationally intensive process. Deep learning based solution methods offer an advantage for model calibration. By introducing economic parameters as pseudo state variables in the neural network, we can efficiently solve models simultaneously across both the state space and the economic parameter space. We can then calibrate the model parameters using the method of simulated moments.

In this section, we outline our algorithm for solving and calibrating the model. Let $\mathbf{\Psi} \in \Omega^{\Psi}$ denote the structural parameters that will be calibrated internally. Let $\hat{\varphi} = (\hat{\varphi}_1, \ldots, \hat{\varphi}_N)$ denote the $N \times 1$ data moments that we want to match. Let $\varphi(\mathbf{\Psi}) = (\varphi_1(\mathbf{\Psi}), \ldots, \varphi_N(\mathbf{\Psi}))$ be the corresponding model moments given structural parameters $\mathbf{\Psi}$, which we will generate by simulating the model. Our goal is to calibrate the model by choosing parameters $\mathbf{\Psi}$ to solve:

$$\hat{\mathbf{\Psi}} = \arg\min_{\mathbf{\Psi}} \sum_{i=1}^{N} \omega_i \left(\frac{\hat{\varphi}_i - \varphi_i(\mathbf{\Psi})}{\hat{\varphi}_i} \right)^2, \tag{2.13}$$

where ω_i are fixed weights.

The main computational challenge of calibration is to solve $\varphi_i(\mathbf{\Psi})$ for many values of the structural parameters. We overcome this challenge by introducing $\mathbf{\Psi}$ as a pseudo state vector and solving the resulting extended master equation, which in its

discretized form is given by:

$$
\begin{aligned}
0 = \mathcal{L}^S S = &-(\rho + \delta(x_i, y_j, z))S(x, y, z, \underline{\boldsymbol{g}}, \boldsymbol{\Psi}) + F(x_i, y_j, z) - b \\
&- (1-\beta)\frac{m(z_t, \underline{\boldsymbol{g}})}{\mathcal{U}(z, \underline{\boldsymbol{g}})\mathcal{V}(z, \underline{\boldsymbol{g}})}\frac{1}{n_x}\sum_{k=1}^{n_x}\alpha(x_k, y_j, z, \underline{\boldsymbol{g}}, \boldsymbol{\Psi})S(x_k, y_j, z, \underline{\boldsymbol{g}}, \boldsymbol{\Psi})g_i^u \\
&- \beta\frac{m(z_t, \underline{\boldsymbol{g}})}{\mathcal{U}(z, \underline{\boldsymbol{g}})\mathcal{V}(z, \underline{\boldsymbol{g}})}\frac{1}{n_y}\sum_{l=1}^{n_y}\alpha(x_i, y_l, z, \underline{\boldsymbol{g}}, \boldsymbol{\Psi})S(x_i, y_l, z, \underline{\boldsymbol{g}}, \boldsymbol{\Psi})g_j^v \\
&+ \sum_{k=1}^{n_x}\sum_{l=1}^{n_y}\partial_{g_{kl}}S(x_k, y_l, z, \underline{\boldsymbol{g}}, \boldsymbol{\Psi})\mu^g(x_k, y_l, z, \underline{\boldsymbol{g}}, \boldsymbol{\Psi}) \\
&+ \sum_{\check{z}\neq z}\lambda(z)(S(x_i, y_j, \check{z}, \underline{\boldsymbol{g}}, \boldsymbol{\Psi}) - S(x_i, y_j, z, \underline{\boldsymbol{g}}, \boldsymbol{\Psi})), \qquad \forall\ i \leq n_x, j \leq n_y
\end{aligned}
$$

We approximate the extended surplus function S by a neural network:

$$\hat{S}:\ \Omega^{\omega} \times \Omega^{\Psi} \times \Omega^{\Theta} \to \mathbb{R}, \quad (\boldsymbol{\omega}, \boldsymbol{\Psi}, \boldsymbol{\Theta}) \mapsto \hat{S}(\boldsymbol{\omega}, \boldsymbol{\Psi}; \boldsymbol{\Theta}).$$

and solve it in a similar way as for equation (2.12), as outlined in Algorithm 2. Compared to the neural network for solving equation (2.12), now the structural parameter vector $\boldsymbol{\Psi}$ is also part of the state space. Compared to traditional techniques, an advantage of deep learning is that solving the extended master equation remains feasible even after expanding the state space.

We now use the solution to the extended master equation to build a surrogate model that maps economic parameters to simulated moments:

$$\Phi : \Omega^{\Psi} \times \boldsymbol{\Theta}^{\Phi} \to \mathbb{R}^N, \quad \Psi \mapsto \Phi(\Psi; \boldsymbol{\Theta}^{\Phi})$$

We do this by simulating the model under many different structural parameter vectors $\{\boldsymbol{\Psi}_l\}_l$ (using parallelized processing) and computing the resulting model moments $\{\varphi(\boldsymbol{\Psi}_l) = (\varphi_1(\boldsymbol{\Psi}_l), \ldots, \varphi_N(\boldsymbol{\Psi}_l))\}_l$. We then construct the surrogate model by training an additional neural network to approximate the relationship $\Phi(\Phi)$ using the simulated data. With $\Phi(\boldsymbol{\Psi})$, we can obtain the calibrated parameters by solving the optimization problem (2.13). Our approach is summarized in Algorithm 2.

Algorithm 2: Generic Solution and Calibration Algorithm

1. Train the model across the state and structural parameter spaces:
   (a) Approximate the surplus function by a neural network that includes the economic parameters $\mathbf{\Psi}$ as inputs $S(x, y, z, g, \mathbf{\Psi}) \approx \widehat{S}(x, y, z, g, \mathbf{\Psi}; \mathbf{\Theta})$.
   (b) Train the neural network, as in Algorithm 1.
2. Compute a surrogate model mapping structural parameters to model moments $\Omega^{\Psi} \times \Omega^{\Phi} : (\Psi, \mathbf{\Theta}^{\Phi}) \mapsto \Phi(\Psi; \mathbf{\Theta}^{\Phi})$ and optimize over Φ to solve the distance minimization problem (2.13).

3 Labor Market With Two-Sided Heterogeneity

In this section, use our method a simple search and matching model with two-sided heterogeneity and aggregate crisis shocks. Conceptually, this model extends Shimer and Smith (2000) by incorporating aggregate shocks and Mortensen and Pissarides (1994) by introducing heterogeneity among workers and firms. The goal of this Section is three-fold: first, we illustrate our DeepSAM method using a canonical labor setup; second, we evaluate its numerical performance, highlighting its accuracy and computational efficiency in addressing high-dimensional problems; lastly, we use the model to study the impact of the COVID-19 shock and disentangle the importance of distribution feedback in aggregate dynamics.

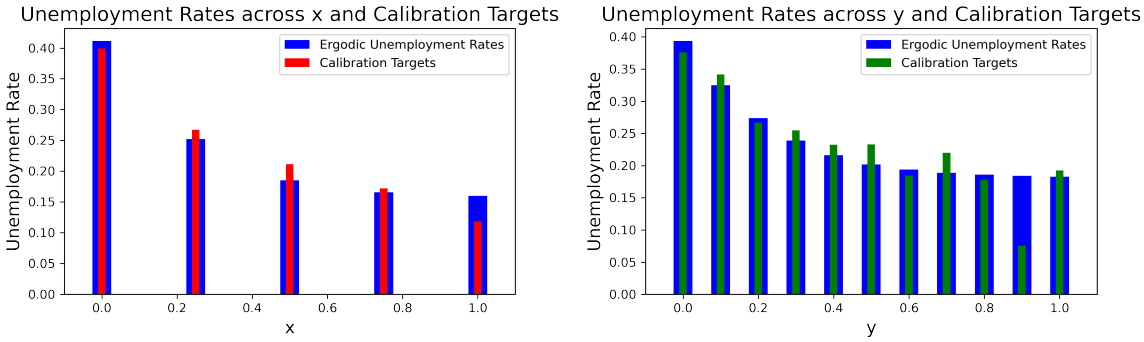
In this section, we use DeepSAM to solve this model. In Section 4, we implement a more comprehensive deep learning based calibration in a richer setup with on-the-job search and endogenous separation to align the model more closely with a broader set of data moments.

3.1 Model Details and Parameters

Our environment is a special case of Section 2.1 with the following features. The agents are interpreted as "workers" and the institutions as "firms". The exogenous aggregate state z follows a three-state Markov chain, corresponding to expansion (z_H), normal recession (z_L), and major crisis (z_D), the later of which we interpret as COVID-19. Match output $F(z, x, y) = A_z f(x, y)$ and the separation rate $\delta_z(x, y)$ both depend upon the state z, where we have adopted the notation that A and δ are

indexed by z.

Economic parameters: The calibration of the economic parameters for our model is presented in Table 6 in Appendix A. We calibrate the model at the annual frequency. For non-crisis states, when possible, we take standard parameters from the literature. We calibrate the matching efficiency κ to target an ergodic average unemployment rate of 6%. For the crisis state, we calibrate the separation rate across workers and firms $\delta_D(x, y)$ to match the observed peak declines in employment levels during the COVID-19 recession calculated in Cajner et al. (2020). More specially, Cajner et al. (2020) uses detailed data from a major US payroll company to estimate the employment drop of workers in different skill groups (corresponding to five groups in our model) and firms in different two-digit NAICS industries (mapped to 11 groups in our model) during the COVID-19 recession. These drops peaked in April 2020, about 0.2 years after the onset of the pandemic in the US. As shown in Figure 1, we calibrate $\delta_D(x, y)$ such that the model's simulated declines in employment for these heterogeneous groups match these empirical moments after the disaster shock z_D hits the ergodic state of the economy for $t = 0.2$ years. The detailed values of $\delta_D(x, y)$ are in Appendix D.1.



(a) Calibration along worker dimension

(b) Calibration along firm dimension

Figure 1: Calibrating $\delta_D(x, y)$ to match effect of COVID-19 shock on heterogeneous workers and firms in Cajner et al. (2020).

Neural network parameters: We describe the details of the neural network approximation and sampling in Table 7 in Appendix A. We use a fully connected feed-forward network with 4 layers, 50 neurons per layer, and a $tanh(\cdot)$ activation function.

3.2 Numerical Performance

In this subsection, we evaluate the numerical performance of the DeepSAM method on the model presented above. We perform a number of checks to verify the usefulness of our solution method. We report the numerical loss and computational time for solving the model in the first column of Table 3. We achieve small errors, on the order of $\mathcal{O}(10^{-6})$, across our sample. This level of accuracy was reached within 4 hours and 20 minutes using an A100 GPU on Google Colab, a platform accessible to all researchers. Notably, no existing method has been able to solve our 58-dimensional PDE within such a timeframe. We discuss the detailed numerical scheme and the numerical stability of the results in Appendix D.3.[7]

	Model with agg. shock	Model without agg. shock
PDE Training Loss	2×10^{-6}	3.9×10^{-6}
MSE to Existing Solution	No existing solution	5×10^{-6}
Computational Time	4h 20min	57 min

Table 3: Numerical performance of DeepSAM for models with and without aggregate shocks. Computations are performed on the A100 GPU at Google Colab.

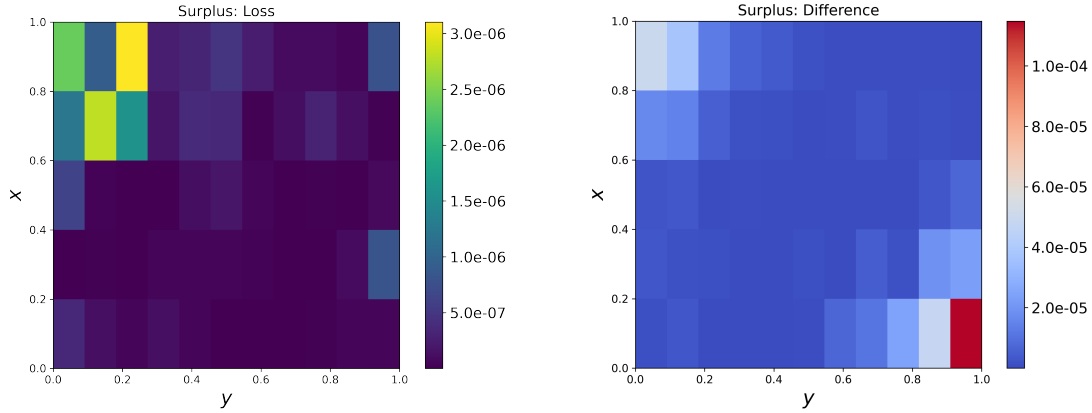
Regarding accuracy, besides verifying low numerical loss, we can further validate our method by applying it to a model that can be solved with conventional methods. For this purpose, we apply the DeepSAM method to a model without aggregate shocks—a 57-dimensional PDE for $S(x, y, \underline{\boldsymbol{g}})$, which we fully specify in Appendix D—by imposing $z_t \equiv \bar{z}$ and $\delta_t \equiv \bar{\delta}$. For this model, we can use traditional fixed-point methods, as in Shimer and Smith (2000) and Hagedorn, Law, and Manovskii (2017), to solve for the deterministic steady state (DSS) solution $S^{\text{DSS}}(x, y)$, which can serve as a benchmark to validate our approach. With the 57-dimension solution we obtain with DeepSAM, we can set $\underline{\boldsymbol{g}} = \underline{\boldsymbol{g}}^{DSS}$ and get the DeepSAM solution at DSS as

$$S^{\text{DSS}}_{\text{DeepSAM}}(x, y) = S(x, y, \underline{\boldsymbol{g}} = \underline{\boldsymbol{g}}^{DSS})$$

Then we can compare our DSS solution with the solution using conventional methods

[7]The computation time varies with different calibrations. For example, if we set a relatively small $\kappa = 0.4$, it only takes less than 30 minutes to solve the 58-dimensional PDE. That's because the high dimensional function approximated by neural networks is flatter in the curvature, which makes it easier to "learn". Also, the disaster state makes it much more time-consuming to solve the model. In Section 4, it takes us 55 minutes to solve the model with on-the-job search and endogenous separation, but without the disaster state.

$S^{\text{DSS}}_{\text{Conventional}}(x, y)$. We define the squared difference of the two methods for each (x, y) pair as $\|S^{\text{DSS}}_{\text{DeepSAM}}(x, y) - S^{\text{DSS}}_{\text{Conventional}}(x, y)\|^2$. The mean squared difference takes the average of the squared difference across all (x, y) pairs. We report the numerical performance in the second column of Table 3. The PDE training loss, measured as the mean square error, averaged 3.9×10^{-6} over the 57-dimensional state space $(x, y, \underline{\boldsymbol{g}})$ after 57 minutes of training. By comparing our solution at this steady state with those obtained using conventional methods, we found differences in the order of 10^{-5}. We interpret these results as indicative of high accuracy.



(a) Model with aggregate shock: loss across state space

(b) Model without aggregate shock: difference from conventional solution

Figure 2: Numerical accuracy across state space.

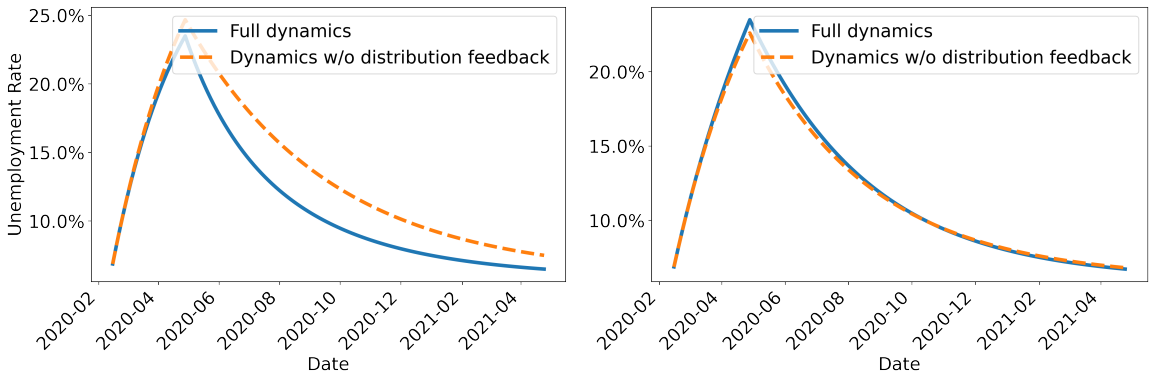
We also depict the numerical accuracy visually. For the model with aggregate shocks, Figure 2a shows the mean squared loss in the surplus Master equation at a given distribution and aggregate shock realization. The loss is in the order of magnitude of 10^{-6} and not biased in a particular part of the state space. For the model without aggregate shocks, Figure 2b plots the squared difference between the DeepSAM solution and conventional solution at the DSS. The difference is always in the order of magnitude of 10^{-5}, which we also interpret as high accuracy.

3.3 Distribution Feedback to Aggregate Dynamics: The COVID-19 Recession

An important advantage of the DeepSAM method is its ability to explicitly solve for α as a function of the distribution, enabling us to assess the contribution of distribution

feedback to aggregate dynamics. This is particularly valuable in scenarios where there is a significant shift in the distribution of matches over business cycles - a phenomenon well-documented in a large empirical literature, such as by Guvenen, Schulhofer-Wohl, Song, and Yogo (2017). In this section, we use our model to study the impact of the COVID-19 shock, and disentangle the importance of distribution feedback in aggregate dynamics. These are important questions that can only be answered using the global solution methods.

As discussed above, we model the COVID-19 shock as the disaster shock z_D that lasts for $t = 0.2$ years, followed by the recovery phase with stochastic aggregate shocks $z_t \in \{z_H, z_L, z_D\}$. The model is globally solved with the DeepSAM method. With the solution in place, we can simulate the dynamics of the g_t using the Kolmogorov forward equation (2.11) and compute aggregate dynamics for unemployment, employment, average wage, and other variables with g_t. The dynamics of aggregate unemployment are presented as the blue line of Figure 3a. As is targeted in the data, we see a sharp increase in unemployment until April 2020, followed by the recovery phase.



(a) $\mathcal{U}_t$ after COVID shock (b) $\mathcal{U}_t$ after counterfactual symmetric shock

Figure 3: Unemployment $\mathcal{U}_t$ after the COVID-19 shock: full dynamics vs "restricted" dynamics in which the decision of workers and firms are myopic to change in distribution g_t as in Equation (3.1). The left panel shows the dynamics after the true asymmetric COVID-19 shock, while the right one corresponds to the counterfactual symmetric shock.

To understand the contribution of distribution feedback to aggregate dynamics, we decompose the impulse responses in the unemployment into two channels:

(i) the change in unemployment when agents' acceptance decision is always evalu-

ated at the long-run ergodic employment distribution, and

(ii) the additional change in unemployment when the acceptance function reacts to the changing matching distribution.

We refer to the former as the "restricted" dynamics for the experiment without the additional feedback from the distribution to the acceptance function. Mathematically, under the restricted dynamics, the distribution g_t^R evolves according to:

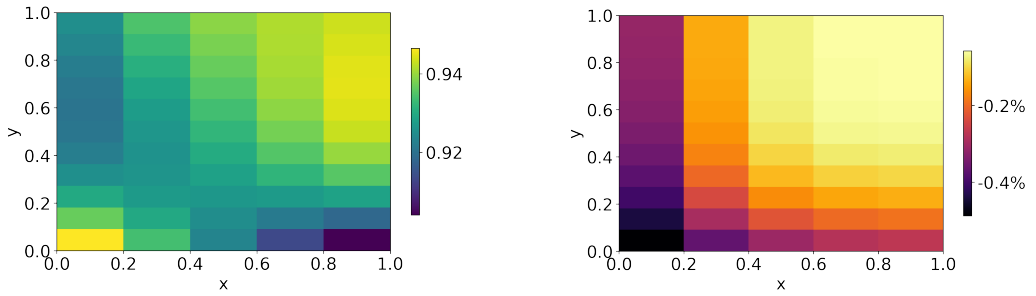
$$\frac{dg_t^R(x,y)}{dt} = -\delta(x,y,z_t)g_t^R(x,y) + \frac{m_t(z,\underline{\boldsymbol{g}}_t^R)}{\mathcal{U}_t(\underline{\boldsymbol{g}}_t^R)\mathcal{V}_t(\underline{\boldsymbol{g}}_t^R)}\alpha(x,y,z_t,\underline{\boldsymbol{g}}^{\text{ergodic}})g_t^{u,R}(x)g_t^{v,R}(y) \quad (3.1)$$

We plot the restricted dynamics with the dashed orange line of Figure 3a. Compared to the full dynamics calculated with the KFE (2.11), the only difference in Equation (3.1) is that agents make decisions assuming the distribution is always at the erdogic state. Thus the gap between full dynamics and restricted dynamics can be interpreted as the contribution of distribution feedback to aggregate dynamics.

From Figure 3a, we can see the distribution feedback accounts for about 30% of the unemployment dynamics in the recovery phase. When agents make decisions based on the true distribution shift, the unemployment rate recovers faster than in the economy where agents are myopic to the distribution change. This divergence is attributable to the asymmetric shock disproportionately increasing unemployment among low-type workers, thereby raising the opportunity cost of waiting for a high-type match. Consequently, in the full model, workers are less selective and more likely to accept matches with low-type partners compared to their counterparts in the model without distribution feedback.

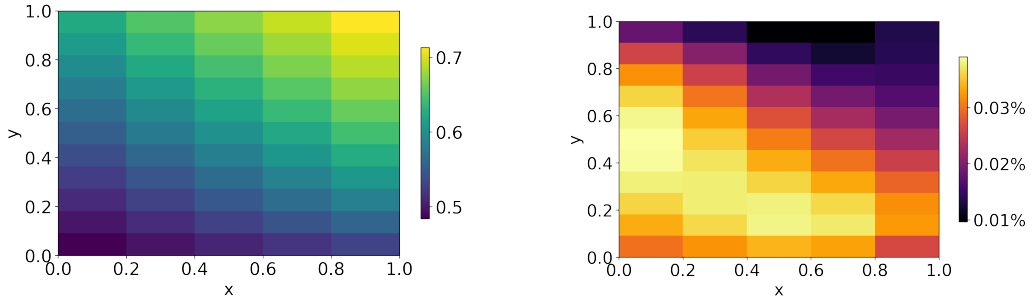
We confirm this mechanism of distribution feedback in Figure 4, which shows how agents' decision function α changes when the distribution shifts from the ergodic steady state to that following the asymmetric COVID-19 shock. Figures 4a and 4b show that matches are disproportionately destroyed at the lower ends of both the worker and firm distributions. Consequently, as shown in Figure 4d, the low type workers and firms exhibit a higher accepting rate for potential matches, as they understand the opportunity cost of waiting is higher due to increased competition in the job market following the asymmetric shock.

To further dissect the role of asymmetric shocks, we conduct a counterfactual analysis with a disaster shock that has symmetric impact across workers and firms



(a) $g^{\text{ergodic}}(x, y)$ at ergodic state

(b) $g^{\text{COVID}}(x, y) - g^{\text{ergodic}}(x, y)$



(c) $\alpha(x, y, g^{\text{ergodic}})$ at ergodic state

(d) $\alpha(x, y, g^{\text{COVID}}) - \alpha(x, y, g^{\text{ergodic}})$

Figure 4: Distribution of matches and associated acceptance decision. Panels (a) and (c): distribution and acceptance at ergodic steady state. Panels (b) and (d): difference of distribution and acceptance after the asymmetric COVID-19 shock, compared to the ergodic steady state.

through the separation rate, namely $\widetilde{\delta}_D(x, y) \equiv \widetilde{\delta}_D$ for all x, y. We calibrate $\widetilde{\delta}_D$ such that the shock generates the same aggregate unemployment rate as our calibrated $\delta_D(x, y)$ at the peak of the COVID-19 recession. We plot the full dynamics and restricted dynamics of the economy after this counterfactual "symmetric" shock in Figure 3b. In this case, the unemployment trajectories of the "restricted" dynamics and the full solution are closely aligned, which means the distribution feedback is small when aggregate shocks affect workers and firms symmetrically.

These findings underscore the significance of distribution feedback on aggregate dynamics in the labor market, particularly when workers and firms have heterogeneous exposure to aggregate shocks. They also suggest that policy design should account for the asymmetric nature of shocks, as well as the heterogeneous effects of policies themselves. For instance, the design of unemployment insurance needs to consider the feedback effect from the distribution shift to fully understand its implications and efficacy.

4 On-The-Job Search and Business Cycles

In this section, we study business cycle shocks in a labor market with endogenous job-to-job and job-to-unemployment transitions. Conceptually, the model extends Lise and Robin (2017) by giving workers non-zero bargaining power and generalizing the free entry condition, both of which are changes that break "block-recursivity". We calibrate the model using the deep learning based simulated method of moments approach outlined in Section 2.4.2. We use the calibrated model to study how employment and wage dynamics vary across the worker and firm types.

4.1 Environment Changes to Section 3.1

Search and matching: All workers now engage in random search. The matching function becomes $m(\mathcal{U}_t + \phi\mathcal{E}_t, \mathcal{V}_t)$ with the interpretation that ϕ is the exogenous relative intensity at which employed workers search. Let $\mathcal{W}_t := \mathcal{U}_t + \phi\mathcal{E}_t$ denote the total mass of searchers. The probabilities that an unemployed or an employed worker meets a potential employer are given by $\mathcal{M}_t^u = \frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{W}_t}$ and $\mathcal{M}_t^e = \phi\frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{W}_t}$ while the probability that a vacant firm meets a potential hire is $\mathcal{M}_t^v = \frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{V}_t}$. Conditional on meeting a worker, we define the probabilities that the worker is unemployed or employed by $\mathcal{C}_u = \frac{\mathcal{U}}{\mathcal{U}+\phi\mathcal{E}}$ and $\mathcal{C}_e = \frac{\phi\mathcal{E}}{\mathcal{U}+\phi\mathcal{E}}$ respectively. The probability for a firm to meet an unemployed worker $x \in \mathcal{X}$ equals $\mathcal{M}_v\mathcal{C}_u \int_X g^u(x)dx$. The probability for a firm to meet an employed worker $x \in \mathcal{X}$ equals $\mathcal{M}_v\mathcal{C}_u \int_X g^e(x)dx$.

Bargaining between unemployed workers and firms: Let $V_t^u(x)$ denote the value of unemployment for a worker of type x. Let $V_t^e(x,y)$ denote the value of worker x employed at a firm of type y. Let $V_t^v(y)$ denote the value of a vacancy for firm y. Let $V_t^p(x,y)$ denote the value of firm y employing an unemployed worker of type x. As before, the surplus of a match between an unemployed worker and an vacant firm is defined as $S_t^u(x,y) := V_t^p(x,y) - V_t^v(y) + V_t^e(x,y) - V_t^u(x)$ and the division of surplus is given by $\beta S_t^u(x,y) = V_t^e(x,y) - V_t^u(x)$ and $(1-\beta)S_t^u(x,y) = V_t^p(x,y) - V_t^v(y)$.

Bargaining on the job: If a worker in match (x,y) moves to another firm with productivity $\tilde{y}$, then the worker gets β share of the incremental surplus $S_t(x,\tilde{y}) - S_t(x,y)$, the new firm gets $1-\beta$ share of the incremental surplus, and the incumbent firm keeps their surplus $(1-\beta)S_t(x,y)$. Conceptually, this is a model where the incumbent firm

cannot be made worse off by the move (e.g. because they have a veto over whether the worker can move), which makes this setup both technically and economically appealing.[8]

Endogenous separation: In addition to allowing workers to search, we also allow matches to breakup. If a worker decides to leave a match, then they return to unemployment at rate η.

4.2 Equilibrium

Agent HJBEs: As before, the aggregate state variables are (z, g), where z is the aggregate productivity and g is the distribution of matches. Let $\tilde{\mu}^g(x, y, z, g)$ denote the agents' belief about the law of motion for the distribution g. Unemployed workers once again have an HJBE given by:

$$\begin{aligned}\rho V^u(x,z,g) = b &+ \mathcal{M}^u \int \alpha(x,\tilde{y},z,g)(V^e(x,\tilde{y},z,g) - V^u(x,z,g))\frac{g^v(\tilde{y})}{\mathcal{V}}d\tilde{y} \\ &+ \sum_{\check{z}\neq z} \lambda(z,\check{z})(V^u(x,\check{z},g) - V^u(x,z,g)) + \langle D_g V^u, \tilde{\mu}^g\rangle\end{aligned}$$

where α is an indicator function for whether workers and firms accept the match, which under generalized Nash Bargaining is again given by (2.6) (positive surplus).

Employed workers now choose whether to accept matches from other firms and whether to leave jobs, which leads to the HJBE:

$$\begin{aligned}\rho V^e(x,y,z,g) = w(x,y,z,g) &+ (\delta(x,y,z) + \eta\alpha^b(x,y,z,g))(V^u(x,z,g) - V^e(x,y,z,g)) \\ &+ \mathcal{M}^e \int \alpha^e(x,y,\tilde{y},z,g)\beta(S(x,\tilde{y},z,g) - S(x,y,z,g))\frac{g^v(\tilde{y})}{\mathcal{V}}d\tilde{y} \\ &+ \sum_{\check{z}\neq z} \lambda(z,\check{z})(V^e(x,\check{z},g) - V^e(x,z,g)) + \langle D_g V^e, \tilde{\mu}^g\rangle\end{aligned}$$

where α^b and α^e are indicator functions for whether a worker or firm chooses to

[8]Our assumption that workers get a share β of incremental surplus from on-the-job transitions is different to Lise and Robin (2017), which imposes that workers get no surplus when moving from unemployment to employment and get the result of Bertrand competition between firms when moving from one job to another. Technically, our setup allows us to relax the $\beta = 0$ restriction in Lise and Robin (2017) while still allowing the problem to be characterized recursively without needing to add match history as a state variable. We also believe our setup makes economic sense because it ensures that the worker has similar bargaining power in all their matches.

breakup the match and whether workers and firms both accept an on-the-job match. Under generalized Nash Bargaining, these satisfy:

$$\alpha^b(x,\tilde{y}) := \begin{cases} 1, & \text{if } S(x,\tilde{y},z,g) < 0 \\ 0, & \text{otherwise} \end{cases}$$

$$\alpha^e(x,y,\tilde{y},z,g) := \begin{cases} 1, & \text{if } S(x,\tilde{y},z,g) \geq S(x,y,z,g) \text{ and } S(x,\tilde{y},z,g) \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

Vacant firms can now hire both unemployed workers and employed workers, which leads to the HJBE:

$$\begin{aligned} \rho V^v(y,z,g) = {} & \mathcal{M}^v \mathcal{C}^u \int \alpha(\tilde{x},y,z,g)(V^p(\tilde{x},y,z,g) - V^v(x,z,g)) \frac{g^u(\tilde{x})}{\mathcal{U}} d\tilde{x} \\ & + \mathcal{M}^v \mathcal{C}^e \int\int \alpha^p(y,\tilde{x},\tilde{y},z,g)(1-\beta)(S(\tilde{x},y,z,g) - S(\tilde{x},\tilde{y},z,g)) \frac{g_m(\tilde{x},\tilde{y})}{\mathcal{E}} d\tilde{x} d\tilde{y} \\ & + \sum_{\check{z} \neq z} \lambda(z,\check{z})(V^v(x,\check{z},g) - V^v(x,z,g)) + \langle D_g V^v, \tilde{\mu}^g \rangle \end{aligned}$$

where α^p is an indicator function for whether a firm and an employed worker accept a match. Under generalized Nash Bargaining, this is given by the analogous function to α^e:

$$\alpha^p(y,\tilde{x},\tilde{y},z,g) := \begin{cases} 1, & \text{if } S(\tilde{x},y,z,g) \geq S(\tilde{x},\tilde{y},z,g) \text{ and } S(\tilde{x},y,z,g) \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

Finally, producing firms can now endogenously break up matches, which leads to the HJBE:

$$\begin{aligned} \rho V^p(x,y,z,g) = {} & F(x,y,z) - w(x,y,z,g) \\ & + (\delta(x,y,z) + \eta \alpha^b(x,y,z,g))(V^v(x,z,g) - V^p(x,y,z,g)) \\ & + \sum_{\check{z} \neq z} \lambda(z,\check{z})(V^p(x,\check{z},g) - V^p(x,z,g)) + \langle D_g V^p, \tilde{\mu}^g \rangle. \end{aligned}$$

Equilibrium: In Appendix E.1, we complete the equilibrium characterization by specifying the KFE and the master equation (the on-the-job search analogues of equations (2.7) and (2.10) respectively).

4.3 Deep Learning-Based Calibration

We calibrate the parameters $\{\beta, \kappa, c, b, \delta\}$ internally to match the ergodic unemployment rate, vacancy rate, employment-to-employment transition rate, unemployment-to-employment transition rate, and employment-to-unemployment transition rates, denoted by $\mathbb{E}[U], \mathbb{E}[V], \mathbb{E}[EE], \mathbb{E}[UE], \mathbb{E}[EU]$ respectively. The other parameters are taken from the literature and outlined in Table 8 in Appendix B. We discretize workers into seven productivity types and firms into eight productivity types, which is sufficiently fine to match the empirical regularities in the data.

Parameter	Interpretation	Value	Fitted moment	Data	Model
β	Surplus division factor	0.73	$\mathbb{E}[U]$	0.058	0.058
κ	Scale for meeting function	15.88	$\mathbb{E}[V]$	0.037	0.037
c	Entry cost	9.46	$\mathbb{E}[EE]$	0.025	0.026
b	Worker unemployment benefit	0.03	$\mathbb{E}[UE]$	0.468	0.431
δ	Separation rates	0.02	$\mathbb{E}[EU]$	0.025	0.026

Table 4: Internally Calibrated Parameters and Targeted Moments.

We undertake the deep learning based internal calibration outlined in Section 2.4.2. The internally calibrated parameters, the data moments, and the model moments are shown in Table 4. The target moments are taken from Lise and Robin (2017) and the transition flow moments (including expected rate of employment-to-employment, employment-to-unemployment, and unemployment-to-employment transitions) are presented as monthly values to match with the original paper. The computational times for each step of the calibration, as well as the associated numerical losses, are shown in Table 5. Evidently, we are able to get a close match to all five moments. To help illustrate this visually, Figure 5 shows the surrogate neural network mapping $\Phi(\Psi)$ from economic parameters to the aggregate moment loss for the dimensions β and κ (holding the other parameters fixed at their optimal values). This illustrates the curvature in the surrogate function that allows DeepSAM to find the optimal parameters. The entire solution and calibration process takes 5 hours and 5 minutes, where the model is solved over the economic parameter space and simulated across 10,000 parameter combinations to build the surrogate model deployed for the simulated method of moments. For reference, solving the problem for given structural parameter values, which is a 59-dimensional PDE, will take 55 minutes. To our knowledge, it's infeasible to solve such a high dimensional problem within such

a time frame using other approaches. Furthermore, the full calibration only takes 5 hours and 5 minutes, making our method practically useful for quantitative analysis.

	Solution Given the Value of Structural Parameters	Solution with Structural Parameters as Pseudo-states	Training Surrogate Model	Simulated Method of Moments	Full Calibration
MSE Loss	1.97×10^{-6}	4.8×10^{-6}	6.13×10^{-7}	1.24×10^{-4}	-
Time	55min	4h 1min	1h 3min	1.4min	5h 5min

Table 5: Training loss and computational time for solving vs calibrating the model. Computations are performed on the A100 GPU at Google Colab.

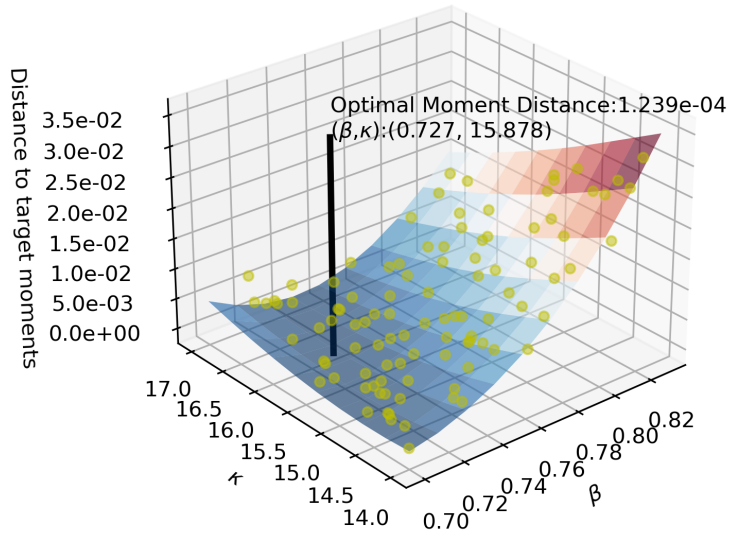


Figure 5: Target moment: $\mathbb{E}[U], \mathbb{E}[V]$. Parameter: matching efficiency κ, worker bargaining power β.

4.4 A Search-Theoretical Explanation for Okun's Hypothesis

We use our calibrated model to revisit the hypothesis put forward by Okun (1973) that longer expansions disproportionally improve labor market outcomes for low-wage workers. Figure 6 shows the impulse responses for an economy that goes into recession for half a year and then recovers to the high state. The left panel shows that the lowest and highest type workers experience a similar decline in unemployment for the first quarter of the expansion but then the low type workers experience a relatively

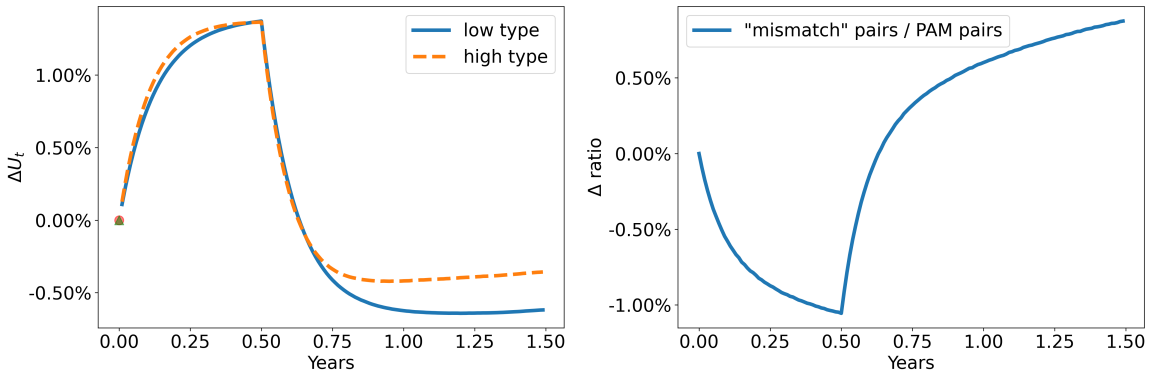


Figure 6: Left panel: ΔU_t for different workers. Low and high type refers to the lowest and highest types in our discretization. Right panel: The ratio of mismatched pairs to PAM pairs.

larger decrease in unemployment in the subsequent quarters. In this sense, low type workers benefit relatively more as the expansion lasts longer.

The explanation for Okun's hypothesis can be seen in the right panel of Figure 6, which depicts the ratio of mismatched pairs to positive assortative matching (PAM) pairs. We call a matched pair (x, y) a "PAM pair" if $|x - y| \leq \frac{1}{2}$, and otherwise call the pair a "non-PAM" or "mismatched pair". Evidently, there is a strong pattern of countercyclical sorting over the business cycles, which more pronounced the longer the expansion or recession extends. During the recession, the mass of "PAM pairs" grows faster than the mass of "non-PAM pairs", while the converse happens during the expansion. This is because high type workers become increasingly plentiful (scarce) during recessions (expansions) and so high type firms become more (less) picky. Ultimately, the weakening of positive assortative throughout the expansion disproportionately benefits low type agents and so generates the patterns predicted by Okun. In this sense, we can recover Okun's hypothesis from the distributional feedback without the need to introduce skill accumulation or other features into the model. This offers a complementary story to other papers in the literature (e.g. Alves and Violante (2023)), which potentially has different policy implications. This is because, in our model, the decrease in low-skilled unemployment during long booms comes from creating "non-PAM" pairs that are quickly broken up during subsequent recessions.

4.5 Dynamics of Wage Distribution

A well-known difficulty in the heterogeneous agent random search literature is that "block-recursive" models cannot solve for wage dynamics. This is because the surplus division does not inherit the block recursive property of total surplus. Or as Lentz, Lise, and Robin (2017) write: "wages cannot be solved for exactly, indeed one needs to solve for a fixed point in worker values where the distribution of workers across jobs is a state variable." By contrast, we can easily use our DeepSAM method to solve for the wage dynamics because it has solved for the surplus function explicitly as a function of the match distribution. We show the wage change after positive and negative aggregate shocks in Figure 7. Evidently, the wages of low-type workers are more procyclical, especially for those in high-type forms.

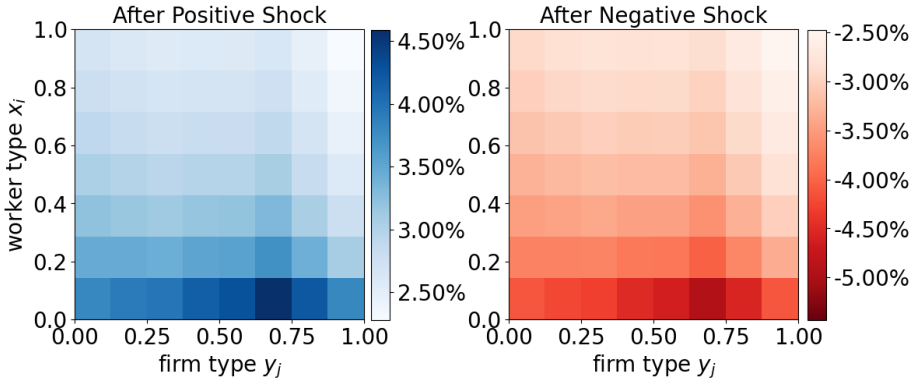


Figure 7: Wage change after aggregate shocks

4.6 Revisiting "Block Recursivity" with On-the-job Search

With the DeepSAM method, we are able to solve heterogeneous agent random search models with on-the-job search and endogenous separation beyond the block recursive case. As discussed in Appendix E.2, a key assumption in Lise and Robin (2017) for getting block recursivity is setting $\beta = 0$ so unemployed workers get zero surplus when bargaining with a firm. In Figure 8, we study how this restriction impacts impulse response. We present the times paths for unemployment (U_t), vacancies (V_t), the total quantity of poaching, and the total hires from unemployment in response to a 1.5% negative productivity shock that lasts for one year and then subsequently

reverts to the stochastic process governing aggregate TFP. [9]

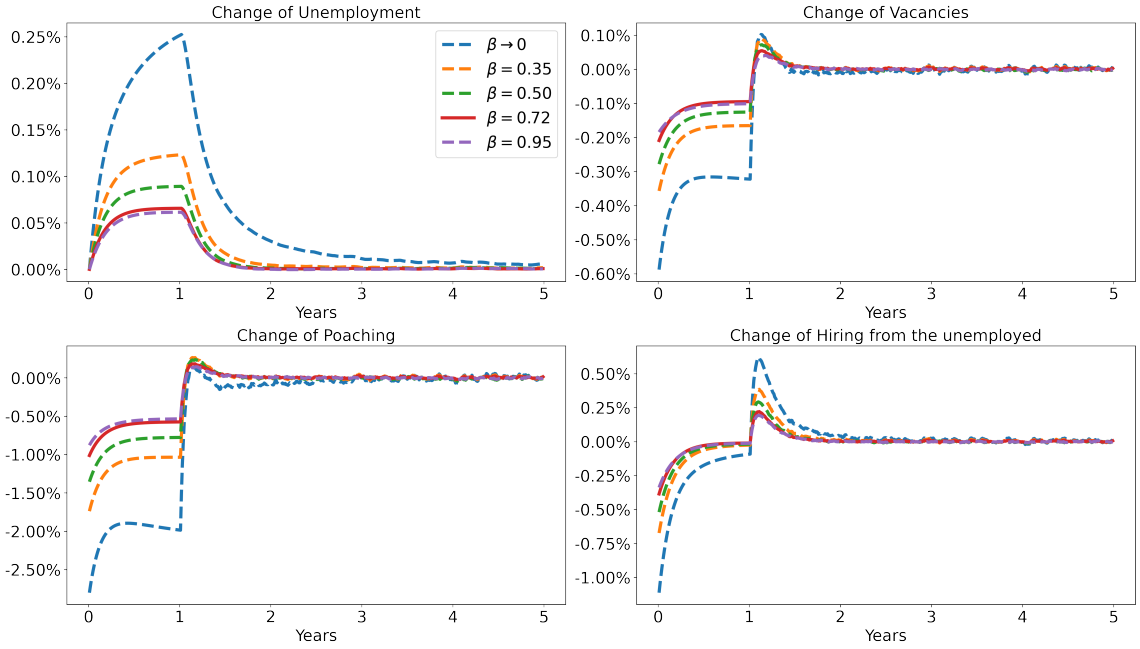


Figure 8: IRF following negative TFP Shock with different β's. After one year, the plots show the average time path across 1000 simulations.

We find that aggregate unemployment, vacancies, poaching, and hiring from unemployment are all most sensitive to aggregate shocks when $\beta = 0$. With on-the-job search, a 1.5% productivity shock leads to an increase in the unemployment rate of about 0.05% under our preferred calibration of $\beta = 0.72$. By contrast, a block recursive setup predicts a more substantial 0.25% increase in the unemployment rate. This is because firm posting is more elastic when firms get the entire surplus from matches with unemployed workers.

5 Over-the-Counter Financial Markets

In this final section, we consider a search model of over-the-counter bonds markets with heterogeneous investors and aggregate default risk. This can be thought of as an extension to Duffie, Gârleanu, and Pedersen (2005) and Weill (2008) that expands the investor and asset heterogeneity. We model bond duration explicitly and discuss

[9] The total quantity of *poaching* is defined as $\phi \frac{m(\mathcal{W}_t, \mathcal{V}_t)}{\mathcal{W}_t \mathcal{V}_t} \int \int \int \alpha_t^e(x, \tilde{y}, y) g_t^v(y) g_t(x, \tilde{y}) d\tilde{y} dx dy$. The total quantity of *hires from unemployment* is defined as $\frac{m(\mathcal{W}_t, \mathcal{V}_t)}{\mathcal{W}_t \mathcal{V}_t} \int \int \alpha_t(x, y) g_t^u(x) g_t^v(y) dx dy$.

the emergent prices. From a technical point of view, relative to the labor models, this section introduces type switching and asset trade.

5.1 Environment

Setting: Time is continuous with infinite horizon. The economy has a collection of assets, indexed by $k \in \{1, \ldots, K\}$ in positive net supply s_k, which we interpret as bonds. Each asset pays a flow dividend $\delta > 0$ each period and 1 good at maturity. Asset k matures at rate $1/\tau_k$ (implying an average maturity of τ_k).

Agents and preferences: The economy is populated by a unit-mass continuum of infinitely-lived and risk-neutral investors. Let $j \in \{1, \ldots, J\}$ denote the set of investor types in the economy. Agents of type j discount the future at rate $\rho_j > 0$. An investor gets marginal utility of 1 from a non-storable numeraire good. In order to make payments, investors are endowed with a technology that instantly produces numeraire goods, at a unit marginal cost. An investor can hold either zero or one share of at most one type of asset. An investor can have heterogeneous valuations of holding the assets. When agent j holds asset k, they get flow utility $\delta - \psi(j, k)$, where $\psi(j, k)$ is interpreted as the holding cost that reflects institutional constraints. Investors switch randomly, and pair-wise independently across types. Let Λ denote the matrix of switching rates and let $\lambda_{i,j}$ denote rate of switching from type i to j.

Financial crisis risk: The aggregate state in the economy is $z \in \{z_1, \ldots, z_n\}$, which follows a continuous time Markov process where $\zeta_{z,z'}$ denotes the rate at which the process switches from z to z'. We potentially allow the aggregate state to affect agent switching rates and haircuts. Formally, at state z, the switching rate from agent type i to agent type j is given by $\lambda_{ij}(z)$. Changes to $\lambda_{ij}(z)$ impact the fraction of agents with low or high holding costs so, as in Duffie, Gârleanu, and Pedersen (2005), we interpret these shocks as changes to the "liquidity constraints" in the investor population. In addition, at state z, asset k pays a fraction $\phi(k, z)$ of the coupon and the principal. We interpret $\phi(k, z)$ as the "haircut" on the bond.

Primary market: When bonds mature, they are replaced my new bonds in the economy. We impose there is an exogenous primary market that allocates new bonds to

non asset holding agents with holding cost j at rate $\xi_{j,k}$.

Distribution: An investor's type is made up of her holding cost $j \in \{1, \ldots, J\}$ and her ownership status, for each asset type $k \in \{1, \ldots, K\}$ (owner o or non-owner n). Hence the set of investor idiosyncratic states is:

$$A = \{1n, 2n, \ldots, Jn, 1o1, \ldots, 1oK, 2o1, \ldots 2oK, Jo1, \ldots, JoK\}$$

Relative to the labor model, the support of the distribution has been expanded to account for type switching. For each $a \in A$, let g_a denote the fraction of total investors that have state a. Let $g = (g_a)_{a \in A}$ denote the distribution.

Meeting and Bargaining: The contact rate between investors with idiosyncratic states a and b is: $\mathcal{M}_{a,b} = \kappa_{a,b} g_a g_b$. When agents with states a and b meet, they engage in Generalized Nash bargaining with bargaining power $\beta_{a,b}$ for agent in state a.

5.2 Equilibrium

The aggregate states are (z, g). We again denote the law of motion for the cross-sectional idiosyncratic state distribution, g, by the form $dg_t(a) = \mu^g(a, z, g)dt$. Let $V(in, z, g)$ denote the value function for an investor of type i without an asset and let $V(iok, z, g)$ denote the value function for an investor of type i with an asset.

Trade and surplus division: Investors with idiosyncratic states a and b trade if the surplus from exchanging assets is positive. Formally, if an agent i with an asset k meets an agent j without the asset, then the total surplus is:

$$S(iok, jn, z, g) = V(in, z, g) - V(iok, z, g) + V(jok, z, g) - V(jn, z, g)$$

Let $\Delta V_{i[ok \to n]}(z, g) := V(in, z, g) - V(iok, z, g)$ and $\Delta V_{j[n \to ok]}(z, g) := V(jok, z, g) - V(jn, z, g)$. Then, the generalized Nash bargaining protocol implies that:

$$\begin{aligned} \beta_{iok,jn} S(iok, jn, z, g) &= \Delta V_{i[ok \to n]} + p_k(in, jok, z, g) \\ (1 - \beta_{iok,jn}) S(iok, jn, z, g) &= \Delta V_{j[n \to ok]} - p_k(in, jok, z, g) \end{aligned}$$

and so the price paid is $p_k(jok, in, z, g) = \beta_{iok,jn}|\Delta V_{j[ok\to n]}(z,g)|+(1-\beta_{iok,jn})|\Delta V_{i[n\to ok]}(z,g)|$. Similarly, if an agent i with asset k meets an agent j with asset l, then the total surplus if investor i with asset k exchanges with investor j with asset l is:

$$S(iok, jol, z, g) = V(iol, z, g) - V(iok, z, g) + V(jok, z, g) - V(jol, z, g)$$

The generalized Nash bargaining protocol implies that:

$$\begin{aligned}\beta_{iok,jol}S(iok, jol, z, g) &= \Delta V_{i[ok\to ol]} + [p_k - p_l](iol, jok, z, g)\\ (1-\beta_{iok,jol})S(iok, jol, z, g) &= \Delta V_{j[ol\to ok]} - [p_k - p_l](iol, jok, z, g)\end{aligned}$$

and the absolute value of the price difference is:

$$[|p_k - p_l|](iok, jol, z, g) = \beta_{iok,jol}|\Delta V_{i[ok\to ol]}(z,g)| + (1-\beta_{iok,jol})|\Delta V_{j[ol\to ok]}(z,g)|.$$

Hamilton-Jacobi-Bellman-Equations: The value function for a non-owner with type i is given by the following equilibrium HJBE:

$$\begin{aligned}\rho_i V(in, g, z) = &\sum_a \kappa_{in,a}\alpha(in, a, g, z)\beta_{in,a}S(in, a, z, g)\\ &+ \sum_k \xi_{i,k}(V(iok, g, z) - V(in, g, z)) + \sum_{j\neq i}\lambda_{i,j}(z)(V(jn, g, z) - V(in, g, z))\\ &+ \sum_{z'}\zeta_{z,z'}(V(in, g, z') - V(in, g, z)) + \sum_{a\in A}\partial_{g_a}V(in, g, z)\mu^g(a, z)\end{aligned}$$

where $\alpha(in, jok, g, z)$ is an indicator function for whether the trade is accepted upon matching, which in equilibrium occurs if the surplus from the trade is positive $S(in, jok, g, z) > 0$. Likewise, the value function for an investor of type i holding asset k, $V(iok, g, z)$, is given by the following equilibrium HJBE:

$$\begin{aligned}\rho_i V(iok, g, z) = &\delta\phi(k, z) - \psi(i, k) + \frac{1}{\tau_k}(V(in, g, z) + \pi(k, z) - V(iok, g, z))\\ &+ \sum_a \kappa_{iok,a}\alpha(iok, a, g, z)g_a\beta_{iok,a}S(iok, a, g, z) + \sum_{a\in A}\partial_{g_a}V(iok, g, z)\mu^g(a, z)\\ &+ \sum_{j\neq i}\lambda_{i,j}(z)(V(jok, g, z) - V(iok, g, z)) + \sum_{z'}\zeta_{z,z'}(V(iok, g, z') - V(iok, g, z)).\end{aligned}$$

Kolmogorov Forward Equation: The distribution evolution for non-owner states and

owner states are given respectively by:

$$\begin{aligned}
\frac{dg_{in}}{dt} &= \mu^g(in,z,g) = \sum_{j\neq i}\lambda_{j,i}(z)g_{jn} + \sum_{j\neq i}\sum_k \kappa_{jn,iok}g_{jn}g_{iok}\alpha(jn,iok,g,z) \\
&\quad - \sum_{j\neq i}\lambda_{i,j}(z)g_{in} - \sum_{j\neq i}\sum_k \kappa_{in,jok}g_{in}g_{jok}\alpha(in,jok,g,z) + \sum_k \frac{1}{\tau_k}g_{iok} - \sum_k \xi_{i,k}g_{in} \\
\frac{dg_{iok}}{dt} &= \mu^g(iok,z,g) = \sum_{j\neq i}\lambda_{j,i}(z)g_{jok} - \sum_{j\neq i}\kappa_{jn,iok}g_{jn}g_{iok}\alpha(jn,iok,g,z) \\
&\quad + \sum_{j\neq i}\kappa_{in,jok}g_{in}g_{jok}\alpha(in,jok,g,z) - \sum_{j\neq i}\sum_{l\neq k}\kappa_{iok,jol}g_{iok}g_{jol}\alpha(iok,jol,g,z) \\
&\quad + \sum_{j\neq i}\sum_{l\neq k}\kappa_{iol,jok}g_{iol}g_{jok}\alpha(iol,jok,g,z) - \sum_{j\neq i}\lambda_{i,j}(z)g_{iok} - \frac{1}{\tau_k}g_{iok} + \xi_{i,k}g_{in}
\end{aligned}$$

In equilibrium, we must have that the flows from assets maturing are equal to the flows from new assets being created. That is, we need that: $\sum_i \xi_{i,k}g_{in} = \frac{1}{\tau_k}\sum_i g_{iok} =: \frac{1}{\tau_k}s_k$.

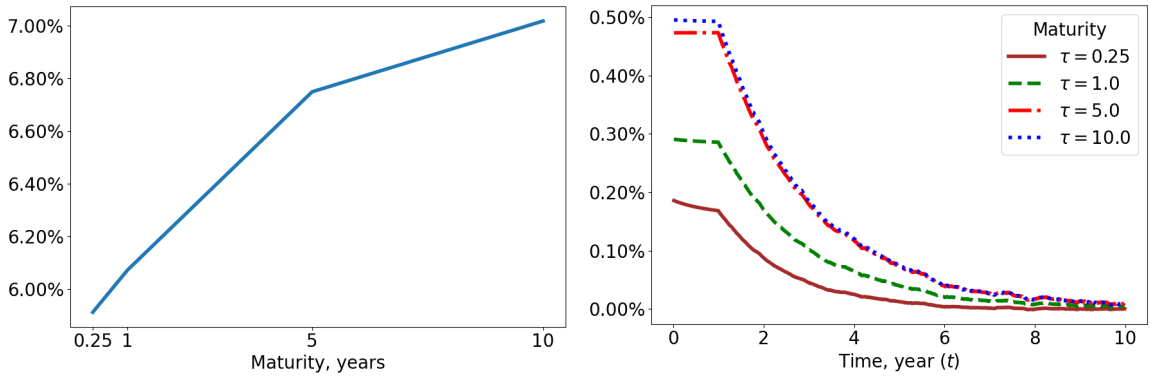
Comparison to the labor model: The OTC market cannot be reduced to one equation for the surplus, like we did in Section 2.2.5. Instead, we solve the two equilibrium HJBEs combined with the KFE.

5.3 Endogenous Yield Curve and Financial Crises

To illustrate the solution, we outline a particular calibration with four types of agents: $\{A, B, C, D\}$, where type A are interpreted as dealers in the primary bond market, type B are interpreted as liquidity constrained hedge funds, type C are non-liquidity constrained hedge funds, and type D are pension/insurance funds with a long investment horizon. This is reflected in their holding costs. Non-liquidity constrained investors have no holding cost while liquidity constrained investors have a holding cost of 0.2. Pension/insurance funds face holding costs of 0.02 for short maturity bonds ($\tau_1 = 0.25, 1.0$), 0.01 for bonds with $\tau = 5.0$, and no holding cost for long term bonds $\tau = 1.0$. We interpret this as reflecting regulatory constraints or financial frictions that encourage the hedge funds to hold long-term bonds. We impose that types dealer and pension/insurance types are constant. By contrast, hedge funds switch from C to B (i.e. becoming liquidity constrained) at rate 0.3 in the good state, 0.5 in the normal state, and 0.7 in the bad state. In all states, they switch from B to C (i.e. becoming non-liquidity constrained) at rate 0.1. We choose parameters so that

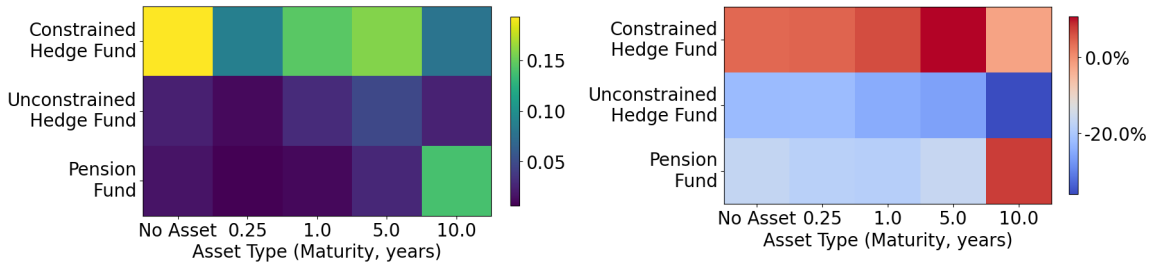
the ergodic yield curve matches the average high grade corporate yield curve over the past 50 years documented by Payne and Szőke (2024) and the haircut rates during the crisis to match Chen, Cui, He, and Milbradt (2017). We explain the calibration in more detail in Appendix F.

Figure 9a shows the ergodic mean bond prices as a function of maturity. Evidently, longer maturity bonds have lower prices indicating an upward sloping yield curve. This shape reflects relative investor willingness to hold short and long maturity bonds in the economy. Hedge funds prefer to hold short maturity bonds because they are worried that they will end up stuck with long maturity bonds if they become liquidity constrained. By contrast, the pension/insurance fund is penalized when they hold short maturity bonds. Under our parametrization, it is the first effect that dominates.



(a) Ergodic Yield Curve (b) Change of Yield to Maturity by Bond

Figure 9: Ergodic Yield Curve and Impulse Responses. Plot (b) shows the proportional bond yield change compared to the ergodic yield at each maturity following a one-year recession. To calculate the figures, we simulate 3000 paths and calculate the mean.



(a) Ergodic Distribution g^{ergodic} (b) Change after Crisis $\frac{g(z_L)-g^{\text{ergodic}}}{g^{\text{ergodic}}}$

Figure 10: Distribution response to a crisis shock.

We use our model to examine the impact of a liquidity crisis in an over-the-counter

bond market. Figure 9b shows the impulse responses for bond prices following liquidity crisis shock. Specifically, the economy starts at the ergodic mean and then moves to the "bad" state z_B for the first year. Afterwards, the economy is simulated 200 times and the mean paths are calculated. Evidently, for short maturity bonds, the yields move very little whereas for long maturity bonds the yields increase significantly. Finally, Figure 10 shows the change in the distribution of investors (relative to the ergodic mean) when the economy stays for a long time in the bad state and the good state. Evidently, the crisis increases the likelihood that a hedge fund becomes constrained and so increases the proportion of liquidity constrained hedge funds. This heightens hedge fund concern that they will end up stuck with long maturity bonds while liquidity constrained and so relative demand for long-term bonds drops during the crisis leading to the relatively large yield increase (price drop) for long-term bonds in Figure 9b.

6 Conclusion

In this paper, we developed a new method for characterizing global solutions to search and matching models with aggregate shocks and heterogeneous agents. This allows us to study dynamics in models where agent decisions depend upon the distribution so the model is not "block-recursive". We believe our methodology is a major breakthrough in our understanding of search and matching dynamics with potential applications in the labor, finance, and spatial literature.

References

Alvarez, F., F. Lippi, and P. Souganidis (2023): "Price setting with strategic complementarities as a mean field game," *Econometrica*, 91, 2005–2039.

Alves, F. (2022): "Job ladder and business cycles," Tech. rep., Bank of Canada.

Alves, F. and G. L. Violante (2023): "Some Like It Hot: Monetary Policy Under Okun's Hypothesis," .

Azinovic, M., L. Gaegauf, and S. Scheidegger (2022): "Deep equilibrium nets," *International Economic Review*, 63, 1471–1525.

Azinovic, M. and J. Žemličkа (2023): "Economics-inspired neural networks with stabilizing homotopies," *arXiv preprint arXiv:2303.14802*.

Baley, I., A. Figueiredo, and R. Ulbricht (2022): "Mismatch cycles," *Journal of Political Economy*, 130, 2943–2984.

Bilal, A. (2023): "Solving heterogeneous agent models with the master equation," Tech. rep., National Bureau of Economic Research.

Birinci, S., F. Karahan, Y. Mercan, and K. See (2024): "Labor market shocks and monetary policy," *Working Paper*.

Cahuc, P., F. Postel-Vinay, and J.-M. Robin (2006): "Wage bargaining with on-the-job search: Theory and evidence," *Econometrica*, 74, 323–364.

Cajner, T., L. D. Crane, R. A. Decker, J. Grigsby, A. Hamins-Puertolas, E. Hurst, C. Kurz, and A. Yildirmaz (2020): "The US Labor Market during the Beginning of the Pandemic Recession," *Brookings Papers on Economic Activity*, 3–33.

Cardaliaguet, P., F. Delarue, J.-M. Lasry, and P.-L. Lions (2015): "The master equation and the convergence problem in mean field games," *arXiv*.

Chen, H., R. Cui, Z. He, and K. Milbradt (2017): "Quantifying Liquidity and Default Risks of Corporate Bonds over the Business Cycle," *The Review of Financial Studies*.

Chen, H., A. Didisheim, and S. Scheidegger (2023): "Deep surrogates for finance: With an application to option pricing," *Available at SSRN 3782722*.

Duarte, V., D. Duarte, and D. H. Silva (2024): "Machine learning for continuous-time finance," *The Review of Financial Studies*, 37, 3217–3271.

Duarte, V. and J. Fonseca (2024): "Global identification with gradient-based structural estimation," .

Duffie, D., N. Gârleanu, and L. H. Pedersen (2005): "Over-the-counter markets," *Econometrica*, 73, 1815–1847.

Engbom, N. (2021): "Contagious unemployment," Tech. rep., National Bureau of Economic Research.

Fernández-Villaverde, J., S. Hurtado, and G. Nuno (2023): "Financial frictions and the wealth distribution," *Econometrica*, 91, 869–901.

Fernández-Villaverde, J., G. Nuno, and J. Perla (2024): "Taming the Curse of Dimensionality: Quantitative Economics with Deep Learning," Tech. rep.

Friedl, A., F. Kübler, S. Scheidegger, and T. Usui (2023): "Deep uncertainty quantification: with an application to integrated assessment models," Tech.

rep., Working Paper University of Lausanne.

Fukui, M. (2020): "A theory of wage rigidity and unemployment fluctuations with on-the-job search," .

Gopalakrishna, G. (2021): "Aliens and continuous time economies," *Swiss Finance Institute Research Paper*.

Gu, Z., M. Lauriere, S. Merkel, and J. Payne (2023): "Deep Learning Solutions to Master Equations for Continuous Time Heterogeneous Agent Macroeconomic Models," Tech. rep.

Guvenen, F., S. Schulhofer-Wohl, J. Song, and M. Yogo (2017): "Worker betas: Five facts about systematic earnings risk," *American Economic Review*, 107, 398–403.

Hagedorn, M., T. H. Law, and I. Manovskii (2017): "Identifying equilibrium models of labor market sorting," *Econometrica*, 85, 29–65.

Han, J., Y. Yang, and W. E (2021): "DeepHAM: A Global Solution Method for Heterogeneous Agent Models with Aggregate Shocks," *arXiv preprint arXiv:2112.14377*.

Huang, J. (2022): "A Probabilistic Solution to High-Dimensional Continuous-Time Macro-Finance Models," *Available at SSRN 4122454*.

——— (2024): "Breaking the Curse of Dimensionality in Heterogeneous-Agent Models: A Deep Learning-Based Probabilistic Approach," *Available at SSRN 4649043*.

Kahou, M. E., J. Fernández-Villaverde, J. Perla, and A. Sood (2021): "Exploiting symmetry in high-dimensional dynamic programming," Tech. rep., National Bureau of Economic Research.

Kaplan, G., B. Moll, and G. Violante (2018): "Monetary Policy According to HANK," *American Economic Review*, 108, 697–743.

Kase, H., L. Melosi, and M. Rottner (2024): "Estimating Nonlinear Heterogeneous Agent Models with Neural Networks," .

Krusell, P., T. Mukoyama, R. Rogerson, and A. Şahin (2017): "Gross worker flows over the business cycle," *American Economic Review*, 107, 3447–3476.

Krusell, P. and A. A. Smith (1998): "Income and Wealth Heterogeneity in the Macroeconomy," *Journal of Political Economy*, 106, 867–896.

Lagos, R. and G. Rocheteau (2009): "Liquidity in asset markets with search frictions," *Econometrica*, 77, 403–426.

Lentz, R., J. Lise, and J.-M. Robin (2017): "The Macro Dynamics of the Wage Distribution," in *2017 Meeting Papers*, vol. 1220.

Lise, J. and J.-M. Robin (2017): "The macrodynamics of sorting between workers and firms," *American Economic Review*, 107, 1104–1135.

Maliar, L., S. Maliar, and P. Winant (2021): "Deep learning for solving dynamic economic models." *Journal of Monetary Economics*, 122, 76–101.

Menzio, G. and S. Shi (2010): "Block recursive equilibria for stochastic models of search on the job," *Journal of Economic Theory*, 1.

——— (2011): "Efficient Search on the Job and the Business Cycle," *Journal of Political Economy*, 119, 468–510.

Moen, E. R. (1997): "Competitive Search Equilibrium," *Journal of Political Economy*, 105, 385–411.

Mortensen, D. T. and C. A. Pissarides (1994): "Job creation and job destruction in the theory of unemployment," *The review of economic studies*, 61, 397–415.

Moscarini, G. and F. Postel-Vinay (2018): "The cyclical job ladder," *Annual Review of Economics*, 10, 165–188.

——— (2023): "The job ladder: Inflation vs. reallocation," Tech. rep., National Bureau of Economic Research.

Norets, A. (2012): "Estimation of dynamic discrete choice models using artificial neural network approximations," *Econometric Reviews*, 31, 84–106.

Okun, A. M. (1973): "Upward mobility in a high-pressure economy," *Brookings Papers on Economic Activity*, 1973, 207–261.

Payne, J. and B. Szőke (2024): "Convenience Yields and Financial Repression," .

Postel-Vinay, F. and J.-M. Robin (2002): "Equilibrium wage dispersion with worker and employer heterogeneity," *Econometrica*, 70, 2295–2350.

Qiu, X. (2023): "Vacant Jobs," .

Raissi, M., P. Perdikaris, and G. E. Karniadakis (2019): "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational physics*, 378, 686–707.

Sauzet, M. (2021): "Projection methods via neural networks for continuous-time models," *Available at SSRN 3981838*.

Schaal, E. (2017): "Uncertainty, Productivity and Unemployment in the Great Recession," *Econometrica*, 85, 250–274.

Shimer, R. (2005): "The cyclical behavior of equilibrium unemployment and vacancies," *American economic review*, 95, 25–49.

Shimer, R. and L. Smith (2000): "Assortative matching and search," *Econometrica*, 68, 343–369.

Weill, P.-O. (2008): "Liquidity premia in dynamic bargaining markets," *Journal of Economic Theory*, 140, 66–96.

Appendix

A Details of the Labor Market Model

Parameter	Interpretation	Value	Target/Source
ρ	Discount rate	0.05	Kaplan et al. (2018)
ξ	Extreme value for α choice	2.0	
$f(x,y)$	Production for match (x,y)	$0.6+0.4\left(\sqrt{x}+\sqrt{y}\right)^2$	Hagedorn et al. (2017)
β	Surplus division factor	0.72	Shimer (2005)
$m(\mathcal{U},\mathcal{V})$	Matching function	$\kappa\mathcal{U}^{\nu}\mathcal{V}^{1-\nu}$	Hagedorn et al. (2017)
ν	Elasticity in meeting function	0.5	Hagedorn et al. (2017)
κ	Scale for meeting function	5.4	Unemployment rate
b	Worker unemployment benefit	0.5	Shimer (2005)
c	Entry cost	4.86	Steady state $\mathcal{V}/\mathcal{U}=1$
Steady State:			
$\bar{z}$	Steady state TFP	1	Shimer (2005)
$\bar{\delta}$	Steady state separation rate	0.2	BLS job tenure 5 years
Exogenous Aggregate Shock Process:			
A_D, A_L, A_H	TFP levels	0.985, 0.985, 1.015	Lise and Robin (2017)
δ_L, δ_H	Separation rates	0.18, 0.22	Shimer (2005)
$\delta_D(x,y)$	TFP and separation at crisis state	See text	Match Cajner et al. (2020)
λ_z	Poisson transition probability	See Appendix D.1	Shimer (2005)
n_x	Discretization of worker types	5	Calibration of $\delta_D(x,y)$
n_y	Discretization of firm types	11	Calibration of $\delta_D(x,y)$

Table 6: Economic Parameters.

Parameter	Value	Parameter	Value
Number of layers	4	Final learning rate	10^{-5}
Neurons per layer	50	Initial sample size per epoch	256
Activation function	$\tanh(\cdot)$	Final sample size per epoch	512
Initial learning rate	10^{-4}	Convergence threshold	10^{-6}

Table 7: Neural network parameters

B Details of the OJS Model

Parameter	Interpretation	Value
ρ	Discount rate	0.05
ξ, ξ^e	Extreme value for α and α^e choices	2.5, 0.02
$f(x,y)$	Production for match (x,y)	$0.6 + 0.4\left(\sqrt{x} + \sqrt{y}\right)^2$
$m(\mathcal{U},\mathcal{V})$	Matching function	$\kappa \mathcal{U}^{\nu}\mathcal{V}^{1-\nu}$
ν	Elasticity in meeting function	0.5
ϕ	Relative intensity	0.075
A_L, A_H	TFP levels	0.985, 1.015
λ_z	Poisson transition probability	0.08
n_x	Discretization of worker types	7
n_y	Discretization of firm types	8

Table 8: Externally Calibrated Parameters.

Online Appendix

C Master Equation with the Free Entry Condition

As in Section 2.2.4, the free entry condition is

$$0 = \int V_t^v(\tilde{y})d\tilde{y}$$

Recall from (2.4) the HJB equation for a vacant institution with productivity y is

$$\rho V_t^v(y) = -c + \mathcal{M}_t^v \int \alpha(\tilde{x}, y)\frac{g_t^u(\tilde{x})}{\mathcal{U}_t}(1-\beta)S_t(\tilde{x}, y)d\tilde{x} + \partial_t V_t^v(y)$$

Integrating and combining these equations, we have that:

$$\begin{aligned}
\rho \int V_t^v(\tilde{y})d\tilde{y} = & -c + \mathcal{M}_t^v \int\int \alpha(\tilde{x}, \tilde{y})\frac{g_t^u(\tilde{x})}{\mathcal{U}_t}(1-\beta)S_t(\tilde{x}, \tilde{y})d\tilde{x}d\tilde{y} + \partial_t \int V_t^v(\tilde{y})d\tilde{y} \\
\Rightarrow c = & \frac{m(\mathcal{U}_t, \mathcal{V}_t)}{\mathcal{V}_t} \int\int \alpha(\tilde{x}, \tilde{y})\frac{g_t^u(\tilde{x})}{\mathcal{U}_t}(1-\beta)S_t(\tilde{x}, \tilde{y})d\tilde{x}d\tilde{y} \\
\Rightarrow \frac{m(\mathcal{U}_t, \mathcal{V}_t)}{\mathcal{V}_t} = & \frac{c}{\int\int \alpha(\tilde{x}, \tilde{y})\frac{g_t^u(\tilde{x})}{\mathcal{U}_t}(1-\beta)S_t(\tilde{x}, \tilde{y})d\tilde{x}d\tilde{y}}
\end{aligned}$$

Assuming that $m(\mathcal{U}_t, \mathcal{V}_t)/(\mathcal{V}_t) = \widehat{m}(\mathcal{V}_t/\mathcal{U}_t)$, we have that:

$$\mathcal{V}_t = \mathcal{U}_t \widehat{m}^{-1}\left(\frac{c}{\int\int \alpha(\tilde{x}, \tilde{y})\frac{g_t^u(\tilde{x})}{\mathcal{U}_t}(1-\beta)S_t(\tilde{x}, \tilde{y})d\tilde{x}d\tilde{y}}\right) \tag{C.1}$$

where $g_t^u = g_t^w - \int g_t(x, y)dy$ and so the RHS can be computed from g_t and S_t. (For example, if $m(\mathcal{U}, \mathcal{V}) = \kappa \mathcal{U}^\nu \mathcal{V}^{1-\nu}$, then $\mathcal{M}_t^v = m(\mathcal{U}, \mathcal{V})/\mathcal{V} = \widehat{m}(\mathcal{V}_t/\mathcal{U}_t) = \kappa(\mathcal{U}/\mathcal{V})^\nu$ and $\mathcal{M}_t^u = m(\mathcal{U}, \mathcal{V})/\mathcal{U} = \kappa(\mathcal{V}/\mathcal{U})^{1-\nu}$). Since firm y draws are uniformly distributed, we have that g_t^f is given by:

$$\begin{aligned}
g_t^f &= \mathcal{V}_t + \mathcal{P}_t \\
&= \mathcal{U}_t \widehat{m}^{-1}\left(\frac{c}{\int\int \alpha(\tilde{x}, \tilde{y})(g_t^u(\tilde{x})/\mathcal{U}_t)(1-\beta)S_t(\tilde{x}, \tilde{y})d\tilde{x}d\tilde{y}}\right) + \int\int g_t(x, y)dydx
\end{aligned}$$

where $\mathcal{V}_t$ is from (C.1), $\mathcal{P}_t = \int\int g_t(x,y)dydx$, and $\mathcal{U}_t = \int(g_t^w(x) - \int g_t(x,y)dy)$. This means that g_t^f can be computed from g_t and S_t, and

$$\begin{aligned}
g_t^v(y) &= g_t^f(y) - g_t^p(y) \\
&= \mathcal{U}_t\widehat{m}^{-1}\left(\frac{c}{\int\int \alpha(\tilde{x},\tilde{y})(g_t^u(\tilde{x})/\mathcal{U}_t)(1-\beta)S_t(\tilde{x},\tilde{y})d\tilde{x}d\tilde{y}}\right) + \int\int g_t(x,y)dydx \\
&\quad - \int g_t(x,y)dx
\end{aligned}$$

We can now calculate the differential equation for surplus:

$$\begin{aligned}
\rho S_t(x,y) &= \rho(V_t^p(x,y) - V_t^v(y) + V_t^e(x,y) - V_t^u(x)) \\
&= f_t(x,y) - w_t(x,y) - \delta(1-\beta)S_t(x,y) + \partial_t V_t^p(x,y) \\
&\quad - \left(\mathcal{M}_t^v \int \alpha(\tilde{x},y)\frac{g_t^u(\tilde{x})}{\mathcal{U}_t}(1-\beta)S_t(\tilde{x},y)d\tilde{x} + \partial_t V_t^v(y)\right) \\
&\quad + w_t(x,y) - \beta\delta S_t(x,y) + \partial_t V_t^e(x,y) \\
&\quad - \left(b + \mathcal{M}_t^u \int \alpha_t(x,\tilde{y})\frac{g_t^v(\tilde{y})}{V_t}\beta S_t(x,\tilde{y})d\tilde{y} + \partial_t V_t^u(x)\right) \\
&= f_t(x,y) - \delta S_t(x,y) - \mathcal{M}_t^v \int \alpha(\tilde{x},y)\frac{g_t^u(\tilde{x})}{\mathcal{U}_t}(1-\beta)S_t(\tilde{x},y)d\tilde{x} \\
&\quad - b - \mathcal{M}_t^u \int \alpha_t(x,\tilde{y})\frac{g_t^v(\tilde{y})}{\mathcal{V}_t}\beta S_t(x,\tilde{y})d\tilde{y} + \partial_t S_t(x,y)
\end{aligned}$$

where:

$$\begin{aligned}
\mathcal{V}_t &= \mathcal{U}_t\widehat{m}^{-1}\left(\frac{c}{\int\int \alpha(\tilde{x},\tilde{y})\frac{g_t^u(\tilde{x})}{\mathcal{U}_t}(1-\beta)S_t(\tilde{x},\tilde{y})d\tilde{x}d\tilde{y}}\right) \\
\mathcal{M}_t^v &= m(\mathcal{U},\mathcal{V})/\mathcal{V} = \widehat{m}(\mathcal{V}_t/\mathcal{U}_t) = \kappa(\mathcal{U}/\mathcal{V})^\nu, \\
\mathcal{M}_t^u &= m(\mathcal{U},\mathcal{V})/\mathcal{U} = \kappa(\mathcal{V}/\mathcal{U})^{1-\nu} \\
g_t^v(y) &= g_t^f(y) - g_t^p(y) \\
&= \mathcal{V}_t + \mathcal{P}_t - g_t^p(y) \\
&= \mathcal{U}_t\widehat{m}^{-1}\left(\frac{c}{\int\int \alpha(\tilde{x},\tilde{y})(g_t^u(\tilde{x})/\mathcal{U}_t)(1-\beta)S_t(\tilde{x},\tilde{y})d\tilde{x}d\tilde{y}}\right) + \int\int g_t(x,y)dydx - \int g_t(x,y)dx
\end{aligned}$$

and the KFE is in the same form as (2.7) with different definitions of $g^f(y)$ and $\mathcal{V}$.

D Appendix For the Labor Search Model in Section 3

D.1 Calibration Details

The Poisson transition rate for aggregate shocks across high, low, and disaster states are:

$$\lambda_z = \begin{bmatrix} - & \lambda_{HL} & \lambda_{HD} \\ \lambda_{LH} & - & \lambda_{LD} \\ \lambda_{DH} & \lambda_{DL} & - \end{bmatrix} = \begin{bmatrix} - & 0.4 & 0.001 \\ 0.4 & - & 0.001 \\ 0.0995 & 0.0995 & - \end{bmatrix}$$

The calibrated separation rate across worker and firm types $\delta_D(x, y) =$

5.2834	4.3853	3.7621	3.3420	3.0671	2.8927	2.7848	2.7191	2.6787	2.6534	2.6370
3.3734	2.5752	2.0345	1.6818	1.4614	1.3303	1.2565	1.2169	1.1963	1.1855	1.1794
2.6337	1.9001	1.4115	1.1002	0.9121	0.8058	0.7505	0.7245	0.7136	0.7094	0.7077
2.3878	1.6936	1.2358	0.9478	0.7773	0.6837	0.6374	0.6175	0.6106	0.6087	0.6084
2.3072	1.6352	1.1938	0.9178	0.7555	0.6676	0.6249	0.6072	0.6014	0.6001	0.6000

D.2 Master equation and loss function for the model without aggregate shocks

To verify the accuracy of the DeepSAM method, we apply it to solve a labor search model without aggregate shocks, which can also be solved with a conventional numerical method such as that in Hagedorn, Law, and Manovskii (2017). The master equation and loss function for the Surplus is given by:

$$\begin{aligned} 0 = \mathcal{L}^S S = &-(\rho + \delta(x,y))S(x,y,\underline{\boldsymbol{g}}) + F(x,y) - b \\ &- (1-\beta)\frac{m(\underline{\boldsymbol{g}})}{\mathcal{U}(\underline{\boldsymbol{g}})\mathcal{V}(\underline{\boldsymbol{g}})}\frac{1}{n_x}\sum_{i=1}^{n_x}\alpha(x_i,y,\underline{\boldsymbol{g}})S(x_i,y,g)\underline{\boldsymbol{g}}^u(x_i) \\ &- \beta\frac{m(\underline{\boldsymbol{g}})}{\mathcal{U}(\underline{\boldsymbol{g}})\mathcal{V}(\underline{\boldsymbol{g}})}\frac{1}{n_y}\sum_{j=1}^{n_y}\alpha(x,y_j,\underline{\boldsymbol{g}})S(x,y_j,\underline{\boldsymbol{g}})\underline{\boldsymbol{g}}^v(y_j) \\ &+ \sum_{i=1}^{n_x}\sum_{i=1}^{n_y}\partial_{g_{ij}}S(x,y,\underline{\boldsymbol{g}})\mu^g(x_i,y_j,\underline{\boldsymbol{g}}) \end{aligned}$$

in which

$$dg_t(x,y)/dt = \mu^g(x,y,\underline{\boldsymbol{g}}_t) = -\delta(x,y)g_t(x,y) + \frac{m(\underline{\boldsymbol{g}}_t)}{\mathcal{U}_t\mathcal{V}_t}\alpha(x,y,\underline{\boldsymbol{g}}_t) \times \left(g^w(x) - \frac{1}{n_y}\sum_{j=1}^{n_y}\underline{\boldsymbol{g}}_t(x,y_j)\right)\left(g^f(y) - \frac{1}{n_x}\sum_{i=1}^{n_x}\underline{\boldsymbol{g}}_t(x_i,y)\right)$$

and $\alpha(x,y,g)$ is given by:

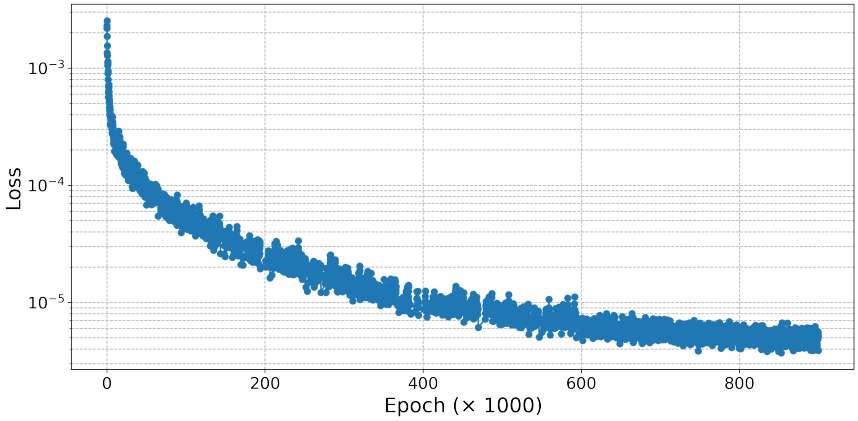
$$\alpha(x,y,\underline{\boldsymbol{g}}) = \left(1 + e^{-\xi S(x,y,\underline{\boldsymbol{g}})}\right)^{-1}.$$

D.3 Numerical method and performance

D.3.1 Hyperparameters for the neural networks

Training loss, learning rate, and sample size. Figure 11 presents the value of the loss function (2.12) along the training process. It takes 1.5 hours on an A100 GPU for the neural network to converge to a stable solution. The learning rate is 10^{-4} for the first 400,000 epoch, is 10^{-5} after that. Sample size: 256 in first 400k, 512 from after that. We use a cosine scheme for to adjust the learning rate over time.

Figure 11: Loss function along training epochs



E Additional Details For The On-The-Job Search Model in Section 4

This Appendix completes the setup for the the on-the-job search model described in Section 4.

E.1 Equilibrium Description

Master Equation For Surplus: Under belief consistency, the differential equation for the surplus becomes the following:

$$
\begin{aligned}
&\rho S(x,y,z,g) = \rho(V^p(x,y,z,g) - V^v(y,z,g) + V^e(x,y,z,g) - V^u(x,z,g)) \\
&= F(x,y,z,g) - w(x,y,z,g) - (\delta + \eta\alpha^b(x,y,z,g))(1-\beta)S(x,y,z,g) \\
&+ \langle D_g V^p, \mu^g\rangle - \Big(\mathcal{M}^v\mathcal{C}^u \int \alpha(\tilde{x},y,z,g)(1-\beta)S(\tilde{x},y,z,g)\frac{g^u(\tilde{x})}{\mathcal{U}}d\tilde{x} + \langle D_g V^v, \mu^g\rangle \\
&+ \mathcal{M}^v\mathcal{C}^e \int \alpha^p(y,\tilde{x},\tilde{y},z,g)\frac{g(\tilde{x},\tilde{y})}{\mathcal{E}}(1-\beta)(S(\tilde{x},y,z,g) - S(\tilde{x},\tilde{y},z,g))d\tilde{x}d\tilde{y}\Big) \\
&+ w(x,y,z,g) + \mathcal{M}^e \int \alpha^e(x,y,\tilde{y},z,g)\beta(S(x,\tilde{y},z,g) - S(x,y))\frac{g^v(\tilde{y})}{\mathcal{V}}d\tilde{y} \\
&- \beta(\delta + \eta\alpha^b(x,y,z,g))S(x,y,z,g) + \langle D_g V^e, \mu^g\rangle \\
&- \Big(b + \mathcal{M}^u \int \alpha(x,\tilde{y},z,g)\beta S(x,\tilde{y},z,g)\frac{g^v(\tilde{y})}{\mathcal{V}}d\tilde{y}\Big) \\
&= F(x,y,z) - (\delta + \eta\alpha^b(x,y,z,g))S(x,y,z,g) - b \\
&- \mathcal{M}^v\mathcal{C}^u \int \alpha(\tilde{x},y,z,g)(1-\beta)S(\tilde{x},y,z,g)\frac{g^u(\tilde{x})}{\mathcal{U}}d\tilde{x} \\
&- \mathcal{M}^v\mathcal{C}^e \int \alpha^p(y,\tilde{x},\tilde{y},z,g)\frac{g(\tilde{x},\tilde{y},z,g)}{\mathcal{E}}(1-\beta)(S(\tilde{x},y,z,g) - S(\tilde{x},\tilde{y},z,g))d\tilde{x}d\tilde{y} \\
&+ \mathcal{M}^e \int \alpha^e(x,y,\tilde{y},z,g)\beta(S(x,\tilde{y},z,g) - S(x,y,z,g))\frac{g^v(\tilde{y})}{\mathcal{V}}d\tilde{y} \\
&- \mathcal{M}^u \int \alpha(x,\tilde{y},z,g)\beta S(x,\tilde{y},z,g)\frac{g^v(\tilde{y})}{\mathcal{V}}d\tilde{y} \\
&+ \lambda(z)(S(x,y,\tilde{z},z,g) - S(x,y,z,g)) + \langle D_g S, \mu^g\rangle
\end{aligned}
$$

where:

$$\alpha(x,\tilde{y},z,g) := \begin{cases} 1, & \text{if } S(x,\tilde{y},z,g) > 0 \\ 0, & \text{otherwise} \end{cases}$$

$$\alpha^b(x,\tilde{y}) := \begin{cases} 1, & \text{if } S(x,\tilde{y},z,g) < 0 \\ 0, & \text{otherwise} \end{cases}$$

$$\alpha^e(x,y,\tilde{y},z,g) := \begin{cases} 1, & \text{if } S(x,\tilde{y},z,g) \geq S(x,y,z,g) \text{ and } S(x,\tilde{y},z,g) \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

$$\alpha^p(y,\tilde{x},\tilde{y},z,g) := \begin{cases} 1, & \text{if } S(\tilde{x},y,z,g) \geq S(\tilde{x},\tilde{y},z,g) \text{ and } S(\tilde{x},y,z,g) \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

Observe that:

$$\frac{\mathcal{M}^v\mathcal{C}^u}{\mathcal{U}} = \frac{m(\mathcal{W},\mathcal{V})}{\mathcal{W}\mathcal{V}}, \qquad \frac{\mathcal{M}^v\mathcal{C}^e}{\mathcal{E}} = \phi\frac{m(\mathcal{W},\mathcal{V})}{\mathcal{W}\mathcal{V}},$$
$$\frac{\mathcal{M}^e}{\mathcal{V}} = \phi\frac{m(\mathcal{W},\mathcal{V})}{\mathcal{W}\mathcal{V}}, \qquad \frac{\mathcal{M}^u}{\mathcal{V}} = \frac{m(\mathcal{W},\mathcal{V})}{\mathcal{W}\mathcal{V}}$$

and so the surplus equation becomes:

$$\begin{aligned}
\rho S(x,y,z,g) &= F(x,y,z,g) - (\delta + \eta\alpha^b(x,y,z,g))S(x,y,z,g) - b \\
&- (1-\beta)\frac{m(\mathcal{W},\mathcal{V})}{\mathcal{W}\mathcal{V}}\int \alpha(\tilde{x},y,z,g)S(\tilde{x},y,z,g)g^u(\tilde{x})d\tilde{x} \\
&- (1-\beta)\phi\frac{m(\mathcal{W},\mathcal{V})}{\mathcal{W}\mathcal{V}}\int \alpha^p(y,\tilde{x},\tilde{y},z,g)g(\tilde{x},\tilde{y})(S(\tilde{x},y,z,g) - S(\tilde{x},\tilde{y},z,g))d\tilde{x}d\tilde{y} \\
&+ \beta\phi\frac{m(\mathcal{W},\mathcal{V})}{\mathcal{W}\mathcal{V}}\int \alpha^e(x,y,\tilde{y},z,g)(S(x,\tilde{y},z,g) - S(x,y,z,g))g^v(\tilde{y})d\tilde{y} \\
&- \beta\frac{m(\mathcal{W},\mathcal{V})}{\mathcal{W}\mathcal{V}}\int \alpha(x,\tilde{y},z,g)S(x,\tilde{y},z,g)g^v(\tilde{y})d\tilde{y} \\
&+ \lambda(z)(S(x,y,\tilde{z},z,g) - S(x,y,z,g)) + \langle D_g S, \tilde{\mu}^g\rangle
\end{aligned}$$

Kolmogorov Forward Equation: The measure of matches evolves according to:

$$\begin{aligned} dg_t(x,y) = & -(\delta + \eta\alpha_t^b(x,y))g_t(x,y)dt \\ & - \underbrace{\mathcal{M}_t^e}_{\substack{\text{Rate. e-worker} \\ \text{meets}}} \underbrace{\int \underbrace{\alpha_t^e(x,y,\tilde{y})}_{\text{Prob. accept}} \underbrace{\frac{g_t^v(\tilde{y})}{\mathcal{V}_t}}_{\text{Prob. meet } \tilde{y}} d\tilde{y} \; \underbrace{g_t(x,y)}_{\text{Mass at } (x,y)}}_{\text{Mass accepting match}} dt \\ & + \underbrace{\mathcal{M}_t^u}_{\substack{\text{Rate. u-worker} \\ \text{meets}}} \alpha_t(x,y)\frac{g_t^v(y)}{\mathcal{V}_t}g_t^u(x)dt \\ & + \mathcal{M}_t^e \int \alpha_t^e(x,\tilde{y},y)\frac{g_t^v(y)}{\mathcal{V}_t}g_t(x,\tilde{y})d\tilde{y}dt \end{aligned}$$

where this KFE has been written from the perspective of the workers (it could equivalently be written from the point of view of the firms) and each term is written as:

(Prob. worker meets) $\times$ (Prob. acceptance) $\times$ (prob. y) $\times$ (mass of workers)

The first term on the RHS is the exit rate due to exogenous separations, the second term is the exit rate due to workers finding better matches, the third term is new matches from unemployed workers matching, and the final term is employed workers moving to (x,y). Observe that:

$$\frac{\mathcal{M}_t^e}{\mathcal{V}_t} = \phi\frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{W}_t\mathcal{V}_t} \qquad\qquad \frac{\mathcal{M}_t^u}{\mathcal{V}_t} = \frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{W}_t\mathcal{V}_t}$$

So, the KFE becomes:

$$\begin{aligned} dg_t(x,y) = & -(\delta + \eta\alpha_t^b(x,y))g_t(x,y)dt - \phi\frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{W}_t\mathcal{V}_t}g_t(x,y)\int \alpha_t^e(x,y,\tilde{y})g_t^v(\tilde{y})d\tilde{y}dt \\ & + \frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{W}_t\mathcal{V}_t}\alpha_t(x,y)g_t^u(x)g_t^v(y)dt \\ & + \phi\frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{W}_t\mathcal{V}_t}\int \alpha_t^e(x,\tilde{y},y)g_t^v(y)g_t(x,\tilde{y})d\tilde{y}dt \end{aligned}$$

If we know measure of matches, then we can recover the other distribution:

$$g_t^e(x) = \int g_t(x,y)dy, \qquad g_t^u(x) = g_t^w(x) - \int g_t(x,y)dy,$$
$$g_t^p(y) = \int g_t(x,y)dx \qquad g_t^v(y) = g_t^f(y) - \int g_t(x,y)dx,$$
$$\mathcal{U}_t = \int g_t^u(x)dx, \qquad \mathcal{V}_t = \int g_t^v(y)dy$$
$$\mathcal{E}_t = \int g_t^e(x)dx, \qquad \mathcal{P}_t = \int g_t^p(y)dy$$

E.2 Relating to Block Recursivity as in Lise and Robin (2017)

In this section, we show what changes are required in our environment to get the block recursive results in Lise and Robin (2017).

E.2.1 Environment

We make the following changes to the environment from subsection 3.1.

Setting: The economy is populated by a continuum of infinitely lived workers indexed by ability x, and a continuum of firms indexed by technology y. The total measures of workers and firms are fixed and normalized to 1 and their densities are given by $g_t^w(x)$ and $g_t^f(y)$. However, now firms can post v job opportunities at exogenous cost $c(v)$. The aggregate state of the economy is indexed by z_t. At the beginning of each period, the aggregate state changes from z to z' at Poisson rate $\pi(z, z')$.

Meeting Technology: The total effective search effort is $\mathcal{W}_t = \mathcal{U}_t + \phi\mathcal{E}_t$. Let $v_t(y)$ denote the measure of type y job opportunities chosen by firm y. Let $V_t = \int v_t(y)dy$ denote the aggregate number of job opportunities. The total measure of meetings at time t is given by $M_t = m(\mathcal{W}_t, V_t)$. Define $\mathcal{M}_t^u := M_t/\mathcal{W}_t$ as the rate at which an unemployed searcher contacts a vacancy, and $\mathcal{M}_t^e = \phi\mathcal{M}_t^u$ is the rate at which an employed searcher contacts a vacancy in period t. Let $\mathcal{M}_t^v := M_t/V_t$ denote the rate per unit of recruiting effort $v_t(y)$ that a firm contacts any searching worker.

E.2.2 Value and Surplus Functions

Value of Unemployment: Let $V_t^u(x)$ denote the value of unemployment to a type x worker at t. Let $V_{0,t}^e(x,y)$ be the value to type x worker who is hired from unemployment by a firm of type y. Lise and Robin (2017) assume the worker has no bargaining power so $V_{0,t}^e(x,y) = B_t(x)$ for all y. So, the HJBE for the unemployed worker is:

$$\begin{aligned}\rho V_t^u(x) &= b(x) + \mathcal{M}_t^u \int \alpha_t(x,\tilde{y})(V_{0,t}^e(x,\tilde{y}) - V_t^u(x))\frac{v_t(\tilde{y})}{\mathcal{V}_t}d\tilde{y} + \partial_t V_t^u(x) \\ &= b(x) + \partial_t V_t^u(x)\end{aligned}$$

where $\alpha_t(x,y) = 0.5$. In recursive form, we have $V_t^u(x) = V^u(x,z_t)$ and so the HJBE becomes:

$$\rho V_t^u(x,z_t) = b(x) + \sum_{z'} \pi(z,z')(V^u(x,z) - V^u(x,z))$$

Value and Surplus of a Match: Let $P_t(x,y)$ denote the present value of a match (x,y), including the continuation values to worker and the firm upon separation (which in our notation would be $P_t(x,y) = V_t^e(x,y) + V_t^p(x,y)$.) Let $\tilde{P}_t(x,y,\tilde{y})$ denote the value to the incumbent firm and the worker after the worker moves to a new firm of type $\tilde{y}$. Then, $P_t(x,y)$ solves the HJBE:

$$\begin{aligned}\rho P_t(x,y) = F(x,y,z_t) &+ \phi\mathcal{M}_t^u \int \alpha_t^e(x,y,\tilde{y})(\tilde{P}_t(x,y,\tilde{y}) - P_t(x,y))\frac{v_t(\tilde{y})}{V_t}d\tilde{y} \\ &+ (\delta + \eta\alpha^b(x,y))(V_t^u(x) - P_t(x,y)) + \partial_t P_t(x,y)\end{aligned}$$

If $P_t(x,\tilde{y}) > P_t(x,y)$, the worker moves and they get the incumbent firm's value. So, after the move the incumbent firm has zero value and the incumbent worker has value $P_t(x,y)$, which implies

$$\tilde{P}_t(x,y,\tilde{y}) = 0 + P_t(x,y)$$

If $P_t(x,\tilde{y}) < P_t(x,y)$, the worker does not move and gets $P_t(x,\tilde{y})$. This redistributes surplus towards the worker but does not change the overall value to workers and firms combined, which implies that $\tilde{P}_t(x,y,\tilde{y}) = P_t(x,y)$.

In summary, the second term in the HJBE is always zero, so we get:

$$\rho P_t(x,y) = F(x,y,z_t) + (\delta + \eta\alpha^b(x,y))(V_t^u(x) - P_t(x,y)) + \partial_t P_t(x,y)$$

Consider the surplus, which is defined as:

$$S_t(x,y) := P_t(x,y) - V_t^v(y) - V_t^u(x)$$

In Lise and Robin (2017), their free entry condition implies that $V_t^v(y) = 0$ for all y, so

$$S_t(x,y) = P_t(x,y) - V_t^u(x)$$

Putting the HJBEs together, we have

$$\begin{aligned}\rho S_t(x,y) &= \rho(P_t(x,y) - V_t^u(x)) \\ &= F(x,y,z_t) + (\delta + \eta\alpha^b(x,y))(V_t^u(x) - P_t(x,y)) + \partial_t P_t(x,y) \\ &\quad - b(x) - \partial_t V_t^u(x) \\ &= F(x,y,z_t) - b(x) - (\delta + \eta\alpha^b(x,y))S_t(x,y) + \partial_t S_t(x,y)\end{aligned} \tag{E.1}$$

Equation (E.1) does not depend upon g and so the surplus is "block recursive"—it can be solved without knowing the distribution. This means that, in recursive form, we have the surplus $S(x,y,z)$ satisfies:

$$\begin{aligned}\rho S(x,y,z) = F(x,y,z) - b(x) - (\delta + \eta\alpha^b(x,y))S(x,y,z) \\ + \sum_{\tilde{z}} \pi(z,\tilde{z})(S(x,y,\tilde{z}) - S(x,y,z))\end{aligned}$$

KFE: The distribution of matches evolves according to:

$$\begin{aligned} dg_t(x,y) = & -(\delta + \eta\alpha_t^b(x,y))g_t(x,y)dt \\ & - \phi\frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{W}_t}\int \alpha_t^e(x,y,\tilde{y})\frac{g_t^v(\tilde{y})}{\mathcal{V}_t}d\tilde{y}g_t(x,y)dt \\ & + \frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{W}_t}\alpha_t(x,y)\frac{g_t^v(y)}{\mathcal{V}_t}g_t^u(x)dt \\ & + \phi\frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{W}_t}\int \alpha_t^e(x,\tilde{y},y)\frac{g_t^v(\tilde{y})}{\mathcal{V}_t}g_t(x,\tilde{y})d\tilde{y}dt \end{aligned}$$

where the $\alpha_t(x,y)$ are calculated from the surplus terms as in the main text.

Vacancies: Vacancies are pinned down by the cost of creation via:

$$c'[g_t^v(y)] = \mathcal{M}_t^v J(y,z)$$

and where $\mathcal{M}_t^v = \frac{m(\mathcal{W}_t,\mathcal{V}_t)}{\mathcal{V}_t}$ and $\mathcal{V}_t = \int g_t^v(y)dy$ and:

$$\begin{aligned} J(y,z) = & \int \frac{g_t^u(\tilde{x})}{\mathcal{W}_t}\alpha^b(\tilde{x},y,z)S(\tilde{x},y,z)d\tilde{x} \\ & + \phi\int\int \frac{g(\tilde{x},\tilde{y})}{\mathcal{W}_t}\alpha^e(\tilde{x},\tilde{y},y,z)(S(\tilde{x},y,z) - S(\tilde{x},\tilde{y},z))d\tilde{x}d\tilde{y} \end{aligned}$$

F Additional Details For The OTC Model in Section 5

F.1 Numerical Illustration

We now consider a calibration of the model that draws on Weill (2008), Chen, Cui, He, and Milbradt (2017), Payne and Szőke (2024), and incorporates our agent and asset specification.

F.1.1 Parameters

Economic parameters: We consider an environment with four types of agents: $\{A, B, C, D\}$, where type A are interpreted as dealers in the primary bond market, type B are interpreted as liquidity constrained hedge funds, type C are non-liquidity constrained

hedge funds, and type D are pension/insurance funds with a long investment horizon. Formally, the matrices for holding costs, $\psi(i,\tau)$, switching rates, $\lambda_{ij}(z)$, and primary market participation, $\xi(i,\tau)$, are given in Tables 9, 10, and 11 respectively. The dealer agents (type A) are the only agents who are assigned assets in the primary market. They do not get a net benefit from holding the asset but instead only from trading the asset. The hedge funds randomly switch between getting net benefit from holding any asset (type C) and getting net loss from holding all assets (type B). In this sense, they face the risk of becoming "liquidity constrained" and highly incentivized to sell assets. The pension/insurance funds face a penalty for holding short maturity assets, interpreted as a regulatory constraint on short asset exposure.

		Maturity (τ)			
		$\tau_1 = 0.25$	$\tau_2 = 1.0$	$\tau_3 = 5$	$\tau_4 = 10$
Agent Type (i)	A	$\delta\phi(1,z)$	$\delta\phi(2,z)$	$\delta\phi(3,z)$	$\delta\phi(4,z)$
	B	0.02	0.02	0.02	0.02
	C	0.0	0.0	0.0	0.0
	D	0.02	0.02	0.01	0.00

Table 9: Holding costs: $\psi(i,\tau)$.

Agent Type (j)

	A	B	C	D
A				
B			0.1	
C		0.7		
D				

(a) $\lambda(i,j)$ for $z = z_L$.

Agent Type (j)

	A	B	C	D
A				
B			0.1	
C		0.5		
D				

(b) $\lambda(i,j)$ for $z = z_M$.

Agent Type (j)

	A	B	C	D
A				
B			0.1	
C		0.3		
D				

(c) $\lambda(i,j)$ for $z = z_H$.

Table 10: Switching rates $\lambda(i,j)$ across different aggregate states.

		Maturity (τ)			
		$\tau_1 = 0.25$	$\tau_2 = 1.0$	$\tau_3 = 5$	$\tau_4 = 10$
Agent Type (i)	A	ξ_1	ξ_2	ξ_3	ξ_4
	B	–	–	–	–
	C	–	–	–	–
	D	–	–	–	–

Table 11: Primary market participation: $\xi(i, \tau)$.

We consider the following matching rates, which specify that agents can trade more quickly with the dealers than with each other (following Chen, Cui, He, and Milbradt (2017)):

$$\kappa_{a,b} = \begin{cases} 50, & \text{if } (a,b) = (in, jok) \text{ and } i, j \neq A, \\ 50, & \text{if } (a,b) = (iok, jok) \text{ and } i, j \neq A, \\ 75, & \text{if } (a,b) = (in, Aok) \text{ and } i \neq A, \\ 0, & \text{if } (a,b) = (iok, Aol) \text{ and } \forall i, \\ 0, & \text{if } (a,b) = (in, jn) \text{ and } \forall i, j, \end{cases}$$

We impose that agents have equal bargaining power unless they match with a dealer, in which case they have bargaining power 0.05 (following Weill (2008) and Chen, Cui, He, and Milbradt (2017)):

$$\beta_{a,b} = \begin{cases} 0.5, & \text{if } (a,b) = (in, jok) \text{ and } i, j \neq A, \\ 0.5, & \text{if } (a,b) = (iok, jol) \text{ and } i, j \neq A, \\ 0.05, & \text{if } (a,b) = (in, Aok) \text{ and } i, j \neq A, \end{cases}$$

The other economic parameters are listed in Table 12. We calibrate the model at the annual frequency. Where possible, we take standard parameters from the literature.

Neural network parameters: We describe the details of the neural network approximation and sampling in Table 7. We use a fully connected feed-forward network with 4 layers, 100 neurons per layer, and a combination of $tanh(\cdot)$ activation functions. The training loss is shown in Figure 12.

Parameter	Interpretation	Value	Target/Source
ρ	Discount rate	0.05	Chen, Cui, He, and Milbradt (2017)
δ	Bond Coupon Rate	0.01	
Aggregate State: $z \in \{z_L, z_M, z_H\}$			
$\phi(z)$	Coupon haircut	$(0.986, 0.991, 0.997)$	Chen, Cui, He, and Milbradt (2017)
$\pi(z)$	Principal haircut	$(0.986, 0.991, 0.997)$	Chen, Cui, He, and Milbradt (2017)
$\zeta_{M,L}$, $\zeta_{M,H}$	Rate from 2 to 1 and 2 to 3	0.1	Crisis every 10 years
$\zeta_{L,M}$, $\zeta_{H,M}$	Rate from 1 to 2 and 3 to 2	0.5	Average crisis duration 2 years

Table 12: Economic Parameters.

Parameter	Value
Number of layers	8
Neurons per layer	100
Activation function	GELU$(\cdot)$
Initial learning rate	10^{-4}
Final learning rate	10^{-6}
Initial sample size per epoch	256
Sample size per epoch	1024
Convergence threshold for target calibration	10^{-6}

Table 13: Neural network parameters

Figure 12: Loss function along training epochs