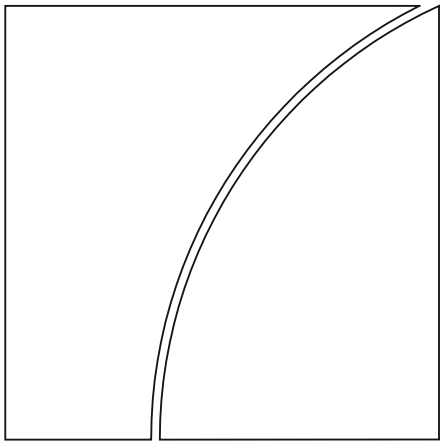

BIS Working Papers
No 1312

Generative economic modeling

by Hanno Kase, Matthias Rottner and Fabio Stohler

Monetary and Economic Department

December 2025

JEL classification: C11, C45, D31, E32, E52

Keywords: machine learning, neural networks, nonlinearities, heterogeneous agents

BIS Working Papers are written by members of the Monetary and Economic Department of the Bank for International Settlements, and from time to time by other economists, and are published by the Bank. The papers are on subjects of topical interest and are technical in character. The views expressed in this publication are those of the authors and do not necessarily reflect the views of the BIS or its member central banks.

This publication is available on the BIS website (www.bis.org).

ISSN 1020-0959 (print)
ISSN 1682-7678 (online)

Generative Economic Modeling*

Hanno Kase
European Central Bank

Matthias Rottner
Bank for International Settlements
Deutsche Bundesbank

Fabio Stohler
University of Bonn

December 2, 2025

Abstract

We introduce a novel approach for solving quantitative economic models: generative economic modeling. Our method combines neural networks with conventional solution techniques. Specifically, we train neural networks on simplified versions of the economic model to approximate the complete model's dynamic behavior. Relying on these less complex submodels circumvents the curse of dimensionality, allowing the use of well-established numerical methods. We demonstrate our approach across settings with analytical characterizations, nonlinear dynamics, and heterogeneous agents, employing asset pricing and business cycle models. Finally, we solve a high-dimensional HANK model with an occasionally binding financial friction to highlight how aggregate risk amplifies the precautionary motive.

Keywords— Machine Learning, Neural networks, Nonlinearities, Heterogeneous Agents
JEL codes— C11, C45, D31, E32, E52

*hanno.kase@ecb.europa.eu, matthias.rottner@bis.org, fabio.stohler@uni-bonn.de. We would like to thank Christian Bayer, Jesús Fernández-Villaverde, Alexandros Gilch, Keith Kuester, Jamie Lenney, Leonardo Melosi, Simon Scheidegger, and Yucheng Yang for their very helpful suggestions. We thank seminar participants at the Barcelona Summer Forum session on Machine Learning in Economics, the Conference on Machine Learning for Economics and Finance, Banque de France AI Methods Conference 2025, the Bank for International Settlements, the Dynare Conference 2025, and the ECONDAT 2025 Spring meeting for helpful comments and discussions. Fabio Stohler gratefully acknowledges support by the German Research Foundation (DFG) through CRC TR 224 (Project C05). The views in this paper are those of the authors and should not be interpreted as reflecting the views of the Bank for International Settlements, the Deutsche Bundesbank, the European Central Bank, or any other person associated with the Eurosystem.

1 Introduction

The advances in artificial intelligence provide substantial opportunities for quantitative economics by shifting the production-possibility frontier of modeling. Deep learning has emerged as a powerful tool for solving dynamic economic models that were previously considered intractable. The curse of dimensionality - first articulated by Bellman (1957) - typically limits the complexity of economic models to only a few state variables. However, deep learning can help to tame this problem, as discussed in Fernández-Villaverde, Nuño and Perla (2024). Unfortunately, successfully employing deep learning in practice often requires meticulous and detailed adjustments tailored to the specific model at hand, which makes it challenging to employ without major investments. In contrast, established conventional solution methods, though constrained by the curse of dimensionality, are already specifically designed and optimized for particular types of economic models and feature well-understood emergence properties. We propose a novel approach that combines the strengths of both artificial intelligence and conventional solution methods: generative economic modeling.

Our approach employs neural networks to approximate the economic model. Rather than training directly on the full model, we train the network on a collection of simplified models, which we call submodels. Each submodel contains only a subset of states and features and therefore represents only part of the full model's dynamics.[1] By relying on submodels, we can solve the simpler problems with conventional methods and avoid the curse of dimensionality. Because each submodel captures only some features, we design the collection with overlapping features to ensure coverage of all features present in the full model. We simulate each submodel separately and merge the resulting data into a single training set. A neural network is then trained on this combined dataset to approximate the full model and recover interactions across states and features.[2]

Our method belongs to the class of generative artificial intelligence because we employ a neural network to generate results for the complete model that includes all features and states, something we have not used for the training process. Generative artificial intelligence has achieved significant success, especially in the context of large language models, which are very large deep learning models. While this success also holds promise for our approach, the generative performance of neural networks in our context of economic modeling is a priori not clear.

[1] Researchers often work with a simplified version of the model of interest because computational constraints limit feasible complexity. Our approach formalizes how to leverage a set of submodels and apply deep learning to approximate a richer model.

[2] The neural network serves as a surrogate model that mimics the behavior of a more complex system.

We validate our approach by computing Euler equation errors to assess the accuracy of the resulting model solution. Specifically, we evaluate the errors in the complete model with all shocks and features while using policy functions learned from networks trained only on a collection of submodels with limited features. This delivers an out-of-sample check against the optimality conditions of the full model without having to solve the full model with conventional methods. Moreover, this metric can provide guidance in designing submodels and specifying the neural network architecture. Rather than providing a fixed set of rules or formal proofs, our approach follows the data science tradition: we can evaluate and experiment to identify which combination of submodels yields the best approximation.

As a first practical demonstration, we study a simple and analytically tractable asset pricing model. We choose this setting because it allows us to show analytically how the full model can be approximated through a set of submodels. The controlled environment serves to illustrate the methodology and provides a necessary first validation step.

We find that our generative modeling approach closely matches the analytical solution. Since analytical solutions are not available for most applications, we also evaluate accuracy through Euler equation errors. The errors in our approach are very low and comparable to those obtained when a network is trained directly on data simulated from the full model. This indicates that our method generates coherent full-model behavior from submodel building blocks and recovers the dynamics of a richer model with more features. In contrast, training on data from a single submodel produces sizable approximation errors and highlights the advantage of our methodology over the common practice of extrapolating from small models to larger ones.

As a next step, we evaluate our method in a more challenging environment. We use different variants of a non-linear real business cycle (RBC) model to assess how well our method provides a global solution. Specifically, we work with three non-linear versions: i) a simplified version that can be solved analytically, ii) a medium-sized version with state-dependent investment costs that result in distinct nonlinearities, and iii) a heterogeneous agents version with partially uninsurable income risk in line with Krusell and Smith (1998). While the variants are increasing in their complexity, we can still rely on conventional solution methods to solve the complete model. Thus, we can evaluate how well our approach is designed to capture nonlinearities and how applicable it is to the class of heterogeneous agent versions, benchmarking it against the complete model. Similarly, we exemplify how to use the Euler equation error as a measure of fit in different setups.

To solve these models with generative economic modeling, we construct for all these models overlapping submodels and then use simulated data to train our neural network.

Our methodology provides a very good approximation in all three versions of the model. When compared with neural networks trained directly with the complete model, the magnitude of these errors is in the same range.[3] Similarly, the Euler equation errors are low and have the same magnitude, highlighting the robustness and accuracy of our approach. This holds for the aggregate dynamics in the analytical model, the heterogeneous agent version as well as the nonlinear dynamics in the medium-sized model with piecewise nonlinear capital adjustment costs. By accelerating the solution of complicated models and enabling the analysis of very hard-to-solve economic environments, our approach provides a powerful tool for advancing research on nonlinear, heterogeneous, and more generally, high-dimensional economic models.

Finally, we apply generative economic modeling to solve a complex Heterogeneous Agent New Keynesian (HANK) model with multiple aggregate shocks and financial frictions. The financial friction prevents firms from hiring as many workers as they desire and introduces a nonlinearity in the model. Solving this class of models is computationally challenging, as the dimensionality of the household state space increases with each additional aggregate shock.[4] Additionally, heterogeneous agent models with multiple aggregate shocks face a threefold curse of dimensionality. First, expanding the state space significantly increases the computational complexity of solving the household problem. Second, incorporating additional shocks complicates the accurate computation of expected values. Third, introducing more states necessitates a more involved calculation of the perceived laws of motion to forecast the future evolution of payoff-relevant aggregate variables.

Our methodology addresses these challenges by effectively reducing the state space by solving smaller submodels, which are then collectively used as inputs to train the neural network. Specifically, we construct a collection of nonlinear submodels, each of which includes the financial friction but only a subset of the aggregate shocks. We then solve and simulate these submodels using a version of the global solution method developed by Krusell and Smith (1998), extended to accommodate multiple shocks. Subsequently, we merge the simulated data from the submodels and train a neural network on the combined dataset to approximate the dynamics of the full model, with all shocks and frictions simultaneously active.

[3] The positive approximation errors for the complete model come from training the neural network instead of working directly with the complete model solution.

[4] Several studies have proposed different approaches and modifications to solve such models; see, for example, Algan, Allais and Den Haan (2008), Reiter (2009), Den Haan (2010), Gornemann, Kuester and Nakajima (2016), Ahn et al. (2017), Boppart, Krusell and Mitman (2018), Bayer and Luetticke (2020), Auclert et al. (2021), and Bayer, Born and Luetticke (2024). Despite these advances, solving fully specified HANK models with multiple nonlinear frictions and shocks remains numerically intractable with conventional solution methods.

The analysis delivers two central insights about the transmission of financial shocks in a HANK model with financial frictions. First, the model highlights how the presence of additional aggregate shocks alters the transmission of any single shock. For instance, the impact of a financial shock is attenuated in a setting with more aggregate shocks. This dampening arises from increased aggregate uncertainty, which strengthens agents' precautionary motives. With more sources of risk, households anticipate greater volatility and self-insure more aggressively through higher precautionary savings. As a result, when a shock hits, they are better prepared, and the overall economic response is less pronounced. Second, the model exhibits strong nonlinear effects in response to a financial shock. While all sizes of negative shocks result in a downturn of the model economy, these effects become much stronger when the financial shock is larger. Larger shocks constrain firms more in their hiring decision, strongly reducing households' labor income, which triggers a larger downturn.

As initial motivation for our approach, we emphasized the potential fragility of deep learning when applied directly to solve the economic model. Our method avoids this problem due to a key difference. In our approach, the training data for the neural network is precalculated based on conventional solution methods and is therefore not endogenously affected by the training of the neural network. By contrast, when deep learning is applied directly to solving economic equations, the inputs used usually depend on the model solution generated by the neural network itself. In this case, the inputs are endogenous rather than exogenous, as they rely on the output of the neural network. This feedback loop makes finding a solution substantially more difficult — an issue that our generative economic modeling approach avoids.

Literature Review Our paper belongs to the fast-growing literature that uses deep learning to solve dynamic economic models. The areas of application have been HANK models (Fernández-Villaverde et al., 2024; Kase, Melosi and Rottner, 2022), heterogeneous agents (Azinovic, Gaegauf and Scheidegger, 2022; Azinovic-Yang and Žemlička, 2025; Fernández-Villaverde, Hurtado and Nuno, 2023; Gorodnichenko et al., 2021; Gu et al., 2024; Han, Yang and E, 2021; Kahou et al., 2021; Maliar and Maliar, 2022; Maliar, Maliar and Winant, 2021), overlapping generations and life-cycle (Azinovic-Yang and Žemlička, 2024; Druedahl and Røpke, 2025; Pascal, 2024), finance (Chen, Didisheim and Scheidegger, 2023; Duarte, Duarte and Silva, 2024; Duarte and Fonseca, 2024; Valaitis and Villa, 2024), labor markets and search (Adenbaum, Babalievsky and Jungerman, 2024; Jungerman, 2024; Payne, Rebei and Yang, 2024), monetary policy (Chen et al., 2021; Nuño, Renner and Scheidegger, 2024), climate change (Fernández-Villaverde, Gillingham and

Scheidegger, 2024; Friedl et al., 2023; Kübler, Scheidegger and Surbek, 2025), and behavorial macroeconomics (Ashwin, Beaudry and Ellison, 2025; Kahou et al., 2024). However, our approach to using neural networks deviates strongly from these papers, as we are not interested in directly solving the economic equations. More closely related to our work is the usage of neural networks as surrogate models as in Kase, Melosi and Rottner (2022) and Chen, Didisheim and Scheidegger (2023). Yet, these papers use the complete underlying model for the training, thereby excluding the generative part. Finally, we also differ by following a hybrid approach that exploits the advantages of conventional solution methods and deep learning.

Our methodology allows us to speak to the large literature on HANK models (see e.g., McKay and Reis, 2016, McKay, Nakamura and Steinsson, 2016 Den Haan, Rendahl and Riegler, 2018, Guerrieri and Lorenzoni, 2017, Ravn and Sterk, 2017, Kaplan, Moll and Violante, 2018, Kaplan and Violante, 2018, Auclert, 2019, Bayer et al., 2019, Ravn and Sterk, 2020, Gornemann, Kuester and Nakajima, 2016, Luetticke, 2021, Auclert, Rognlie and Straub, 2024, Bayer, Born and Luetticke, 2024). We show that aggregate risk substantially increases the precautionary motive. Even though recent work in this literature has increasingly focused on incorporating aggregate risk and non-linearities by approximating the dynamics to higher-orders (Bhandari et al., 2023, Bayer et al., 2024), adding occasionally binding constraints (Lin and Peruffo, 2024) or focusing on uncertainty (Ilut, Luetticke and Schneider, 2025), these methods do not attempt to solve the model globally. Only very few papers can solve a medium-scale HANK model globally, and they do not use conventional solution methods, except Schaab (2020). Instead, our approach provides a solution that combines an adapted version of the widely used global solution method of Krusell and Smith (1998) with neural networks to model nonlinearities and heterogeneity jointly.[5] Our work is also directly related to integrating financial frictions and shocks in HANK models (see e.g., Fernández-Villaverde, Hurtado and Nuno, 2023, Faccini et al., 2024, Nord, Peruffo and Mendicino, 2024).

Our work also builds on the broader literature on conventional solution methods that do not rely on deep learning. Given the vast array of contributions across different fields of computational economics, providing a comprehensive review would be infeasible. Instead, we refer to the influential books on numerical methods by Judd (1998), Miranda and Fackler (2004) and Heer and Maussner (2024), which provide a great overview of the methods available. Our approach builds upon these traditional methods while leveraging deep learning to enhance their capacity to handle higher levels of complexity. The combination

[5] For instance, Meriküll and Rottner (2025) show empirically that the distributional effects of monetary policy are nonlinear.

of the methods makes it possible to tackle problems that were previously computationally intractable.

2 Generative Economic Modeling

This section outlines our generative economic modeling approach, which is designed to solve a large class of dynamic general equilibrium models.[6]

2.1 Underlying Complete Dynamic General Equilibrium Model

The dynamics of a dynamic general equilibrium model can be expressed as a transition equation:

$$\mathbb{S}_t = f\Big(\mathbb{S}_{t-1}, \nu_t, \psi(\mathbb{S}_{t-1}, \nu_t)|\Theta\Big) \tag{1}$$

where the state vector $\mathbb{S}_t \in \mathbb{R}^m$ describes the economy in period t. Note that such a representation can contain heterogeneous agents or behavioral models. The economy is also subject to exogenous shocks that follow a Markov process, which is captured by the vector $\nu_t \in \mathbb{R}^n$. There is also a vector of structural parameters $\Theta \in \mathbb{R}^d$, which affects the dynamics of the model. The function $f(\cdot)$ determines the mapping from the previous period state variables $\mathbb{S}_{t-1}$ and current period shocks ν_t to the current period state variables $\mathbb{S}_t$ conditional on the structural parameters. To solve the transition equation, it is needed to find the policy function (decision function) $\psi(\mathbb{S}_{t-1}, \nu_t)$, which maps the model state variables $\mathbb{S}_{t-1}$ and ν_t to a set of choices ψ_t. For notational convenience, we directly include the policy function in the transition equation.

This mapping from the state variables to the policy function is usually unknown and needs to be solved with numerical methods. Luckily, there already exists a large set of solution methods - more general solution approaches, like value function iteration, policy function iteration, or the endogenous grid point method, and very tailored solution methods, such as the Krusell-Smith approach for heterogeneous agent economies with aggregate risk. In general, the idea is to find an equilibrium function that maps the state variables to a set of control variables, $\psi_t = \psi(\mathbb{S}_{t-1}, \nu_t|\Theta)$. These policy functions satisfy a set of equations derived from the model.

$$\mathbb{F}(\psi_t) = 0 \tag{2}$$

Once equipped with these policy functions, we can solve for the transition equation.

[6] We focus on dynamic Markov economic models, where agents solve a Markov decision problem, as e.g. in Maliar, Maliar and Winant (2021) and Fernández-Villaverde, Nuño and Perla (2024).

The advantage of our approach is that we keep the conventional policy-function solution steps unchanged, since they have been refined for years for specific problems. However, such conventional global solution methods face the curse of dimensionality. The exponential growth of the grid points of the state space as the number of states increases limits the complexity of the economic model that can be considered. For instance, solution methods, which use full grid-based approaches with G points per dimension, have exponential computing costs, as shown by the number of grid points $\mathcal{O}(G^{m+n})$.[7] Importantly, this problem already occurs for a small number of states.

2.2 Submodels of the Complete Model

The curse of dimensionality often forces the modeler to reduce the complexity of the studied model by limiting the number of state variables and shocks. In other words, a simplified *submodel* is derived from the underlying complete model.[8] In practice, using a submodel instead of the most comprehensive model available is mostly the norm when working with global solution methods.[9] In that regard, economists are well-trained to use submodels, and it is likely the common approach.

We denote the variables in a submodel version with $\sim$ and rewrite the transition equation of the submodel as:

$$\tilde{\mathbb{S}}_t = \tilde{f}\left(\tilde{\mathbb{S}}_{t-1}, \tilde{\nu}_t, \tilde{\psi}_t | \tilde{\Theta}\right) \tag{3}$$

where the dimension of the states $\tilde{\mathbb{S}}_t \in \mathbb{R}^{\tilde{m}}$ and shocks $\tilde{\nu}_t \in \mathbb{R}^{\tilde{n}}$ is smaller than in the full model, that is $(\tilde{m} < m) \vee (\tilde{n} < n)$. Note that the set of structural parameters $\tilde{\Theta} \in \mathbb{R}^{\tilde{d}}$ is then also potentially smaller, that is $\tilde{d} \leq d$.[10]

However, we can now specify not only one submodel, but instead several submodels that capture different elements of the underlying model, that is $\tilde{\mathbb{S}}_t^a, \tilde{\mathbb{S}}_t^b, \tilde{\mathbb{S}}_t^c, \ldots$, where the superscript indicates the submodel:

$$\tilde{\mathbb{S}}_t^i = \tilde{f}^i\left(\tilde{\mathbb{S}}_{t-1}^i, \tilde{\nu}_t^i, \tilde{\psi}_t^i | \tilde{\Theta}^i\right), \text{ for } i = a, b, c, \ldots \tag{4}$$

[7] Note that refinements to the solution method can lower the computational costs, e.g. adaptive sparse grids (see Brumm and Scheidegger (2017)).

[8] This approach is related - though conceptually distinct - to dimensional decomposition, in which the behavior of a high-dimensional function is represented as a sum of lower-dimensional functions. For applications in economics to reduce the curse of dimensionality, see Eftekhari and Scheidegger (2022) and Eftekhari et al. (2025).

[9] Models that are solved with perturbation methods are usually much larger than models solved with global solution methods, however, there also exist limits on the size of the problems. In the HANK literature, the size of the household problem is such a limiting factor.

[10] We impose that each submodel in itself is still a general equilibrium model.

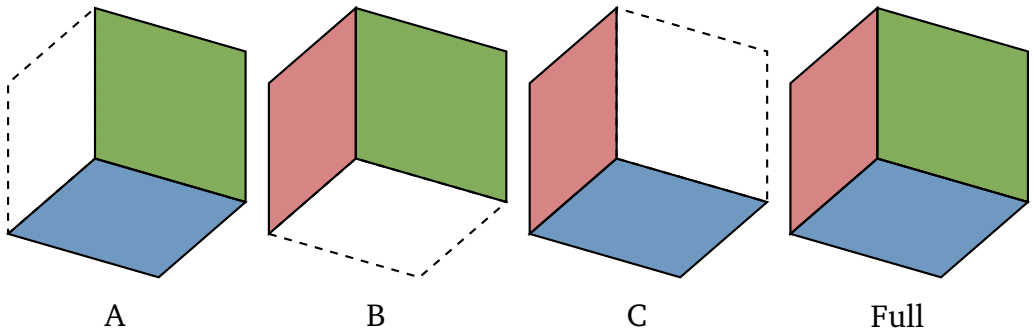

Figure 1 Illustration of the concept of submodels, where we have three submodels that cover a different subset of states from the full model, as marked with different colours (green, red, and blue). In total, we have three submodels (A, B, and C), capturing all possible subset combinations of the full model.

All these submodels together form a set $\bar{f}$:

$$\bar{f}\left(\mathbb{S}_{t-1}, \nu_t, \psi_t | \tilde{\Theta}\right) = \left\{ \tilde{f}^a \left(\tilde{\mathbb{S}}^a_{t-1}, \tilde{\nu}^a_t, \tilde{\psi}^a_t | \tilde{\Theta}^a\right), \tilde{f}^b \left(\tilde{\mathbb{S}}^b_{t-1}, \tilde{\nu}^b_t, \tilde{\psi}^b_t | \tilde{\Theta}^b\right), \tilde{f}^c \left(\tilde{\mathbb{S}}^c_{t-1}, \tilde{\nu}^c_t, \tilde{\psi}^c_t | \tilde{\Theta}^c\right), \dots \right\}, \tag{5}$$

where we omitted the dependence of the policy function for conciseness.

Importantly, the numerical costs to extend the set increase linearly instead of exponentially (conditional on keeping the same number of states and shocks for each submodel). The submodels can be specified to be complete and overlapping. We define completeness as each state $\mathbb{S}_t$ and shock ν_t is at least covered in one submodel. Therefore, at least one submodel captures one specific part of the underlying model. This requirement ensures that the set of submodels and the true model have the same states, shocks, and parameters. We define overlap as each state $\mathbb{S}_t$ and shock ν_t should be at least covered in two different submodels, so that the different subsets overlap.

The number of required submodels to achieve completeness and overlap as defined above is determined by the binomial coefficient:

$$\binom{m+n}{\tilde{m}+\tilde{n}} = \frac{(m+n)!}{(\tilde{m}+\tilde{n})!\,(m+n-\tilde{m}+\tilde{n})!}, \tag{6}$$

where $m+n$ and $\tilde{m}+\tilde{n}$ denote the number of states in the full model and the submodel, respectively. For each of the submodels, the number of grid points is now substantially lower due to the reduced dimension, that is $\mathcal{O}(G^{\tilde{m}+\tilde{n}})$.

Illustration of submodels To illustrate the notion of a submodel, consider an economic model with a state vector $\mathbb{S}$ that is too complex to solve in full. Instead, we solve submodels, each based on a subset $\tilde{\mathbb{S}}$. Figure 1 illustrates three submodels, each containing only a subset of the states from the full model. However, we solve the submodels with overlap,

such that each submodel features two of the subsets of states. For instance, submodel A includes the blue and green subset of states, while others incorporate different combinations. In total, we have three submodels (A, B, and C), capturing all possible subset combinations.

2.3 Deep Learning Approach to Reconstruct the Full Model

Each submodel is a subset of the true underlying model, that is

$$\tilde{f}^i\left(\tilde{\mathbb{S}}^i_{t-1}, \tilde{\nu}^i_t, \tilde{\psi}^i_t|\tilde{\Theta}\right) \subset f\left(\mathbb{S}_{t-1}, \nu_t, \psi_t|\Theta\right), \forall i = a, b, c, \ldots \tag{7}$$

The idea of this paper is to evaluate whether a rich specified set of submodels is sufficient to approximate the true underlying model. Although the set of submodels is complete and overlapping, the submodels are only partial representations of the full models. For this reason, we want to use deep learning to learn the underlying dynamics of the full model from the set of submodels, leveraging deep learning's generative capacity. We approximate the transition equation, including the policy function, of the submodels using a *surrogate model* in the form of a deep neural network $\bar{f}_{NN}$ such that

$$\bar{f}_{NN}\left(\mathbb{S}_{t-1}, \nu_t, \psi_t|\Theta\right) \approx \bar{f}\left(\mathbb{S}_{t-1}, \nu_t, \psi_t|\Theta\right). \tag{8}$$

Below, we illustrate the individual steps of the analysis, and delegate the approach of assessing whether the neural network accurately approximates the true model's dynamics to the next subsection.

Steps of the generative economic modeling approach Our generative economic modeling approach proceeds in three steps, as illustrated graphically in Figure 2.

1. First, we solve the simplified submodels, ensuring completeness and overlap of the submodels. The choice of solution algorithm is left to the researcher, as our approach is compatible with any method that can solve the dynamics of the model.[11]

2. Second, we simulate each submodel separately to create the data series for the different variables. We then prepare the dataset by merging all simulations, creating a long data series in which only a specific subset of states and shocks is active in different

[11] While any solution method can be used, the methodology yields optimal performance when applied to solution techniques that minimize approximation errors. The surrogate model's accuracy depends on how well the training data represents the true data generation process. In our applications later on, we employ global solution techniques to solve the model as accurately as possible.

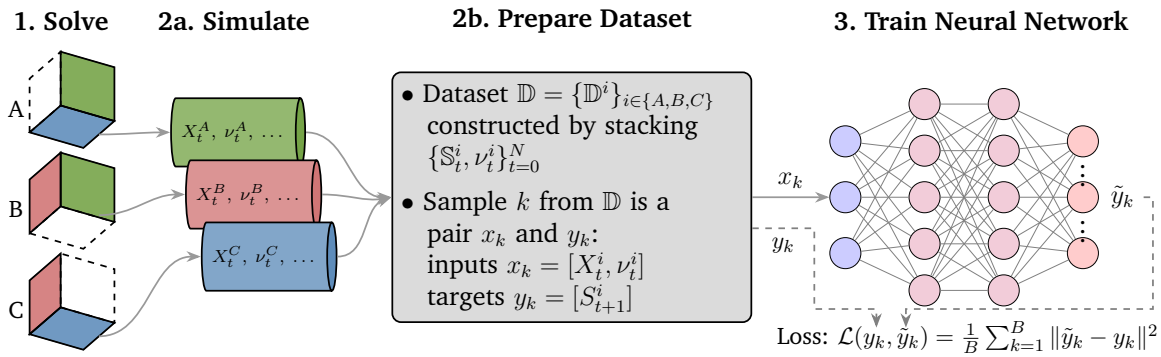

Figure 2 Flow chart of the generative economic modeling method.

periods. The dataset that holds the simulation for all submodels is:[12]

$$\bar{\mathbb{D}} = \left\{ \left\{ \tilde{S}_{t-1}^a, \tilde{\nu}_t^a, \tilde{\psi}_t^a \right\}_{t=1}^N, \left\{ \tilde{S}_{t-1}^b, \tilde{\nu}_t^b, \tilde{\psi}_t^b \right\}_{t=1}^N, \left\{ \tilde{S}_{t-1}^c, \tilde{\nu}_t^c, \tilde{\psi}_t^c \right\}_{t=1}^N, \ldots \right\}. \tag{9}$$

3. Finally, we train the neural network using the datasets from the submodels by minimizing the mean squared errors between the predicted values from the neural network and the observed values from the submodels:[13]

$$\bar{f}_{NN} = \arg\min_{\bar{W}} \mathbb{L}(\bar{W}|\bar{\mathbb{D}}) \tag{10}$$

By training the network on data from multiple submodels, it learns the distinct transmission mechanisms of individual shocks, effectively generating an economic model that integrates the key features of the underlying data. At the same time, some higher-order interaction terms remain unobserved during training because each submodel includes only a subset of features. As a result, the surrogate approximation will contain some residual error. However, this error decreases as the number of features per submodel increases. Additionally, in many economic models the contribution of higher-order interaction terms—such as those arising from multiple shocks—tends to diminish. Consequently, even when trained on submodels with a limited number of features or shocks, the surrogate is expected to achieve high accuracy for most applications. In what follows, we demonstrate how to evaluate whether the neural network approximates to a high degree of accuracy the complete model's dynamics by leveraging insights from its individual components.

[12] Because our set of submodels is complete, this training data covers all states and shocks, that is $\mathbb{S}_t$ and ν_t.
[13] We also divide the collected dataset into a training and validation sample.[14]

2.4 Ex-post Validation: Euler Equation Errors as Criteria of Fit

To evaluate the accuracy of our generative modeling approach, we need to evaluate whether it approximates the dynamics of the true model. Hence, we need to test whether for the true states $\mathbb{S}_t$, shocks ν_t and parameters Θ it holds that

$$\bar{f}_{NN}\left(\mathbb{S}_{t-1}, \nu_t, \psi_t | \Theta\right) \approx f\left(\mathbb{S}_{t-1}, \nu_t, \psi_t | \Theta\right). \tag{11}$$

When showcasing our method in the following chapters, we rely mostly on models where we can solve the complete model for illustration purposes and can benchmark the performance of our methodology against the true data-generating process. Hence, we can check whether equation (11) holds approximately.[15]

However, this possibility is, in practice, usually unavailable as the methodology is designed to be applied to a model that is otherwise unsolvable, as in the case of our HANK model with financial frictions. Even though we cannot provide a proof that a sufficiently rich set of submodels approximates the true underlying model, we can directly employ standard methods to check the approximation error. In particular, we can use selected equilibrium conditions and calculate the associated Euler equation errors, as e.g. in Judd (1998) and Aruoba, Fernández-Villaverde and Rubio-Ramirez (2006). Hence, we can check whether the following relation approximately holds:

$$\mathbb{F}\left(\psi_{NN}(\mathbb{S}_{t-1}, \nu_t)\right) \approx 0. \tag{12}$$

where $\psi_{NN}(\cdot)$ denotes the neural network-based approximation of the policy function. The equilibrium conditions can be taken directly from the conventional solution step for the complete model, which we never solve. However, we now use our mapping from the generative neural network to assess the fit. Note that we can also evaluate expectations over variables using methods such as Monte Carlo or Quadrature rules, that have been used for solving the submodels.

3 Illustration of the Method with a Tractable Example

This section illustrates our methodology using a simple, analytically tractable asset pricing model. We choose this model because it illustrates analytically how our method is able to approximate a full model through submodels. Specifically, the higher-order interaction

[15] Since training a neural network introduces errors in the approximation, besides checking the approximation (11), we also check the relative performance of our neural network trained on data from the submodels compared to a neural network trained on the true data generated from the true full model.

terms of the shocks can be characterized analytically, ensuring that the submodels jointly capture all components of the complete model's solution. We interpret success in this environment as a necessary condition for validating our approach before we evaluate the method using more complicated models later.

3.1 Asset Pricing Model and Decomposition into Submodels

The model is a basic asset pricing framework based on Canzoneri, Cumby and Diba (2007). To solve the model, we need to determine the price of a nominal pure discount bond, denoted by q_t, which satisfies the following Euler equation:

$$q_t = \beta \, \mathbf{E}_t \left[\exp(-\pi_{t+1} - \gamma \Delta c_{t+1}) \right], \tag{13}$$

where γ is the coefficient of relative risk aversion, $\pi_t = \log(\Pi_t)$ is the log gross inflation rate, and $\Delta c_t = \log(C_t) - \log(C_{t-1})$ is the change in log consumption. This equation arises from a standard consumption-savings problem in which households price nominal bonds under uncertainty.

To close the model, we assume that log inflation and log consumption growth follow a first-order vector autoregressive process (VAR(1)):

$$y_t = A y_{t-1} + \eta \epsilon_t, \tag{14}$$

where $y_t = [\pi_t, \Delta c_t]'$ is the state vector, $\epsilon_t = [\epsilon_t^a, \epsilon_t^\zeta, \epsilon_t^\mu]'$ is a vector of structural shocks, and η is a 2×3 matrix of shock loadings. We associate each shock with an economic interpretation: a technology (TFP) shock (a), a discount factor shock (ζ), and a markup shock (μ). This labeling is primarily for interpretability and does not affect our results.

Analytical Solution of the Complete Model Given the log-linear structure of the model, we can solve it analytically.[16] Then the full solution for the price of the asset can be written as:

$$\begin{aligned} q_t = \beta \exp \Bigg(& - (a_{11} + \gamma a_{21}) \, \pi_t \; - \; (a_{12} + \gamma a_{22}) \, \Delta c_t \\ & + \tfrac{1}{2} \Big[\underbrace{(\eta_{11} + \gamma \eta_{21})^2}_{\epsilon^a} + \underbrace{(\eta_{12} + \gamma \eta_{22})^2}_{\epsilon^\zeta} + \underbrace{(\eta_{13} + \gamma \eta_{23})^2}_{\epsilon^\mu} \Big] \Bigg) \end{aligned} \tag{15}$$

[16] A detailed derivation is provided in Appendix A.

where a_{ij} denotes the (i,j)-th entry of the matrix A, and η_{ij} denotes the (i,j)-th entry of the impact matrix η. The terms in the square bracket of equations (15) captures the contribution of all present shocks to the conditional variance of the pricing kernel. All terms in the second line represent the effect that fluctuations introduced by the existence of the aggregate shocks have on the pricing kernel, with the contributions directly connected to the specific shocks.

Analytical Solution of the Submodels To use this example for our methodology, we assume that we cannot solve the full model. Instead, we can only solve submodels that have only two of the three shocks. Shutting down an individual shock i is identical to setting $\eta_{i1} = \eta_{i2} = 0$ in (14). The following equations denote the equilibrium prices without a shock i by $q_t^{\setminus i}$:

$$q_t^{\setminus a} = \beta \exp\left(-(a_{11} + \gamma a_{21})\,\pi_t \;-\; (a_{12} + \gamma a_{22})\,\Delta c_t + \tfrac{1}{2}\left[(\eta_{12} + \gamma\eta_{22})^2 + (\eta_{13} + \gamma\eta_{23})^2\right]\right) \quad (16)$$

$$q_t^{\setminus \zeta} = \beta \exp\left(-(a_{11} + \gamma a_{21})\,\pi_t \;-\; (a_{12} + \gamma a_{22})\,\Delta c_t + \tfrac{1}{2}\left[(\eta_{11} + \gamma\eta_{21})^2 + (\eta_{13} + \gamma\eta_{23})^2\right]\right) \quad (17)$$

$$q_t^{\setminus \mu} = \beta \exp\left(-(a_{11} + \gamma a_{21})\,\pi_t \;-\; (a_{12} + \gamma a_{22})\,\Delta c_t + \tfrac{1}{2}\left[(\eta_{11} + \gamma\eta_{21})^2 + (\eta_{12} + \gamma\eta_{22})^2\right]\right) \quad (18)$$

As each submodel lacks one shock (i.e. $\eta_{i1} = \eta_{i2} = 0$ for a shock i), the variance term varies from submodel to submodel.[17]

Mapping of the Solutions The submodels together feature all elements from the impact matrix of the full underlying model.[18] Each submodel features two of the pricing terms for aggregate risk, as can be seen in the equations for the submodels (16) - (18). Additionally, each submodel contains the terms for the direct impact of inflation and log consumption growth on the asset price.

The idea of our approach is that during training, the surrogate neural network learns these pricing coefficients from the data of each submodel in the combined dataset. The surrogate learns these submodel specific coefficients because the pooled dataset includes zero-shock realizations that let it distinguish observations by submodel. Overlapping simulations from different submodels ensures coverage of all coefficients and shock interactions such that we are able to approximate the solution of the complete model in equation (15).

[17] Note also that the VAR(1) process for π_t and Δc_t differs, since these series are affected by only two shocks in the submodels, compared to three shocks in the true model.

[18] This property is not a general feature of models, since it relies on the submodels collectively featuring all terms of the full model. In our case, we have that all terms above the second order are zero, so that the combination of submodels fully captures it.

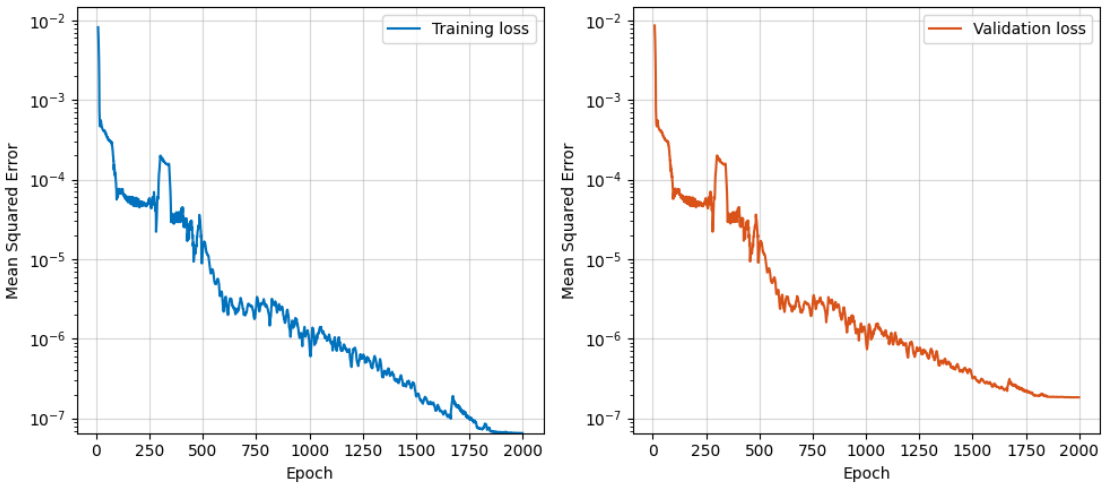

Figure 3 Training and validation convergence for the asset pricing model. The figure shows the loss over the training sample (left) and validation sample (right).An epoch is completed when all the training or validation sample points are utilized. The vertical axis is expressed in logarithmic scale.

3.2 Data Simulation and Dataset Generation

To generate our datasets, we simulate synthetic data for three submodels separately. Specifically, we simulate each model for 20000 periods by drawing random realizations of the shock vector ϵ_t and use the analytical solution to simulate the variables forward. Specifically, we generate paths for inflation and log consumption growth using equation (14), and compute the corresponding bond prices using equations (16) - (18), depending on the submodel. For each submodel where shock i is inactive, we obtain simulated time series $\{q_t^{\setminus i}, \pi_t^{\setminus i}, \Delta c_t^{\setminus i}, \epsilon_t^{\setminus i}\}$, where $\epsilon_t^{\setminus i}$ is a $T \times 3$ matrix of shock realizations. Since shock i is turned off in the respective submodel, the corresponding column of $\epsilon_t^{\setminus i}$ is zero throughout the simulation.

After simulating all three submodels, we concatenate the resulting datasets into a single, unified dataset.[19] This combined dataset includes observations on π_t, Δc_t, and ϵ_t from each of the submodels, along with the associated bond prices q_t.

3.3 Neural Network Training

We then use the combined dataset from the simulations to train a neural network as a surrogate model. By including shock realizations equal to zero in the training data, we allow the neural network to learn from data in which a particular shock is absent with

[19] When combining the datasets, we shuffle the data such as to reduce autocorrelation of the input data, as well as to mix the input data from different submodels. Moreover, we cut the total number of input data to 20000, such that only a third of the available data is used for training and evaluation.

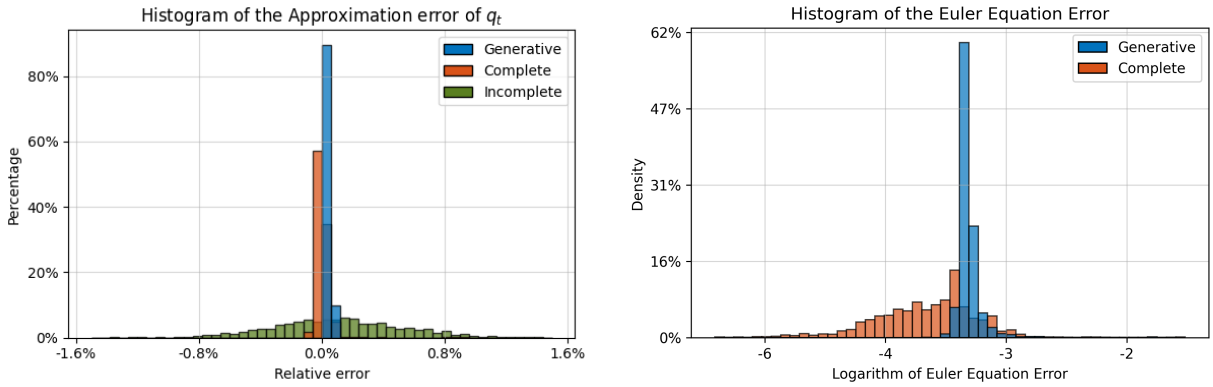

Figure 4 Approximation error of the asset price q_t (LHS) and Euler equation error (RHS) in the asset pricing model. For the approximation error, we compare the simulated value to the true value for 16000 periods for our generative modeling approach (blue). It is compared to using the complete model (orange) and incomplete model (green), for which we train a neural network using data generated from the full analytical solution and from a single submodel, respectively. The Euler equation error uses the predictions of generative economic modeling and calculates how well the model fulfills the model-implied Euler equation. We compare the Euler equation error for our generative method (blue) to the Euler equation error generated by the complete model (orange).

certainty. The neural network, as a flexible nonlinear function approximator, is thus trained to recognize that the policy function changes systematically in the absence of shocks.

The neural network architecture features five hidden layers with 128 neurons each, linear activation functions in the input and output layers, and HardSigmoid activations in all hidden layers. We train the network using the AdamW optimizer to minimize the mean squared error between predicted and true values. The learning rate follows a cosine annealing schedule, starting at 10^{-3} and decaying to 10^{-7} over the course of training.

The neural network is trained for 2000 epochs with a batch size of 200. For the training of the neural network, we divide our dataset into a training sample, which we use to train the neural network directly. We compare the loss with a validation sample to avoid overfitting the neural network. Figure 3 shows the mean squared residual error for the training and validation samples, which shows that the loss converges to 10^{-7} after 2000 epochs. While the validation loss is slightly larger, there is no overfitting, as the validation loss is not increasing.

3.4 Results

We now evaluate our generative approach using the approximation error, the Euler equation error, and model moments.

Approximation Error The numerical accuracy of our method is shown in Figure 4. Specifically, the left panel displays the histogram of the approximation error for the asset price. We compute the error as the difference between the neural network's predicted value and the true analytical value, using a sample of 16000 periods. The average error is close to zero, and the distribution is tightly concentrated around zero, showing a very good fit in this laboratory setting. The neural network extrapolates based on its fit to the training data from the different submodels. In doing so, it draws on the pricing coefficients it has learned separately from each submodel. Since each submodel isolates different combinations of shocks, the neural network learns all three aggregate risk pricing terms through exposure to their respective datasets. As a result, its extrapolations to the full model setting yield highly accurate predictions.

We also compare our approach to using the complete model, for which we train a neural network using data generated from the full analytical solution with all three aggregate shocks present. The error is very similar to the generative approach, confirming our method. By contrast, restricting training to a single submodel produces much higher errors.[20] The distribution for the incomplete model is more spread, as the model misses key features.

Euler Equation Error The right panel of Figure 4 reports our external validation measure — the Euler equation error. As mentioned earlier, the advantage of this measure is that it can be computed even without directly solving the complete model. The generative approach produces a very low Euler equation error, centered below 10^{-3}. For comparison, we also report the Euler equation error for the complete model, which is slightly smaller. However, in both cases the values are small and the difference is negligible. Thus, the Euler equation provides a direct validation of our approach.

We do not report the Euler equation error for the incomplete model, as this is only a measure of numerical accuracy and not a criterion for model selection. While a low Euler equation error indicates that the model is solved with high precision, it does not provide guidance on the appropriateness of the model's design. For this purpose, the model's intended use or its fit with the data should be the decisive consideration.

Moments Another important criterion is how well our approach recovers the underlying model moments. A particularly useful criterion here is the standard deviation of the asset

[20] We train another neural network that now only uses the simulations from a submodel that features only a TFP shock. Note that we include in the autoregressive process of equation (14), which governs inflation and consumption growth, all three shocks for the approximation and Euler equation. When using only a submodel also for equation (14), the errors would be substantially higher, resulting in a different order of magnitude for the error.

price, where our generative approach predicts a value of 0.1416, which is very close to the true value of 0.1423. In contrast to this, a submodel with only one shock, which is used to simulate the model and calculate the asset price, underestimates the standard deviation substantially. For instance, the standard deviation with only the TFP shocks would be only 0.0850. The reason is that two sources of risk are omitted. If we instead recalibrate the model to match a standard deviation of 0.1423 for the asset price, the contribution of the two shocks becomes highly overstated. Thus, it is critical to work with the model that best fits the purpose or the data, reiterating the need to solve the complete model.

4 Evaluation with an RBC model with Nonlinearities and Heterogeneous Agents

In this section, we evaluate our generative economic modeling approach using variants of an RBC model that we solve globally. We use different variants of the RBC model to assess how well our method handles nonlinear model solutions, with a particular focus on i) strong state dependencies and ii) the presence of heterogeneous agents. Crucially, the complete nonlinear models can still be solved using traditional solution techniques, allowing us to benchmark our approach against an existing global model solution technique.

4.1 Model Environment: Variants of the Real Business Cycle Model

We choose three different variants of the quantitative real business cycle (RBC) describd below to evaluate our method: i) a simplified but nonlinear version that can be solved analytically, ii) a medium-sized version with state-dependent investment costs that result in distinct nonlinearities, and iii) a heterogeneous agents version with partially uninsurable income risk in line with Krusell and Smith (1998). We work with the global solution for all variants.

RBC Model We use an extended stochastic RBC model composed of a firm sector, a household sector, and a government sector to test the predictive power of our method. The firm sector comprises of (i) final goods producers who bundle the intermediate goods to produce the final good Y_t, (ii) intermediate goods producers who hire labor services N_t for the wage w_t, rent capital K_t at price r_t from perfectly competitive markets, adjust their capital utilization u_t, but face monopolistic competition in the goods market as they produce differentiated goods, and (iii) producers of capital goods who invest I_t subject to adjustment costs to produce capital that they sell at price q_t.

Households earn income from supplying labor N_t and capital K_t, and earn profits Π_t from owning the firm sector. Households spend their income for consumption C_t and capital K_{t+1}. Finally, the government levies distortionary labor- and capital-income taxes with tax rates τ^L and τ^K, besides a value-added tax on consumption τ^C. Raising taxes is purely distortionary since the government returns the tax revenues to the household via lump-sum transfers T_t. Importantly, the model features up to five shocks. The markup μ_t, TFP A_t, labor-augmenting productivity Z_t, discount factor shock ζ_t, and depreciation rate δ_{0t} generate stochastic fluctuations.

We show the system of equations below, while the full derivation of the model equations is delegated to Appendix B. The production sector is described by:

$$Y_t = A_t \left(u_t K_t\right)^{\alpha} \left(Z_t N_t\right)^{1-\alpha}, \tag{19}$$

$$r_t + q_t \delta(u_t) = \frac{\alpha}{\mu_t} \frac{Y_t}{K_t}, \tag{20}$$

$$w_t = \frac{1-\alpha}{\mu_t} \frac{Y_t}{N_t}, \tag{21}$$

$$q_t \left[\delta_1 + \delta_2 (u_t - 1)\right] = \frac{\alpha}{\mu_t} A_t K_t \left(\frac{u_t K_t}{Z_t N_t}\right)^{\alpha-1} = \frac{\alpha}{\mu_t} \frac{Y_t}{u_t}, \tag{22}$$

$$q_t = \left[1 - \phi_t \left(\frac{I_t}{K_t} - \delta_{0t}\right)^{\kappa-1}\right]^{-1}, \tag{23}$$

$$K_{t+1} = \left(1 - \delta(u_t)\right) K_t + I_t - \frac{\phi_t}{\kappa} \left(\frac{I_t}{K_t} - \delta\right)^{\kappa} K_t. \tag{24}$$

where $\phi_t = \phi$. The equations characterizing the representative household are:

$$(1+\tau^C) C_t + q_t k_{t+1} = \left(q_t + (1-\tau^K) r_t\right) k_t + (1-\tau^L) w_t N_t + T_t + \Pi_t, \tag{25}$$

$$q_t u_C(C_t, N_t) = \beta \mathbb{E}_t \left[\frac{\zeta_{t+1}}{\zeta_t} \left(q_{t+1} + (1-\tau^K) r_{t+1}\right) u_C(C_{t+1}, N_{t+1})\right] \tag{26}$$

$$-u_N(C_t, N_t) = (1-\tau^L) w_t h_t \frac{u_C(C_t, N_t)}{1+\tau^C}, \tag{27}$$

where u_C and u_N denote the deviation of household utility function with respect to C_t and N_t. The government sector and the market clearing condition are given as:

$$T_t = \tau^C C_t + \tau^K r_t K_t + \tau^L w_t N_t, \tag{28}$$

$$Y_t = C_t + I_t \tag{29}$$

The different shock processes[21] are given as:

$$\ln A_t = (1-\rho_A)\left(\ln A - \frac{\sigma_A^2}{2}\right) + \rho_A \ln A_{t-1} + \epsilon_t^A, \quad \text{with} \quad \epsilon_t^A \sim N(0,\sigma_A^2), \tag{30}$$

$$\ln Z_t = (1-\rho_Z)\left(\ln Z - \frac{\sigma_Z^2}{2}\right) + \rho_Z \ln Z_{t-1} + \epsilon_t^Z, \quad \text{with} \quad \epsilon_t^Z \sim N(0,\sigma_Z^2), \tag{31}$$

$$\ln \mu_t = \left(\ln \mu + \frac{\sigma_\mu^2}{2}\right) + \epsilon_t^\mu, \quad \text{with} \quad \epsilon_t^\mu \sim N(0,\sigma_\mu^2), \tag{32}$$

$$\delta_{0t} = \delta_0 + \epsilon_t^\delta, \quad \text{with} \quad \epsilon_t^\delta \sim N(0,\sigma_\delta^2), \tag{33}$$

$$\ln \zeta_t = -(1-\rho_\zeta)\frac{\sigma_\zeta^2}{2} + \rho_\zeta \ln \zeta_{t-1} + \epsilon_t^\zeta, \quad \text{with} \quad \epsilon_t^\zeta \sim N(0,\sigma_\zeta^2). \tag{34}$$

We vary the activated shocks in the different model variants. Next, we illustrate the model variants and their specific features.

Variant 1: Analytical solution To begin with, we illustrate our methodology using a representative agent version of the model, which admits an analytical solution following Brock and Mirman (1972). The following proposition summarizes the assumptions necessary to obtain an analytical solution to the model.

Proposition 1. *If depreciation is deterministic and full, $\delta(u_t) = 1$, capacity utilization is fixed at $u_t = 1$, there are no capital adjustment costs $\phi = 0$, the discount factor shock is inactive $\sigma_\zeta^2 = 0$, and per period felicity is of King, Plosser and Rebelo (1988) (KPR)-form given by $u(C_t, N_t) = \ln C_t - \omega \frac{N_t^{1+\gamma}}{1+\gamma}$. Then the policy functions of the representative household are*

$$N_t = \left[\frac{\mu(1-\tau^L)(1-\alpha)}{\mu_t \omega (1+\tau^C)(\mu - \alpha\beta)}\right]^{\frac{1}{1+\gamma}}, \tag{35}$$

$$C_t = \left(1 - \frac{\alpha\beta}{\mu}\right) Y_t, \tag{36}$$

$$K_{t+1} = \frac{\alpha\beta}{\mu} Y_t, \tag{37}$$

with $Y_t = A_t K_t^\alpha (Z_t N_t)^{1-\alpha}$. Given the policy functions of the household, we can determine the prices in the economy. The transfers to the households are then determined by the government budget constraint.

Proof. See Appendix B.1. □

[21] Note the different formulation of the markup shock. The formulation with a positive adjustment for the variance ensures that $\mathbb{E}_t[\mu_t^{-1}] = \mu^{-1}$.

Proposition 1 presents the solution for the representative agent economy. Fluctuations in the markup μ_t drive changes in labor supply N_t, while shocks to productivity (A_t, Z_t) affect output Y_t. Consumption C_t and capital investment K_{t+1} are linear functions of output. This version serves as a simple starting point to illustrate the fit of our method to standard RBC environments.

Variant 2: State-Dependent Adjustment Costs To highlight the importance of working with a global solution, we solve a model that includes an additional non-linear element. We assume that the adjustment costs are state-dependent instead of being constant:

$$\phi_t = \begin{cases} \overline{\phi} & \text{if } K_t > \overline{K} \\ \underline{\phi} & \text{if } K_t \leq \overline{K} \end{cases} \tag{38}$$

where $\overline{\phi} \geq \underline{\phi} \geq 0$.[22] The non-linear adjustment costs directly affect the price of capital and the law of motion of capital.

Variant 3: Heterogeneous Households Next, we introduce household heterogeneity in capital holdings k_{it}, in their profit income Π_{it}, and in their idiosyncratic income component h_{it}. Households optimize their utility function subject to their individual budget constraint

$$(1+\tau^C)c_{it} + q_t k_{it+1} = \left(q_t + (1-\tau^K)r_t\right) k_{it} + (1-\tau^L)w_t h_{it} n_{it} + T_t + \Pi_{it}. \tag{39}$$

Households face a borrowing constraint, such that they are prohibited from holding negative amounts of assets. Individual productivity h_{it} evolves according to

$$\log h_{it} = -(1-\rho_h)\frac{\sigma_h^2}{2} + \rho_h \log h_{it-1} + \epsilon_{it}^h \quad \text{with} \quad \epsilon_{it}^h \sim N(0, \sigma_h^2). \tag{40}$$

with ϵ_{it}^h as a normally distributed shock with variance σ_ϵ^2 and mean zero.

The solution of the household problem can be characterized by the Euler equation on capital

$$q_t u_C(c_{it}, n_{it}) = \beta \mathbb{E}_t \left[\frac{\zeta_{t+1}}{\zeta_t} \left(q_{t+1} + (1-\tau^K) r_{t+1}\right) u_C(c_{it+1}, n_{it+1}) \right], \tag{41}$$

$$\tag{42}$$

[22] Note that capital good producers take the adjustment costs as given, as they depend on aggregate capital.

Calibrations We calibrate the different model version to conventional values, which are summarized in Appendix B.

Calculation of the Euler Equation Error For all models, we compute the Euler Equation Error to assess the accuracy of the solution. We compute the relative Euler equation error along for a simulation of the economy going forward for T periods as follows:

$$\frac{1}{N}\sum_{t=1}^{T}\sum_{i=1}^{N}\left(\frac{(u')^{-1}(\beta\mathbb{E}_t[R_{t+1}u'(c_{it+1})])}{c_{it}}-1\right) \tag{43}$$

where $R_{t+1}=\frac{\zeta_{t+1}}{\zeta_t}\frac{q_{t+1}+(1-\tau^K_{t+1})r_{t+1}}{q_t}$, and u' and $(u')^{-1}$ denote marginal utility and the inverse of marginal utility, respectively. For the economies with heterogeneous agents, we average the Euler equation error across the number of grid points N, we use to discretize the household distribution. For the representative agent variants 1 and 2, we set $N=1$.

4.2 Generative Economic Modeling

We are now using our generative economic modeling approach to solve the different variants of the real business cycle model, illustrated above.

4.2.1 Generative Economic Modeling with an Analytical Solution

We decompose the initial model into three submodels with varying shock combinations: (i) TFP and labor-augmenting productivity shocks, (ii) TFP and markup shocks, (iii) labor-augmenting productivity and markup shocks. We use the analytical solution, adapted for the varying shock combinations, to simulate each economy for 10000 periods. We combine the simulations in a single dataset, where we set the shocks to zero if they are not active in the submodel. The neural network is then trained for 1000 epochs with a batch size of 100 on the combined dataset. The neural network converges to a very training loss, with no evidence of overfitting. The loss for the training and validation samples is shown, together with a detailed description of the neural network architecture, in Appendix B.

Results Figure 5 illustrates the performance of our methodology. The left panel presents the approximation error between the predicted value and the analytical values of capital, K_t. The errors are very small and centered around zero. When compared to the complete model, for which a neural network is trained on data generated from the analytical solution, the magnitude of the error is very similar. In contrast, the approximation error increases substantially when using an incomplete model. We use here the submodel that

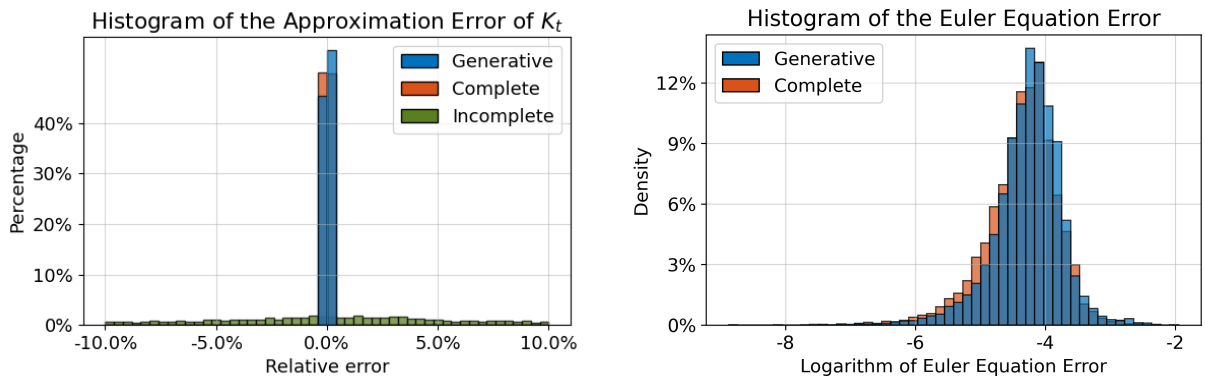

Figure 5 Approximation error of the capital stock K_t (LHS) and Euler equation error (RHS) in the analytical RBC model. For the approximation error, we compare the simulated value to the true value for 10000 periods for our generative modeling approach (blue). It is compared to using the complete model (orange) and incomplete model (green), for which we train a neural network using data generated from the full analytical solution and from a single submodel, respectively. The Euler equation error uses the predictions of generative economic modeling and calculates how well the model fulfills the model-implied Euler equation. We compare the Euler equation error for our generative method (blue) to the Euler equation error generated by the complete model (orange).

features only Markup shocks. Figure C.1 (in the Appendix) illustrates the prediction errors for consumption and investment, which have similar patterns to those observed for capital.

The right panel of Figure 5 shows the Euler equation error. We obtain average Euler equation errors of 0.00009 and 0.00011 for the complete model and using generative economic modeling, respectively. Consequently, we demonstrate that our method achieves a level of precision comparable to the complete model trained on the true data-generating process, mirroring the findings for the analytical asset pricing model. The last important criterion is how well our approach approximates the moments of the true data generating process. For instance, the standard deviation of capital is 0.0413 in both the generative and complete model, while an incomplete model version would result in a substantially lower value. For instance, a version without the TFP shock generates only about half the standard deviation for capital.

4.2.2 Generative Economic Modeling with a Nonlinear Model

In this section, we illustrate our methodology using a medium-sized version of our RBC model. The model features a strong non-linearity in the adjstment costs. We abstract from shocks to labor-augmenting productivity Z_t and to the markup μ_t, hence shocks to TFP A_t are the only drivers of supply-side fluctuations, but we complement them through shocks to the discount factor ζ_{t+1} and the depreciation rate δ_{0t}. We assume that households remain ex-ante and ex-post identical, such that we can represent the household side with a repre-

sentative agent. Moreover, households have a common utility function that is separable in consumption and labor:

$$u(C_t, N_t) = \frac{C_t^{1-\sigma} - 1}{1-\sigma} - \omega \frac{N_t^{1+\gamma}}{1+\gamma} \tag{44}$$

The optimal household behavior can be described by the Euler equation and the optimal labor supply condition:

$$\begin{aligned} C_t^{-\sigma} &= \beta \mathbb{E}_t \left[\frac{\zeta_{t+1}}{\zeta_t} (q_{t+1} + (1-\tau^K) r_{t+1}) C_{t+1}^{-\sigma} \right], \\ \omega N_t^{\gamma} &= \frac{1-\tau_t^L}{1+\tau^C} w_t C_t^{-\sigma}. \end{aligned}$$

Contrary to the former section, we allow for capacity utilization choice and allow for capital adjustment costs with a nonlinear specification as illustrated in equations (24) and (38).

Generative Economic Modeling We decompose the underlying complete model into three submodels with varying shock combinations: (i) TFP and discount factor shocks, (ii) TFP and depreciation shocks, and (iii) discount factor and depreciation shocks. The submodels are solved with global methods, specifically policy function iteration, to account for all nonlinear features.[23] Using our global solution, we simulate time series data for three submodels for 10000 periods. We combine the simulations in a single dataset, where we set the shocks to zero if they are not active in the submodel. Specifying the neural network similarly to before, it is then trained for 1000 epochs with a batch size of 100. The loss for the training and validation samples converges to low values. The details on the training and architecture are in Appendix B.

Results Figure 6 reports the performance of our algorithm for the medium-sized nonlinear DSGE model, using capital as an example. Comparing the generative approach's predictions with the complete model demonstrates that our methodology provides a strong fit, even for a medium-sized DSGE model with nonlinearities. The predicted values closely align with the true values, addressing the issue of under- and overprediction observed in the analytical version from the previous section. In contrast, the approximation error increases when using an incomplete model. We use here the submodel that features only

[23] Within the class of policy function iteration methods, we use time iteration with linear interpolation as in Richter, Throckmorton and Walker (2014) and Bianchi, Melosi and Rottner (2021). The parameter choices are conventional and chosen to ensure strong nonlinearities in the shock propagation, as shown in Appendix B.2.

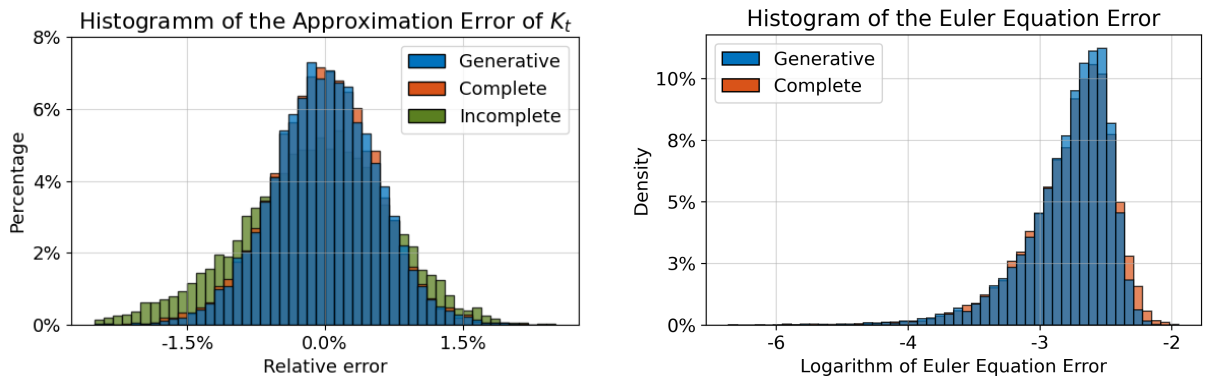

Figure 6 Approximation error of the capital stock K_t (LHS) and Euler equation error (RHS) in the nonlinear RBC model. For the approximation error, we compare the simulated value to the true value for 10000 periods for our generative modeling approach (blue). It is compared to using the complete model (orange) and incomplete model (green), for which we train a neural network using data generated from the full analytical solution and from a single submodel, respectively. The Euler equation error uses the predictions of generative economic modeling and calculates how well the model fulfills the model-implied Euler equation. We compare the Euler equation error for our generative method (blue) to the Euler equation error generated by the complete model (orange).

depreciation shocks. The error is particularly larger on the left-hand side of the distribution, in line with our state-dependent adjustment cost function for investment. Figure C.3 (in the Appendix) shows the prediction errors for other key variables.

The right panel of Figure 5 shows the Euler equation error. On average, we obtain Euler equation errors of 0.0032 and 0.0035 for the complete model and using generative economic modeling, respectively. The histograms illustrate that our method achieves a level of precision comparable to the complete model trained on the true data-generating process. As before, we observe substantial differences in the moments if we were to use only a submodel. Overall, our generative approach effectively captures the dynamics of the underlying model, even in the case of pronounced nonlinearities.

4.2.3 Generative Economic Modeling with Heterogeneous Agents

This section applies our methodology to a model with heterogeneous households. In contrast to earlier sections, we now incorporate both ex-ante and ex-post heterogeneity, following the framework of Krusell and Smith (1998). As a result, the joint distribution of wealth and income becomes a state variable when solving the model.

To simplify the solution of the model, we abstract from endogenous capacity utilization ($u_t = 1$) and set capital adjustment costs to zero ($\phi_t = \kappa = 0$). Households maximize a standard CRRA utility function:

$$u(c_{it}) = \frac{c_{it}^{1-\sigma} - 1}{1 - \sigma}$$

subject to the budget constraint in equation (39). Since labor supply entails no disutility, households supply one unit of labor inelastically. We shut down government activity and keep three aggregate shocks: productivity (A_t), discount factor (ζ_t), and depreciation (δ_t).

Generative Economic Modeling We decompose the underlying complete model into three submodels with varying shock combinations: (i) TFP and discount factor shocks, (ii) TFP and depreciation shocks, and (iii) discount factor and depreciation shocks. Finally we also solve the model version with all three shocks active. A detailed description of the solution algorithm for the heterogeneous agent model with multiple aggregate shocks is provided in Appendix B.3. Here, we briefly summarize the approach. We solve the household problem using the endogenous grid point method of Carroll (2006), and simulate the economy using the non-stochastic simulation method from Young (2010). Since the model includes aggregate shocks, households require a perceived aggregate law of motion (ALM) for aggregate capital. Following Krusell and Smith (1998), we assume a state-dependent log-linear ALM and update it iteratively until convergence, while ensuring that the gap between the true and perceived laws is minimal, in line with the concerns raised by Den Haan (2010). We extend the original method to allow for a nonlinear law of motion with multiple aggregate shocks. We ensure that the gap between the true and perceived laws is minimal. Figure B.4 (in the appendix) shows that the model-generated capital series closely match the ALM, with a maximum deviation of less than 0.01. We solve the model using four idiosyncratic income states and four aggregate states per shock, which we approximate as four-state Markov chains using the method of Tauchen (1986). Unlike Krusell and Smith (1997) and Krusell and Smith (1998), we abstract from any dependence of idiosyncratic income risk on aggregate TFP.[24] We use an exponentially space asset grid with 100 grid points and 20 grid points for aggregate capital. The full model therefore spans $4^4 \times 100 \times 20 = 512{,}000$ grid points.

Using our global solution, we simulate time series data for three submodels over 10000 periods. We combine the simulations into a single dataset, where we set the shocks to zero if they are not active in the submodel. Specifying the neural network similarly to before, it is then trained for 1000 epochs with a batch size of 100. The loss for the training and

[24] Although we can allow for a correlation between idiosyncratic and aggregate risk, with more than two aggregate states there exist various possibilities to allow for cyclical variation in idiosyncratic income risk. We leave the exploration of the impact of cyclical idiosyncratic income risk on the performance of our method for future research.

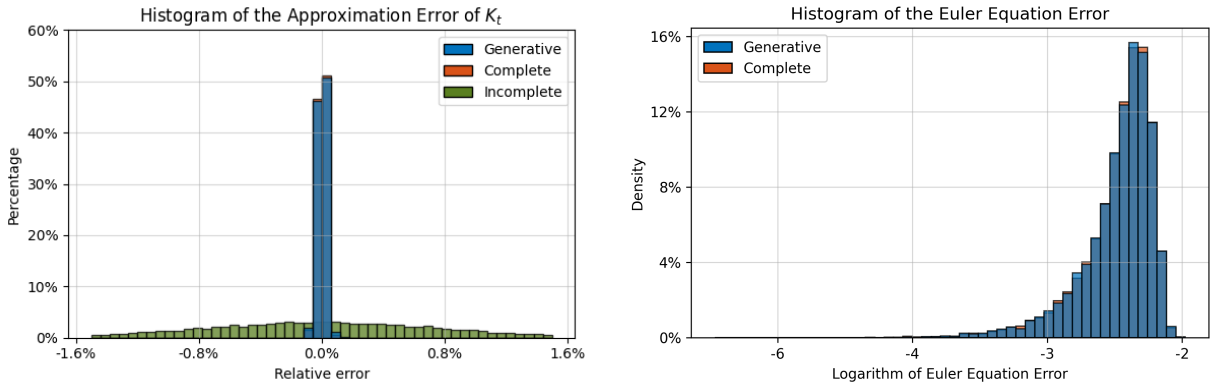

Figure 7 Approximation error of the capital stock K_t (LHS) and Euler equation error (RHS) in the heterogeneous agent RBC model. For the approximation error, we compare the simulated value to the true value for 10000 periods for our generative modeling approach (blue). It is compared to using the complete model (orange) and incomplete model (green), for which we train a neural network using data generated from the full analytical solution and from a single submodel, respectively. The Euler equation error uses the predictions of generative economic modeling and calculates how well the model fulfills the model-implied Euler equation. We compare the Euler equation error for our generative method (blue) to the Euler equation error generated by the complete model (orange).

validation samples converges to low values. The details on the training and architecture are in Appendix B.

Results Figure 7 evaluates the performance of our algorithm for the heterogeneous agent model of capital, where the approximation error of capital, K_t, and the Euler equation error are shown. In the left panel, we compare the predictions of the generative approach with those of the complete model. The comparison demonstrates that our methodology provides a strong fit, even in the presence of household heterogeneity. In contrast, the approximation error increases when using an incomplete model. We use here the submodel that features only TFP- and discount-factor-shocks. Figure C.4 (in the Appendix) shows the prediction errors for other aggregate variables.

The right panel of Figure 7 shows the Euler equation error. The histograms illustrate that our method achieves basically the same level of precision as the complete model trained on the true data-generating process. On average, we achieve Euler equation errors of 0.00806 and 0.00807 for the complete model and using generative economic modeling, respectively. Finally, we can compare moments simulated from the solution obtained from the policy functions. As before, our methodology replicates moments close to the true data generation process, while we observe substantial differences in the moments if we only use a submodel. Overall, our generative approach effectively captures the dynamics of the underlying model.

To calculate the Euler Equation Error of the model, we also applied our methodology

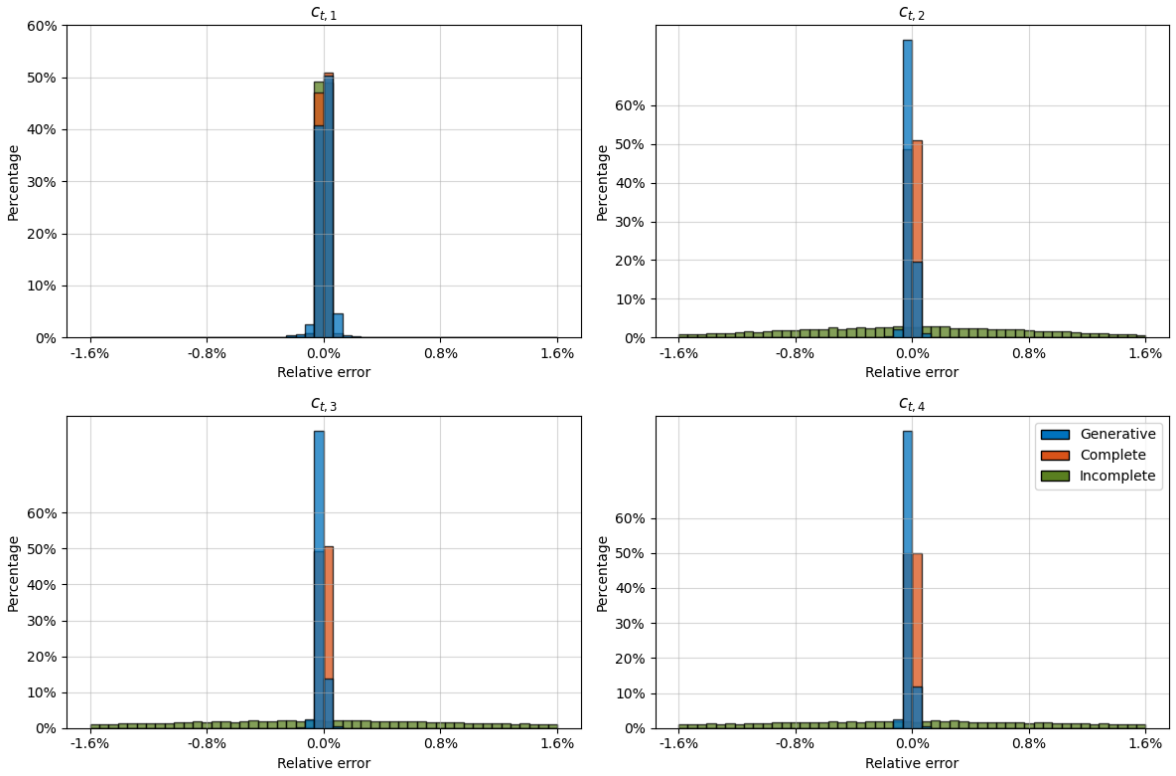

Figure 8 Approximation error of the policy functions of the heterogeneous agent economy. The plots indicate the prediction error for the consumption policy function at the borrowing constraint over four income states in descending order. We compare here the simulated value to the true value for 10000 periods for our generative modeling approach (blue). It is compared to using the complete model (orange) and incomplete model (green), for which we train a neural network using data generated from the full analytical solution and from a single submodel, respectively.

to predict the policy functions of households at each individual point of the discretized individual state space. Figure 8 illustrates the fit of the method when predicting the policy functions for households at the borrowing constraint. As the figure shows, the fit is excellent on the household level, as well. Additional results, corroborating the previous findings, are available in Appendix C. Finally, our method captures the moments of the model again very well, satisfying another key objective. This result speaks to the heterogeneous agent literature, as it substantially facilitates the global solution of heterogeneous agent models.

5 HANK with an Occasionally Binding Financial Friction

In this section, we use our method of generative economic modeling to globally solve a medium-scale HANK model with an occasionally binding financial friction. The financial frictions occasionally prevent firms from hiring as many workers as they desire and introduce a nonlinearity that requires a global solution to be fully taken into account. How-

ever, solving such a high-dimensional model directly is computationally very demanding, as the dimensionality of the state space increases with each additional state variable and aggregate shock. To address this challenge, we rely on our generative economic modeling approach. Importantly, the model's complexity precludes us from directly solving the complete version. Instead, we evaluate the accuracy of our approach using our established evaluation criteria - the Euler equation error.

5.1 Description of the Model

The model is a HANK model that includes both idiosyncratic and aggregate risk. Households insure against both types of risks by saving in liquid assets subject to a borrowing limit. Intermediate goods are produced using labor under monopolistic competition, where firms face Rotemberg price adjustment costs. The firms also face an occasionally binding cash-in-advance constraint limiting production in times of financial distress. A final goods bundler bundles intermediate goods into a final good. The government raises taxes to issue bonds and for government consumption, while the central bank sets the nominal interest rate as a function of price inflation. The model features shocks to the discount factor, shocks to aggregate productivity, monetary policy shocks, as well as shocks to the ability of firms to borrow through the financial sector.

Households There exists a continuum $i \in [0, 1]$ of households which choose to obtain utility from consumption c_{it} and leisure, and save in liquid assets b_{it+1} such as to insure against idiosyncratic income fluctuations in labor productivity h_{it}. Labor productivity follows an AR(1) process in logs as in equation (40). Households maximize the following utility:

$$\mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t \zeta_t u(c_{it}, n_{it}), \tag{45}$$

where n_{it} denotes their labor supply and ζ_t denotes the shock to the discount factor. We assume that the discount factor shock follows an AR(1) process as in equation (34). We assume that household felicity is of Greenwood, Hercowitz and Huffman (1988) (GHH) form together with a CRRA specification:

$$u(c_{it}, n_{it}) = \frac{\left(c_{it} - \omega h_{it} \frac{n_{it}^{1+\gamma}}{1+\gamma}\right)^{1-\sigma} - 1}{1-\sigma}, \tag{46}$$

where ω is a scalar for multiplying the disutility of supplying labor. Households maximize utility subject to the budget and borrowing constraint

$$c_{it} + b_{it+1} = (1 - \tau_t) w_t n_{it} h_{it} + R_t b_{it} + (1 - \tau_t) d_{it}, \tag{47}$$

$$b_{it+1} \geq \bar{B} \tag{48}$$

where b_{it+1} denotes savings of the household, τ_t the income tax, $R_t = 1 + r_t$ the real interest rate, and profits d_{it} from owning the firm sector. We distribute profits proportional to the idiosyncratic productivity h_{it}. $\bar{B}$ denotes the exogenous borrowing limit of households.

Firms A final goods producer bundles a continuum of differentiated varieties $j \in [0, 1]$ into a final good according to a Dixit-Stiglitz aggregator

$$Y_t = \left(\int_0^1 y_{jt}^{\frac{\eta-1}{\eta}} dj \right)^{\frac{\eta}{\eta-1}}, \tag{49}$$

with elasticity of substitution η. This yields an optimal demand for each variety j of

$$y_{jt} = \left(\frac{p_{jt}}{P_t} \right)^{-\eta} Y_t, \tag{50}$$

where $P_t = (\int_0^1 p_{jt}^{1-\eta} dj)^{\frac{1}{1-\eta}}$ denotes the price level. Each differentiated variety is produced by an intermediate goods producer with index j using labor as input. Production follows the linear production function

$$Y_{jt} = A_t N_{jt}, \tag{51}$$

where A_t denotes aggregate productivity that follows an AR(1) process in logs. Intermediate goods producers are subject to quadratic price adjustment costs in logarithmic price changes. Hence, for price-setting, the firm maximizes

$$\mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t Y_t \left\{ \left(\frac{p_{jt}}{P_t} - MC_t \right) \left(\frac{p_{jt}}{P_t} \right)^{-\eta} - \frac{\eta}{2\kappa} \left(\log \frac{p_{jt}}{p_{jt-1}} \right)^2 \right\}, \tag{52}$$

with a time-constant discount factor.The producer's first-order condition gives rise to a New Keynesian Phillips curve in goods price inflation

$$\log(\pi_t) = \beta \mathbb{E}_t \left[\log(\pi_{t+1}) \frac{Y_{t+1}}{Y_t} \right] + \kappa \left(MC_t - \frac{\eta - 1}{1} \right), \tag{53}$$

where Π_t is the gross inflation rate $\Pi_t \equiv \frac{P_t}{P_{t-1}}$, and MC_t is the real marginal costs. The price adjustment then creates real costs $\frac{\eta}{2\kappa} Y_t \log(\Pi_t)^2$.

Finally, intermediate goods producers are subject to a financing constraint when paying their labor bill. Firms need to borrow their wage bill from a perfectly competitive financial intermediary at a zero intratemporal interest rate; however, due to agency costs are not able to do so up to the full level of their revenue. Hence firms face the following borrowing constraint:

$$w_t N_{jt} \leq \lambda_t y_{jt}, \tag{54}$$

where λ_t denotes the fraction of output that firms are allowed to borrow. We assume λ_t to follow an AR(1) process in logs:

$$\ln \lambda_t = -(1-\rho_\lambda)\frac{\sigma_\lambda^2}{2} + \rho_\lambda \ln \lambda_{t-1} + \epsilon_t^\lambda \quad \text{with} \quad \epsilon_t^\lambda \sim N(0, \sigma_\lambda^2). \tag{55}$$

This implies that if $\lambda_t < MC_t$, the household is constrained in its labor bill and firms can only demand labor up to the wage rate $w_t = \lambda_t A_t$. Hence, if firms are financially constraint, they cannot produce up to their capacity, because they are limited in the wages they can pay. This introduces a nonlinearity in the economy, which makes the solution of the model numerically more demanding.

Government The government operates a monetary and a fiscal authority. The monetary authority controls the nominal interest rate on liquid assets, while the fiscal authority issues government bonds to finance deficits and adjusts expenditures to stabilize debt in the long run and output in the short run.

We assume that monetary policy sets the nominal interest rate i_t on bonds following a Taylor-type rule:

$$(1 + i_{t+1}) = \Pi_t^{\phi_\pi} \exp(\iota_t), \tag{56}$$

where ϕ_π governs the extent to which the central bank attempts to stabilize inflation. ι_t is an exogenous monetary policy shock that follows an AR(1) process in logs:

$$\ln \iota_t = -(1-\rho_\iota)\frac{\sigma_\iota^2}{2} + \rho_\iota \ln \iota_{t-1} + \epsilon_t^\iota \quad \text{with} \quad \epsilon_t^\iota \sim N(0, \sigma_\iota^2) \tag{57}$$

The real interest rate is then determined using a Fisher relation $R_t = 1 + r_t = \frac{1+i_t}{\Pi_t}$

Moreover, we assume that the government issues bonds according to the rule

$$\frac{B_{t+1}}{\bar{B}} = \left(\frac{R_t B_t}{\bar{R}\bar{B}}\right)^{\rho_B} \left(\frac{\Pi_t}{\bar{\Pi}}\right)^{-\gamma_\pi} \left(\frac{\mathcal{T}_t}{\bar{\mathcal{T}}}\right)^{-\gamma_\mathcal{T}}, \tag{58}$$

using tax revenues, $\mathcal{T}_t = \tau Y_t$, to finance government consumption, G_t, and interest on outstanding debt. The coefficients $\bar{B}$, $\bar{\Pi}$, and $\bar{\mathcal{T}}$ are normalization constants. ρ_B captures whether and how fast the government seeks to repay its outstanding obligations, $B_t R_t$. For $\rho_B < 1$, the government actively stabilizes real government debt, and for $\rho_B = 1$, the government rolls over all outstanding debt, including interest. The coefficients γ_π band $\gamma_\mathcal{T}$ capture the cyclicality of debt issuance: for $\gamma_\pi = \gamma_\mathcal{T} = 0$, new debt does not actively react to tax revenues and inflation, but only to the value of outstanding debt; for $\gamma_\pi > 0 > \gamma_\mathcal{T}$, debt is countercyclical; for $\gamma_\pi < 0 < \gamma_\mathcal{T}$, debt is procyclical. We assume that government expenditure G_t adjusts such that the government budget constraint is satisfied $G_t + R_t B_t = B_{t+1} + \mathcal{T}_t$

Market Clearing Market clearing requires that the labor market, the bond market, as well as the goods market, clear. GHH preferences imply that households supply labor according to $n_{it} = (\frac{(1-\tau)w_t}{\omega})^{\frac{1}{\gamma}} = N_t$ where the last equality follows from $\int_0^1 h_{it} di = 1$. Hence, labor market clearing is achieved if

$$N_t = \begin{cases} \left(\frac{(1-\tau)A_t MC_t}{\omega}\right)^{\frac{1}{\gamma}} & \text{if unconstrained} \\ \left(\frac{(1-\tau)A_t \lambda_t}{\omega}\right)^{\frac{1}{\gamma}} & \text{if constrained} \end{cases}. \tag{59}$$

Bonds market clearing is achieved if

$$B_{t+1} = \int_0^1 b_{it+1} di, \tag{60}$$

and goods market clearing is achieved if

$$(1 - \frac{\eta}{2\kappa} (\ln \Pi_t)^2) Y_t = C_t + G_t, \tag{61}$$

where the left-hand side indicates production adjusted for price-adjustment costs.

Computational Challenges The problem faces three computational challenges. First, to solve the model, we need to employ the algorithm of Krusell and Smith (1997) to forecast the forward-looking part $\mathbb{E}_t \left[\log(\pi_{t+1}) \frac{Y_{t+1}}{Y_t}\right]$ in the Philips curve, as well as today's inflation Π_t using a perceived law-of-motion. This requires solving and simulating the model multiple times to update the law of motion until convergence. This issue is identical to the computation issue illustrated in section 3.2.3. Second, we need to calculate the market-clearing inflation rate in the simulation step to update the prediction of the nowcast of

inflation, which is necessary for the solution of the household problem. Hence, after solving the household problem globally, we need to introduce a root-finding step. Concretely, we guess an inflation rate $\tilde{\Pi}_t$ that then determines the nominal interest rate i_{t+1}, relevant for the savings choice of the households. Given the aggregate states $(B_t, \lambda_t, A_t, \zeta_t, \iota_t)$, we impose labor market clearing by using equation (59) and then update the guess for inflation $\tilde{\Pi}_t$ until bond market clearing is achieved. This additional step requires us to solve the household problem (although only for one backward iteration) multiple times. This adds additional computational time. Third, adding the cash-in-advance constraint implies that the economy features nonlinear dynamics if intermediate good firms are financially constrained. This requires additional runtime to solve for accurate perceived laws of motion.

Calibration We calibrate the submodels to have values of Bayer et al. (2019) with exception to the parameters guiding the bond rule. As we solve the model globally, the rule parameters have important implications for the dynamics and the stability of the economy. We set the parameters such as to ensure nondivergent paths of government debt. All parameter values, as well as plots illustrating the goodness of fit of our solution can be found in Appendix D.

5.2 Generative Economic Modeling Solution

The global solution of this model with conventional methods remains numerically intractable. Consequently, we solve the model using our methodology of generative economic modeling. For that, we generate time series data for three satellite models, each subject to two out of the four possible shocks. We solve and simulate satellite models with (i) financial, and TFP shocks, (ii) financial and discount factor shocks, and (iii) financial and monetary shocks. We also solve a model version without financial shocks. For the approximation of the true solution, we solve three satellite models, each including one of the three non-financial shocks. Hence we solve and simulate models with (i) TFP shocks, (ii) discount factor shocks, and (iii) monetary shocks. We use this solution of the model to understand the effect of introducing financial shocks into a HANK model. We train two neural networks, each on the combined datasets of our satellite models. The neural networks consist of five hidden layers, each with 128 neurons, using the CELU activation function. The optimizer employed is AdamW, and the training minimizes the mean squared error between the predicted and true values. The learning rate follows a cosine annealing schedule, starting at 10^{-3} and decaying to 10^{-10}.

The left panel of Figure 9 reports the mean squared error during the training. The loss for the training sample is around 1e-5. The validation loss indicates that the neural network

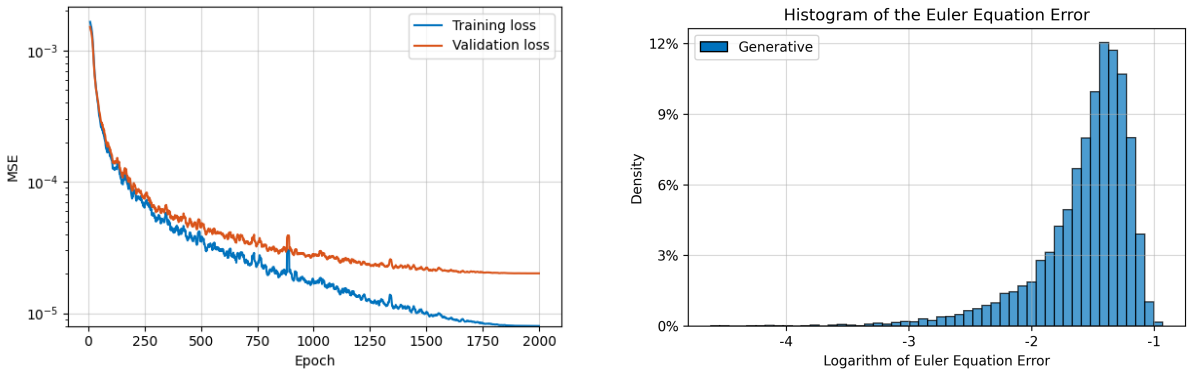

Figure 9 Training convergence and histogram of Euler equation error for the HANK model with an occasionally binding constraint. The left panel shows the loss over the training and validation samples. An epoch is completed when all the training or validation sample points are utilized. The vertical axis is expressed in logarithmic scale. The right panel shows the histogram of the Euler equation error, where we calculate how well the model fulfills the model-implied Euler equation.

does not overfit. The right panel shows the histogram of the Euler equation error. While the error is slightly larger than for the Krusell-Smith model, this is natural as the model is considerably more complex. At the same time, the shape of the histogram is also very similar to before and well behaved, without any large outliers. Taken together, we consider the Euler equation error in a very reasonable range and find this outcome very reassuring.

5.3 Results: Financial Friction Shock and Aggregate Risk

Having solved the HANK model with an occasionally binding financial friction, we now investigate the transmission of shocks and the implications of aggregate risk.

5.3.1 Impact of a Financial Friction Shock

First, we are interested in the transmission of financial shocks in our model. To analyze the impact of a financial friction shock on the economy, we investigate the generalized impulse responses of the model to financial shocks of varying sizes. Concretely, we illustrate response of the economy to a shock in λ_t of -5% and -7.5% relative to its mean. Figure (10) illustrates the generalized impulse response function of the solved model to these shocks.

A drop in λ_t, the variable determining the space of the financial sector for intratemporal lending induces a recession for both shock sizes. As result of the drop, firms are constraint and hire less labor N_t, which reduces production Y_t. As result of this negative supply shock, consumption C_t drops, while inflation Π_t increases. The central bank increases the nominal interest rate I_t going forward in response to the hike in inflation, thereby increasing the real

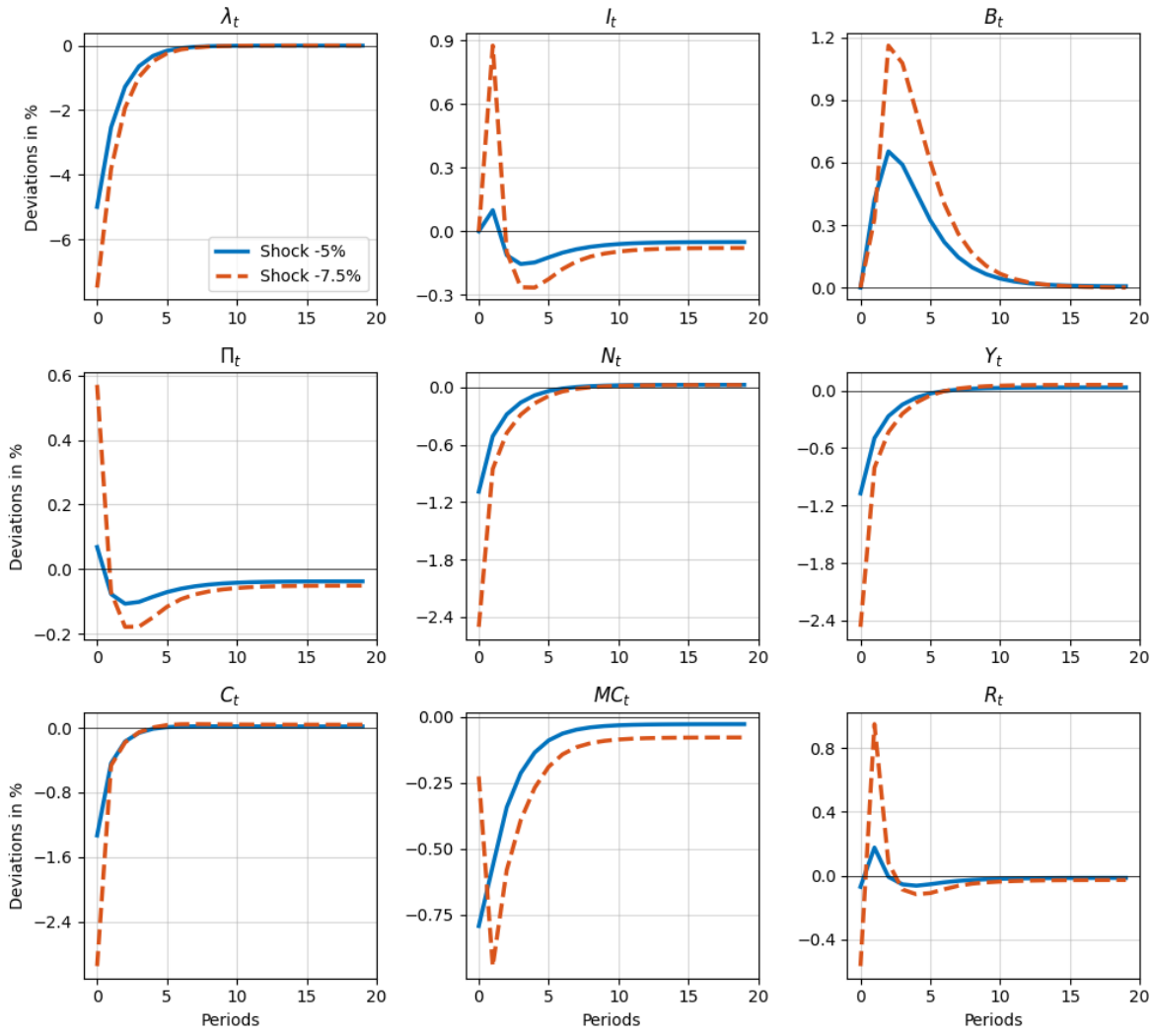

Figure 10 Generalized impulse response functions for a heterogeneous agent New Keynesian economy with a financial friction, solved with generative economic modeling. Each panel reports percentage deviations from the stochastic steady state. We compute the responses by initializing the economy at its stochastic steady state and applying a one-time innovation to the financial wedge λ_t, which follows an AR(1) process. The same policy functions are used in both experiments, once with a $5\,\%$ and once with a $7.5\,\%$ decline in λ_t. One period corresponds to one year in the baseline calibration.

interest rate R_t. Bond supply B_t increases by the government, triggering countercyclical government expenditure.

Besides having these qualitative responses in common, the economy features nonlinearity in response to different sizes of shocks. The 7.5% decrease in financial space λ_t triggers a substantially larger recession than the 5% decrease in lambda. This manifests itself in a larger reduction in labor N_t, a larger decrease in output Y_t, and a larger decrease in consumption C_t. As the shock to the supply side is more severe, inflation Π_t increases more, triggering a larger increase in the nominal interest rate I_t. The shock is so contractionary

that the marginal costs for production does even increases in response to the shock before dropping, indicating that in the first period the costs of production increase due to the financial friction.

To summarize, shocks to λ_t that serves as a financial shock here triggers nonlinear dynamics in response to different shock sizes. Such dynamics can be especially important when trying to understand financial crisis. Generative economic modeling serves as a useful tool to solve models with nonlinear dynamics or heterogeneity, while keeping it numerically tractable.

5.3.2 Implications of Increased Aggregate Risk

Second, we are interested in the interaction of different sources of risk with each other. For that, we compare the dynamics of the economy with four shocks with an economy that only features the financial shock and a TFP shock. Through this analysis, we aim to understand how the presence of more shocks in the model shapes the response of the model. Figure (11) illustrates impulse responses of the economy with four shocks to a TFP shock compared to the impulse response of the satellite economy with only the TFP shock and the lambda shock.

The comparison between the two impulse responses shows how incorporating more shocks in a model alters the dynamics of the model. For all variables illustrated, the responses of the variables after the financial shock is attenuated in the model with all four shocks compared to only featuring a TFP shock. Hence, for the model environment we solve here, introducing more shocks beyond the financial shock and the TFP shock reduces the response of endogenous variables to the financial shock. One economic explanation for this is that in the presence of more aggregate shocks, households have a larger precautionary incentive. Hence, knowing that they will face larger aggregate volatility with more shocks, they build up more insurance through precautionary savings. In response to one of these shocks realizes, households rely on this savings and react less to the aggregate shock.

In the context of our model here, integrating more shocks dampens the response of the economy to the nonlinear financial shock. The methodology of generative economic modeling allows users to integrate more shocks to study the implications that integrating more shocks has for their model.

6 Conclusion

Our study introduces generative economic modeling, a novel approach that combines conventional solution methods with artificial intelligence to overcome computational barriers

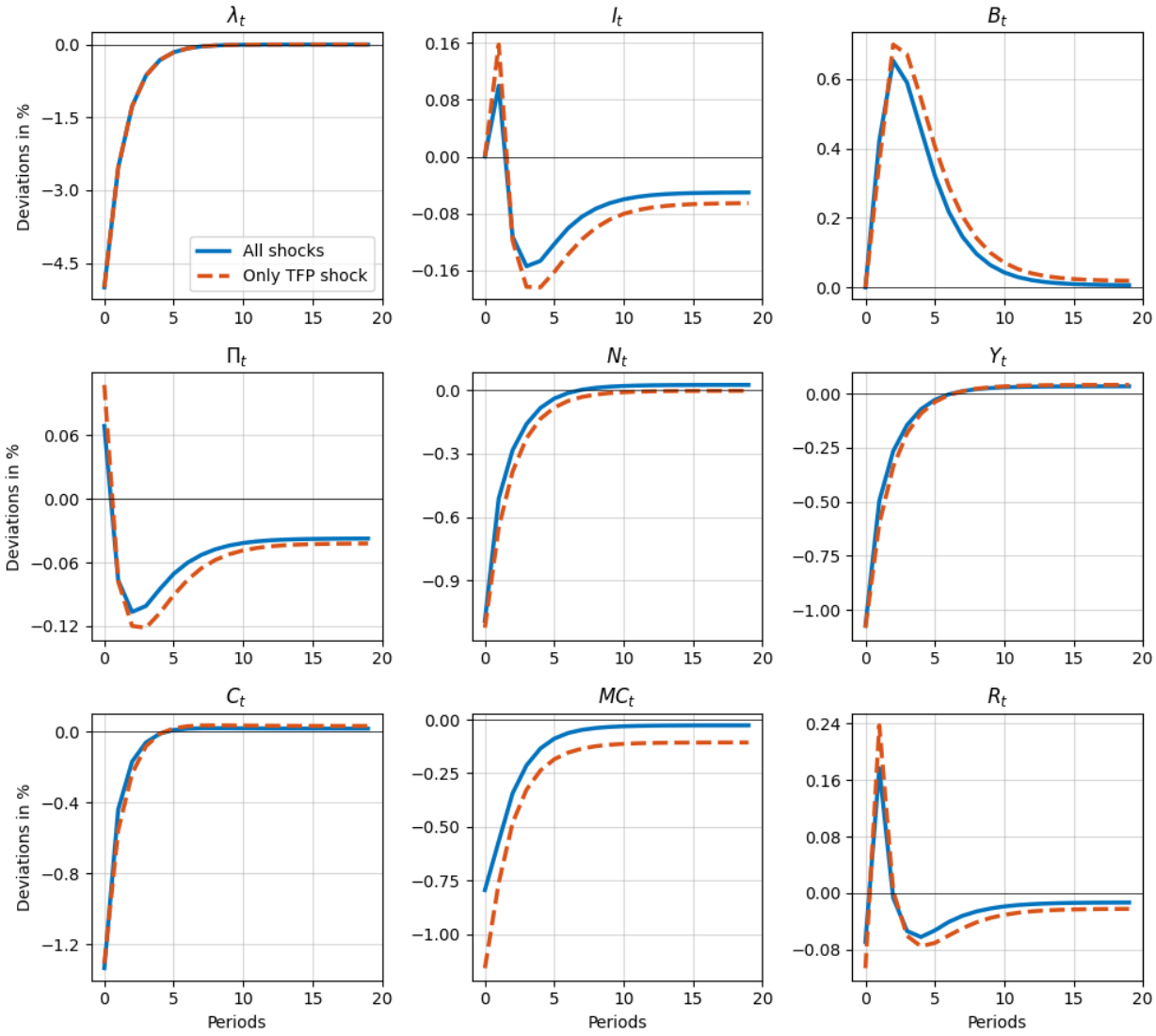

Figure 11 Generalized impulse response functions for a heterogeneous agent New Keynesian economy with a financial friction, solved with generative economic modeling. Each panel shows percentage deviations from the stochastic steady state. We initialize the economy at its stochastic steady state and apply a one time innovation to the financial wedge λ_t, which follows an AR(1) process. In both cases the AR(1) shock implies an initial decline of $7.5\,\%$ in λ_t. We compare two solution networks: one trained on all shocks (blue) and one trained only on the financial shock and a TFP shock (orange). One period corresponds to one year in the baseline calibration.

in solving complex dynamic economic models. By using neural networks trained on data generated from satellite models, we provide an alternative to standard deep learning-based approaches, which often require extensive fine-tuning and can suffer from instability due to their endogenous feedback loops. In contrast, our methodology ensures stability and scalability by using precomputed solutions from conventional methods, allowing for efficient training and accurate approximations of the full economic model.

The results demonstrate that this approach can successfully capture model dynamics with high precision, yielding prediction errors comparable to those of deep neural networks trained on full model data. We also show that the Euler equation error can be used as a direct measure of fit for our approach, enabling an ex-post validation on a case-by-case basis. Importantly, our method extends the applicability of conventional global solution methods by using recent advances in artificial intelligence. This is particularly valuable for models featuring higher-order nonlinearities and heterogeneous agents, where the curse of dimensionality poses significant computational challenges.

Our general approach offers several promising avenues in the future. First, it can be applied to more complex environments, such as solving nonlinear HANK models by training on simplified RANK and linearized HANK models. Second, it has the potential to enhance model estimation techniques, where fast and reliable solutions are critical. Lastly, training the neural network on multiple distinct models would potentially open up the possibility of using it as a general starting point for economic analysis, as different model features could then be easily added or removed.

References

Adenbaum, J., Babalievsky, F. and Jungerman, W. (2024). Learning on the Job. Working Paper.

Ahn, S., Kaplan, G., Moll, B., Winberry, T. and Wolf, C. (2017). When Inequality Matters for Macro and Macro Matters for Inequality. Tech. rep.

Algan, Y., Allais, O. and Den Haan, W. J. (2008). Solving heterogeneous-agent models with parameterized cross-sectional distributions. *Journal of Economic Dynamics and Control* **32** (3), 875–908.

Aruoba, S. B., Fernández-Villaverde, J. and Rubio-Ramirez, J. F. (2006). Comparing solution methods for dynamic equilibrium economies. *Journal of Economic dynamics and Control* **30** (12), 2477–2508.

Ashwin, J., Beaudry, P. and Ellison, M. (2025). Neural network learning for nonlinear economies. *Journal of Monetary Economics* **149**, 103723.

Auclert, A. (2019). Monetary Policy and the Redistribution Channel. *American Economic Review* **109** (6), 2333–2367.

Auclert, A., Bardóczy, B., Rognlie, M. and Straub, L. (2021). Using the Sequence-Space Jacobian to Solve and Estimate Heterogeneous-Agent Models. *Econometrica* **89** (5), 2375–2408.

Auclert, A., Rognlie, M. and Straub, L. (2024). The intertemporal keynesian cross. *Journal of Political Economy* **132** (12), 4068–4121.

Azinovic, M., Gaegauf, L. and Scheidegger, S. (2022). Deep equilibrium nets. *International Economic Review* **63** (4), 1471–1525.

Azinovic-Yang, M. and Žemlička, J. (2024). Intergenerational consequences of rare disasters.

— (2025). Deep Learning in the Sequence Space.

Barillas, F. and Fernández-Villaverde, J. (2007). A generalization of the endogenous grid method. *Journal of Economic Dynamics and Control* **31** (8), 2698–2712.

Bayer, C., Born, B. and Luetticke, R. (2024). Shocks, Frictions, and Inequality in US Business Cycles. *American Economic Review* **114** (5), 1211–1247.

Bayer, C. and Luetticke, R. (2020). Solving Heterogeneous Agent Models in Discrete Time with Many Idiosyncratic States by Perturbation Methods. *Quantitative Economics* **11** (4), 1253–1288.

Bayer, C., Luetticke, R., Pham-Dao, L. and Tjaden, V. (2019). Precautionary Savings, Illiquid Assets, and the Aggregate Consequences of Shocks to Household Income Risk. *Econometrica* **87** (1), 255–290.

Bayer, C., Luetticke, R., Weiss, M. and Winkelmann, Y. (2024). An endogenous gridpoint method for distributional dynamics. Tech. rep. ECONtribute Discussion Paper.

Bellman, R. (1957). *Dynamic Programming*. Princeton, NJ: Princeton University Press.

Bhandari, A., Bourany, T., Evans, D. and Golosov, M. (2023). A Perturbational Approach for Approximating Heterogeneous Agent Models. Tech. rep.

Bianchi, F., Melosi, L. and Rottner, M. (2021). Hitting the elusive inflation target. *Journal of Monetary Economics* **124**, 107–122.

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

Brock, W. A. and Mirman, L. J. (1972). Optimal economic growth and uncertainty: The discounted case. *Journal of Economic Theory* **4** (3), 479–513.

Brumm, J. and Scheidegger, S. (2017). Using adaptive sparse grids to solve high-dimensional dynamic models. *Econometrica* **85** (5), 1575–1612.

Canzoneri, M. B., Cumby, R. E. and Diba, B. T. (2007). Euler equations and money market interest rates: A challenge for monetary policy models. *Journal of Monetary Economics* **54** (7), 1863–1881.

Carroll, C. D. (2006). The method of endogenous gridpoints for solving dynamic stochastic optimization problems. *Economics Letters* **91** (3), 312–320.

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

Chen, M., Joseph, A., Kumhof, M., Pan, X. and Zhou, X. (2021). Deep reinforcement learning in a monetary model.

Den Haan, W. J., Rendahl, P. and Riegler, M. (2018). Unemployment (fears) and deflationary spirals. *Journal of the European Economic Association* **16** (5), 1281–1349.

Den Haan, W. J. (2010). Assessing the accuracy of the aggregate law of motion in models with heterogeneous agents. *Journal of Economic Dynamics and Control* **34** (1), 79–99.

Druedahl, J. and Røpke, J. (2025). Deep Learning Algorithms for Finite-Horizon Models.

Duarte, V., Duarte, D. and Silva, D. H. (2024). Machine learning for continuous-time finance. *The Review of Financial Studies* **37** (11), 3217–3271.

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

Eftekhari, A., Juillard, M., Rion, N. and Scheidegger, S. (2025). Scalable Global Solution Techniques for High-Dimensional Models in Dynare. *arXiv preprint arXiv:2503.11464*.

Eftekhari, A. and Scheidegger, S. (2022). High-dimensional dynamic stochastic model representation. *SIAM Journal on Scientific Computing* **44** (3), C210–C236.

Faccini, R., Lee, S., Luetticke, R., Ravn, M. O. and Renkin, T. (2024). Financial frictions: Micro vs. macro volatility. Tech. rep. Danmarks Nationalbank Working Papers.

Fernández-Villaverde, J., Gillingham, K. and Scheidegger, S. (2024). Climate Change Through the Lens of Macroeconomic Modeling. NBER Working Paper 32963. 1050 Massachusetts Avenue, Cambridge, MA 02138: National Bureau of Economic Research.

Fernández-Villaverde, J., Hurtado, S. and Nuno, G. (2023). Financial frictions and the wealth distribution. *Econometrica* **91** (3), 869–901.

Fernández-Villaverde, J., Marbet, J., Nuño, G. and Rachedi, O. (2024). Inequality and the zero lower bound. *Journal of Econometrics*, 105819.

Fernández-Villaverde, J., Nuño, G. and Perla, J. (2024). Taming the curse of dimensionality: quantitative economics with deep learning. Tech. rep. National Bureau of Economic Research.

Friedl, A., Kübler, F., Scheidegger, S. and Usui, T. (2023). Deep uncertainty quantification: with an application to integrated assessment models. Tech. rep. Working Paper University of Lausanne.

Gornemann, N., Kuester, K. and Nakajima, M. (2016). Doves for the Rich, Hawks for the Poor? Distributional Consequences of Monetary Policy. *International Finance Discussion Paper* **2016** (1167), 1–40.

Gorodnichenko, Y., Maliar, L., Maliar, S. and Naubert, C. (2021). Household savings and monetary policy under individual and aggregate stochastic volatility.

Greenwood, J., Hercowitz, Z. and Huffman, G. (1988). Investment, Capacity Utilization, and the Real Business Cycle. *American Economic Review* **78** (3), 402–417.

Gu, Z., Lauriere, M., Merkel, S. and Payne, J. (2024). Global solutions to master equations for continuous time heterogeneous agent macroeconomic models. *arXiv preprint arXiv:2406.13726*.

Guerrieri, V. and Lorenzoni, G. (2017). Credit Crises, Precautionary Savings, and the Liquidity Trap. *The Quarterly Journal of Economics* **132** (3), 1427–1467.

Han, J., Yang, Y. and E, W. (2021). DeepHAM: A global solution method for heterogeneous agent models with aggregate shocks.

Hayashi, F. (1982). Tobin's Marginal q and Average q: A Neoclassical Interpretation. *Econometrica* **50** (1), 213.

Heer, B. and Maussner, A. (2024). *Dynamic general equilibrium modeling*. Springer.

Hintermaier, T. and Koeniger, W. (2010). The method of endogenous gridpoints with occasionally binding constraints among endogenous variables. *Journal of Economic Dynamics and Control* **34** (10), 2074–2088.

Ilut, C. L., Luetticke, R. and Schneider, M. (2025). HANK's Response to Aggregate Uncertainty in an Estimated Business Cycle Model. Tech. rep. National Bureau of Economic Research.

Judd, K. L. (1998). *Numerical Methods in Economics*.

Jungerman, W. (2024). Dynamic Monopsony and Human Capital. Working Paper.

Kahou, M. E., Fernández-Villaverde, J., Gomez-Cardona, S., Perla, J. and Rosa, J. (2024). Spooky Boundaries at a Distance: Inductive Bias, Dynamic Models, and Behavioral Macro. Tech. rep. National Bureau of Economic Research.

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

Kaplan, G., Moll, B. and Violante, G. L. (2018). Monetary Policy According to HANK. *American Economic Review* **108** (3), 697–743.

Kaplan, G. and Violante, G. L. (2018). Microeconomic heterogeneity and macroeconomic shocks. *Journal of Economic Perspectives* **32** (3), 167–194.

Kase, H., Melosi, L. and Rottner, M. (2022). Estimating Nonlinear Heterogeneous Agents Models with Neural Networks. Tech. rep. DP17391. Paris & London: CEPR Discussion Paper No. 17391.

King, R. G., Plosser, C. I. and Rebelo, S. T. (1988). Production, growth and business cycles. *Journal of Monetary Economics* **21** (2-3), 195–232.

Krusell, P. and Smith, A. A. (1997). Income and Wealth Heterogeneity, Portfolio Choice, and Equilibrium Asset Returns. *Macroeconomic Dynamics* **1** (02).

— (1998). Income and Wealth Heterogeneity in the Macroeconomy. *Journal of Political Economy* **106** (5), 867–896.

Kübler, F., Scheidegger, S. and Surbek, O. (2025). Using Machine Learning to Compute Constrained Optimal Carbon Tax Rules.

Lin, A. and Peruffo, M. (2024). Aggregate Uncertainty, HANK, and the ZLB.

Luetticke, R. (2021). Transmission of Monetary Policy with Heterogeneity in Household Portfolios. *American Economic Journal: Macroeconomics* **13** (2), 1–25.

Maliar, L. and Maliar, S. (2022). Deep learning classification: Modeling discrete labor choice. *Journal of Economic Dynamics and Control* **135**, 104295.

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

McKay, A., Nakamura, E. and Steinsson, J. (2016). The Power of Forward Guidance Revisited. *American Economic Review* **106** (10), 3133–3158.

McKay, A. and Reis, R. (2016). The Role of Automatic Stabilizers in the U.S. Business Cycle. *Econometrica* **84** (1), 141–194.

Meriküll, J. and Rottner, M. (2025). Monetary policy and earnings inequality: inflation dependencies. Tech. rep. Bank for International Settlements.

Miranda, M. J. and Fackler, P. L. (2004). *Applied computational economics and finance*. MIT press.

Nord, L., Peruffo, M. and Mendicino, C. (2024). Distributive Effects of Banking Sector Losses. *Available at SSRN*.

Nuño, G., Renner, P. and Scheidegger, S. (2024). Monetary policy with persistent supply shocks.

Pascal, J. (2024). Artificial neural networks to solve dynamic programming problems: A bias-corrected Monte Carlo operator. *Journal of Economic Dynamics and Control* **162**, 104853.

Payne, J., Rebei, A. and Yang, Y. (2024). Deep learning for search and matching models.

Ravn, M. O. and Sterk, V. (2017). Job uncertainty and deep recessions. *Journal of Monetary Economics* **90**, 125–141.

— (2020). Macroeconomic Fluctuations with HANK&SAM: an Analytical Approach. *Journal of the European Economic Association* **19** (2), 1162–1202.

Reiter, M. (2009). Solving heterogeneous-agent models by projection and perturbation. *Journal of Economic Dynamics and Control* **33** (3), 649–665.

Richter, A. W., Throckmorton, N. A. and Walker, T. B. (2014). Accuracy, speed and robustness of policy function iteration. *Computational Economics* **44**, 445–476.

Schaab, A. (2020). Micro and macro uncertainty. *Available at SSRN 4099000*.

Tauchen, G. (1986). Finite state markov-chain approximations to univariate and vector autoregressions. *Economics Letters* **20** (2), 177–181.

Valaitis, V. and Villa, A. T. (2024). A machine learning projection method for macro-finance models. *Quantitative Economics* **15** (1), 145–173.

Young, E. R. (2010). Solving the incomplete markets model with aggregate uncertainty using the Krusell–Smith algorithm and non-stochastic simulations. *Journal of Economic Dynamics and Control* **34** (1), 36–41.

Appendix

A Appendix: Solution of the Asset Pricing Model

This section illustrates our methodology using an analytical asset-pricing model and develops intuition for why the methodology works. In all our applications, the solution of a model is a set of policy functions that express controls as a function of state variables. In this first analytical example, the control variable we are looking for is the price of a nominal bond q_t. Let's assume that the price satisfies the Euler equation:

$$q_t = \beta \mathbb{E}_t \left[\frac{1}{\Pi_{t+1}} \frac{C_{t+1}^{-\gamma}}{C_t^{-\gamma}} \right] = \beta \mathbb{E}_t \Big[\exp\big(-\pi_{t+1} - \gamma \Delta c_{t+1} \big) \Big]$$

with $\pi_t \equiv \ln \Pi_t$, $\Delta c_t = \ln C_t - \ln C_{t-1}$, where Π_t is gross inflation between period t and $t-1$, C_t denotes consumption, as well as β, and γ are the discount factor and risk-aversion, respectively. We can write the equation more compact as

$$q_t = \beta \mathbb{E}_t \exp(\vec{\psi}' y_{t+1}), \tag{62}$$

where $y_t = [\pi_t, \Delta c_t]'$ and $\vec{\psi} = [-1, -\gamma]'$. Hence, the control variable q_t is a forward-looking variable, which depends nonlinearly on the expectations over the dynamics of the state variables y_{t+1}. In quantitative models, the policy functions we aim to solve for have a similar form as (62). For example, households make consumption-savings decisions and firms make capital accumulation decision by forming expectations about the future. To solve for the exact policy function, we need to introduce a law of motion for the states. We assume that the dynamics of the states y_t are described by a VAR(1) without intercept

$$y_t = A_1 y_{t-1} + e_t, \tag{63}$$

with $e_t = \eta \epsilon_t$, where ϵ_t is distributed $N(0, I)$. η is a $n_y \times n_\epsilon$ matrix, where $n_y = 2$ is the number of states, and n_ϵ is the number of shocks. This implies that e_t is distributed according to a $N(0, \Sigma)$, with variance-covariance matrix $\Sigma = \eta\eta'$. We give the shocks an economic interpretation by assuming $\epsilon_t = [\epsilon_t^a, \epsilon_t^\zeta, \epsilon_t^\mu]'$. Hence, the first shock denotes a TFP-shock, the second shock denotes a discount factor shock, and the last shock denotes a markup shock. The η matrix then denotes the loadings of the shocks onto the state variables. With these assumptions, the solution for the asset price q_t can be expressed as[25]

[25] The VAR(1) specifies that the vector of variables y_t is distributed according to a multivariate normal distribution. Together with the identity $\mathbb{E}_t \exp(x_{t+1}) = \exp(\mu_x + \frac{1}{2}\Sigma_x)$ for a normally distributed vector

$$q_t = \beta \exp\left(\vec{\psi}'\mu_t + \frac{1}{2}\vec{\psi}' V_t \vec{\psi}\right) \tag{64}$$

where $\mu_t = A_1 y_t$, denotes the conditional mean forecast, and $V_t = \Sigma$ the conditional variance of the variables in y_t. Let η_{ij}, and a_{ij} denote the entries in the i'th row and j'th column of the shock impact matrix η and the matrix of the VAR A_1 respectively. Then the full solution for the price of the asset can be written as

$$\begin{aligned} q_t = \beta \exp\Bigg(& -(a_{11} + \gamma a_{21})\,\pi_t \; - \; (a_{12} + \gamma a_{22})\,\Delta c_t \\ & + \tfrac{1}{2}\Big[(\eta_{11} + \gamma\eta_{21})^2 + (\eta_{12} + \gamma\eta_{22})^2 + (\eta_{13} + \gamma\eta_{23})^2\Big]\Bigg). \end{aligned} \tag{65}$$

Equation (65) expresses how the bond price depends on the inclusion of different shocks to our dynamic system of equations. While the first line components remain unchanged, the second line changes depending on the shocks that we include in the model. Consequently, this model serves as a natural illustration for the functionality of our approach.

When solving a simplified version of the above model that contains only two of the three shocks, the entries $\eta_{1\cdot}$ and $\eta_{2\cdot}$ are equal to zero, where the $\cdot$ is a placeholder for the shock which is not included, anymore. Let us denote the resulting equilibrium price without a shock i by $q_t^{\setminus i}$. Consequently, simplified model versions feature the prices

$$\begin{aligned} q_t^{\setminus a} = \beta \exp\Bigg(& -(a_{11} + \gamma a_{21})\,\pi_t \; - \; (a_{12} + \gamma a_{22})\,\Delta c_t \\ & + \tfrac{1}{2}\Big[(0 + \gamma\cdot 0)^2 + (\eta_{12} + \gamma\eta_{22})^2 + (\eta_{13} + \gamma\eta_{23})^2\Big]\Bigg) \end{aligned} \tag{66}$$

$$\begin{aligned} q_t^{\setminus \zeta} = \beta \exp\Bigg(& -(a_{11} + \gamma a_{21})\,\pi_t \; - \; (a_{12} + \gamma a_{22})\,\Delta c_t \\ & + \tfrac{1}{2}\Big[(\eta_{11} + \gamma\eta_{21})^2 + (0 + \gamma\cdot 0)^2 + (\eta_{13} + \gamma\eta_{23})^2\Big]\Bigg) \end{aligned} \tag{67}$$

$$\begin{aligned} q_t^{\setminus \mu} = \beta \exp\Bigg(& -(a_{11} + \gamma a_{21})\,\pi_t \; - \; (a_{12} + \gamma a_{22})\,\Delta c_t \\ & + \tfrac{1}{2}\Big[(\eta_{11} + \gamma\eta_{21})^2 + (\eta_{12} + \gamma\eta_{22})^2 + (0 + \gamma\cdot 0)^2\Big]\Bigg) \end{aligned} \tag{68}$$

$x_{t+1} \sim N(0, \Sigma_x)$ we obtain the closed form expression for the asset price.

Equations (66) - (68) illustrate that shutting down one individual shocks reduces the price of the risk-free bond compared to the specification with all shocks by a constant. Moreover, we can also illustrate the solution of the bond price with only one shock active at each time. For each shock i, we indicate the equilibrium price of this bond as q^i:

$$q_t^a = \beta \exp\Bigg(-(a_{11}+\gamma a_{21})\,\pi_t \; - \; (a_{12}+\gamma a_{22})\,\Delta c_t + \tfrac{1}{2}\Big[(\eta_{11}+\gamma\eta_{21})^2 + (0+\gamma\cdot 0)^2 + (0+\gamma 0)^2\Big]\Bigg) \tag{69}$$

$$q_t^\zeta = \beta \exp\Bigg(-(a_{11}+\gamma a_{21})\,\pi_t \; - \; (a_{12}+\gamma a_{22})\,\Delta c_t + \tfrac{1}{2}\Big[(0+\gamma\cdot 0)^2 + (\eta_{12}+\gamma\eta_{22})^2 + (0+\gamma 0)^2\Big]\Bigg) \tag{70}$$

$$q_t^\mu = \beta \exp\Bigg(-(a_{11}+\gamma a_{21})\,\pi_t \; - \; (a_{12}+\gamma a_{22})\,\Delta c_t + \tfrac{1}{2}\Big[(0+\gamma\cdot 0)^2 + (0+\gamma 0)^2 + (\eta_{13}+\gamma\eta_{23})^2\Big]\Bigg) \tag{71}$$

B Appendix: Real Business Cycle Model

We use an extended stochastic RBC model composed of a firm sector, a household sector, and a government sector to test the predictive power of our method. The firm sector comprises (i) final goods producers who bundle the intermediate goods, (ii) intermediate goods producers who rent out labor services and capital from perfectly competitive markets but face monopolistic competition in the goods market as they produce differentiated goods, and (iii) producers of capital goods who turn final goods into capital subject to adjustment costs.

Households earn income from supplying labor n_{it} and capital k_{it}, and earn profits Π_{it} from owning the firm sector. Households spend their income for consumption c_{it} and capital investment k_{it+1}.

Finally, the government levies distortionary labor- and capital-income taxes with tax rates τ^L and τ^K, besides a value-added tax on consumption τ^C. Raising taxes is purely distortionary since the government returns the tax revenues to the household via lump-sum transfers T_t.

Production sector: The production sector features final, intermediate, and capital goods producers. Final goods producers bundle varieties j of differentiated intermediate goods according to the Dixit-Stiglitz aggregator

$$Y_t = \left(\int y_{jt}^{\frac{1}{\mu_t}} dj \right)^{\mu_t}, \tag{72}$$

with elasticity of substitution $\frac{\mu_t - 1}{\mu_t}$. We allow the markup μ_t to evolve stochastically as an AR(1) in its log

$$\ln \mu_t = (1 - \rho_\mu) \left(\ln \mu - \frac{\sigma_\mu^2}{2} \right) + \rho_\mu \ln \mu_{t-1} + \epsilon_t^\mu \quad \text{with} \quad \epsilon_t^\mu \sim N(0, \sigma_\mu^2). \tag{73}$$

The shock ϵ_t^μ is normally distributed with mean zero and variance σ_μ^2. Firms can adjust prices in each period, hence markup shocks only redistribute between profits and the factor incomes.[26]

Final goods producers purchase a variety of goods from a continuous range of intermediate producers indexed by j. Production of intermediate goods occurs according to constant returns to scale Cobb-Douglas production technology which combines labor N_{jt} and capital services $u_{jt}K_{jt}$ taking into account capital utilization u_{jt} according to

$$Y_{jt} = A_t \left(u_{jt} K_{jt} \right)^\alpha \left(Z_t N_{jt} \right)^{1-\alpha}, \tag{74}$$

where α denotes the capital share in the Cobb-Douglas production function, A_t denotes aggregate productivity and Z_t denotes labor-augmenting technology. Firms can choose the intensity with which they use their capital stock K_{jt} by adjusting the capacity utilization u_{jt}. An intensity higher than normal results in increased depreciation of capital according to $\delta(u_{jt}) = \delta_{0t} + \delta_1(u_{jt} - 1) + \delta_2/2(u_{jt} - 1)^2$, which is an increasing an convex function of utilization if $\delta_1, \delta_2 > 0$.

The producer minimizes costs, $w_t N_{jt} - [r_t + q_t \delta(u_{jt})] K_{jt}$, where r_t and q_t are the rental rate and the (producer) price of capital goods and w_t is the real wage. Factor markets are perfectly competitive and all intermediate goods producers are symmetric. Therefore, we drop all indices j and only refer to the aggregate variables. We can characterize the

[26] Each differentiated good is offered at price p_{jt}, the aggregate price level is $P_t = \left(\int p_{jt}^{\frac{1}{1-\mu_t}} dj \right)^{1-\mu_t}$ and demand for each of the varieties is $y_{jt} = \left(\frac{p_{jt}}{P_t} \right)^{\frac{1-\mu_t}{\mu_t}} Y_t$. In a symmetric equilibrium, this boils down to $p_{jt} = P_t \, \forall j$ and $y_{jt} = Y_t \, \forall j$ and we do not need to keep track of prices hereafter.

first-order conditions for labor and effective capital as

$$r_t + q_t\delta(u_t) = \frac{\alpha}{\mu_t} A_t u_t \left(\frac{u_t K_t}{Z_t N_t}\right)^{\alpha-1} = \frac{\alpha}{\mu_t}\frac{Y_t}{K_t}, \tag{75}$$

$$\text{and} \quad w_t = \frac{1-\alpha}{\mu_t} A_t Z_t \left(\frac{u_t K_t}{Z_t N_t}\right)^{\alpha} = \frac{1-\alpha}{\mu_t}\frac{Y_t}{N_t}. \tag{76}$$

The optimal utilization choice is given by

$$q_t\left[\delta_1 + \delta_2(u_t - 1)\right] = \frac{\alpha}{\mu_t} A_t K_t \left(\frac{u_t K_t}{Z_t N_t}\right)^{\alpha-1} = \frac{\alpha}{\mu_t}\frac{Y_t}{u_t}. \tag{77}$$

As a result, aggregate profits are $\Pi_t = \mu_t Y_t$. The logarithm of productivities A_t and Z_t evolve stochastically according to AR(1) processes

$$\ln A_t = (1-\rho_A)\left(\ln A - \frac{\sigma_A^2}{2}\right) + \rho_A \ln A_{t-1} + \epsilon_t^A \quad \text{with} \quad \epsilon_t^A \sim N(0, \sigma_A^2), \tag{78}$$

$$\text{and} \quad \ln Z_t = (1-\rho_Z)\left(\ln Z - \frac{\sigma_Z^2}{2}\right) + \rho_Z \ln Z_{t-1} + \epsilon_t^Z \quad \text{with} \quad \epsilon_t^Z \sim N(0, \sigma_Z^2). \tag{79}$$

ρ_i and σ_i^2 with $i \in \{A, Z\}$ denote the autocorrelation of the log-technology shocks and the variance of their normally distributed innovations, while A and Z denote the unconditional means of the stochastic processes. Moreover, we allow time-varying depreciation rates δ_{0t}, which evolves according to

$$\delta_{0t} = \delta_0 + \epsilon_t^\delta \quad \text{with} \quad \epsilon_t^\delta \sim N(0, \sigma_\delta^2). \tag{80}$$

Finally, capital goods producers take the relative price of capital goods, q_t, as given when determining their output. They face capital adjustment costs as in Hayashi (1982) and maximize

$$\max_{\{I_t\}_{t=0}^\infty} \mathbb{E}_0 \sum_{t=0}^{\infty} \Lambda_{0,t}\left\{q_t\left[I_t - \frac{\phi_t}{\kappa}\left(\frac{I_t}{K_t} - \delta_{0t}\right)^\kappa K_t\right] - I_t\right\}. \tag{81}$$

where $\kappa > 1$. To enhance the complexity of the mode, the adjustment costs feature a non-linear element. In particular, ϕ_t is state-dependent and depends on the level of aggregate capital:

$$\phi_t = \begin{cases} \overline{\phi} & \text{if } K_t > \overline{K} \\ \underline{\phi} & \text{if } K_t \geq \overline{K} \end{cases} \tag{82}$$

where $\overline{K} \geq \underline{K} \geq 0$. Note that capital good producers take the adjustment costs as given as they depend on aggregate capital.

Optimization yields the optimality condition

$$q_t = \left[1 - \phi_t \left(\frac{I_t}{K_t} - \delta_{0t}\right)^{\kappa-1}\right]^{-1}. \tag{83}$$

Each capital goods producer will adjust its production, until (83) is satisfied. Since all capital goods producers are symmetric, we obtain a law of motion for aggregate capital

$$K_{t+1} = (1 - \delta(u_t))\, K_t + I_t - \frac{\phi_t}{\kappa}\left(\frac{I_t}{K_t} - \delta\right)^{\kappa} K_t. \tag{84}$$

Having specified the production sector, we now describe the households in the economy.

Household sector: There exists a unit continuum of (potentially heterogeneous) households indexed by $i \in [0, 1]$ which maximize their lifetime utility discounted by the factor β. The households obtain utility from consumption c_{it} and disutility from supplying labor n_{it}. To smooth consumption, households accumulate capital k_{it+1}. The household's objective function is

$$U_{it} = \max_{c_{it}, n_{it}, k_{it+1}} \mathbb{E}_t \sum_{t=0}^{\infty} \beta^t \zeta_t u(c_{it}, n_{it}), \tag{85}$$

with $\mathbb{E}_t$ denoting the expectation operator over all stochastic processes given the information set as of time t and $u(c_{it}, n_{it})$ denotes the per period felicity function of the household over consumption c_{it} and labor n_{it}. ζ_t is a stochastic aggregate shock to the discount factor in period t. The logarithm of the discount factor shock ζ_t evolves stochastically according to an AR(1) process

$$\ln \zeta_t = -(1 - \rho_\zeta)\frac{\sigma_\zeta^2}{2} + \rho_\zeta \ln \zeta_{t-1} + \epsilon_t^\zeta \quad \text{with} \quad \epsilon_t^\zeta \sim N(0, \sigma_\zeta^2). \tag{86}$$

ρ_ζ denotes the autocorrelation of the logarithmic discount factor shock, and the shock ϵ_t^ζ is normally distributed with mean zero and variance σ_ζ^2. Households optimize the objective function (85) subject to the budget constraint

$$(1 + \tau^C)c_{it} + q_t k_{it+1} = \left(q_t + (1 - \tau^K)r_t\right) k_{it} + (1 - \tau^L) w_t h_{it} n_{it} + T_t + \Pi_{it}. \tag{87}$$

r_t and w_t denote the interest rate and wage rate as specified above and h_{it} denotes households' idiosyncratic income component. τ^C, τ^K, and τ^L denote the value-added-tax, the capital income tax, and the labor income tax, while T_t denotes the transfers the households

obtain from the government. Π_{it} denotes the individual part in aggregate profits. In all applications, households face a borrowing constraint, such that they are prohibited from holding negative amounts of assets. Individual productivity h_{it} evolves according to

$$\log h_{it} = -(1-\rho_h)\frac{\sigma_h^2}{2} + \rho_h \log h_{it-1} + \epsilon_{it}^h \quad \text{with} \quad \epsilon_{it}^h \sim N(0, \sigma_h^2). \tag{88}$$

with ϵ_{it}^h as a normally distributed shock with variance σ_ϵ^2 and mean zero.

The solution of the household problem can be characterized by the Euler equation on capital and the optimal labor supply schedule below

$$q_t u_C(c_{it}, n_{it}) = \beta \mathbb{E}_t \left[\frac{\zeta_{t+1}}{\zeta_t} \left(q_{t+1} + (1-\tau^K) r_{t+1} \right) u_C(c_{it+1}, n_{it+1}) \right] \tag{89}$$

$$-u_L(c_{it}, n_{it}) = (1-\tau^L) w_t h_{it} \frac{u_C(c_{it}, n_{it})}{1+\tau^C}. \tag{90}$$

$u_C(c_{it}, n_{it}) = \frac{\partial u}{\partial c_{it}}(c_{it}, n_{it})$ denotes the partial derivative of the felicity function with respect to consumption and $u_L(c_{it}, n_{it}) = \frac{\partial u}{\partial n_{it}}(c_{it}, n_{it})$ denotes the partial derivative of the felicity function with respect to labor.

Government sector: The government levies distortionary capital and labor income taxation at flat rates τ^L and τ^K, and claims a value-added-tax τ^C on consumption. It uses the tax revenues to finance lump-sum transfers T_t to the household. Therefore, the role of the government is purely to redistribute between factor incomes and consumption and leisure. The budget constraint is

$$T_t = \tau^C C_t + \tau^K r_t K_t + \tau^L w_t N_t. \tag{91}$$

Government transfers T_t adjust residually to make the government budget constraint hold.

Market clearing and equilibrium: The labor market, the capital market, and the goods market have to clear at all periods. Labor and capital market clearing requires

$$N_t = \int_0^1 n_{it} di \quad \text{and} \quad K_t = \int_0^1 k_{it} di. \tag{92}$$

Given these aggregate quantities, prices are determined by their marginal products on the factor inputs as denoted in equations (75) and (76). The goods market clears when

$$Y_t = C_t + I_t, \tag{93}$$

where $I_t = K_{t+1} - (1 - \delta(u_t))K_t + \frac{\phi}{\kappa}\left(\frac{I_t}{K_t} - \delta_0\right)^{\kappa} K_t$ denotes aggregate investment into next periods capital stock net of adjustment costs and $C_t = \int_0^1 c_{it} di$ is aggregate consumption. The goods market clears due to Walras-Law whenever the capital and the labor market clear.

Dynamic equilibrium: We define a dynamic equilibrium in this economy as follows. Firms and households take prices as given. Households behave optimally to maximize their lifetime utility (85) subject to the associated budget constraint (87) and the stochastic processes. Firms choose their factor inputs to maximize profits given their Cobb-Douglas production technology until the optimality conditions (75), (76), and (77). Lump sum taxes adjust such that the government budget constraint (91) holds, while the labor and asset markets (92), and the goods market (93) clear.

B.1 The analytical model

Derivation of the analytical model The part below derives the proof related to the analytical model in section 4.2.1. To solve the model analytically, we abstract from shocks to the discount factor (ζ_t) and depreciation rate (δ_t). We keep the shocks to technology (A_t), productivity (Z_t), shocks to the markup (μ_t). Finally, we abstract from capital income taxation ($\tau^K = 0$) and assume full depreciation ($\delta_{0t} = 1$).[27] Moreover, we abstract from household heterogeneity and let households be ex-ante identical by assuming away differences in idiosyncratic income $h_{i0} = 1$ and $\Pi_{it} = \Pi_t \, \forall \, i$ and initial capital holdings are identical $k_{i0} = K_0 \, \forall \, i$. We make households ex-post identical by disregarding idiosyncratic income risk $\sigma_h^2 = 0$. The absence of ex-ante or ex-post heterogeneity enables us to represent the household side through a representative agent. Therefore, we drop the individual index i to describe the variables of interest.

Proof. The proof employs a guess-and-verify approach. Guess that the policy function for savings is given by $K_{t+1} = \Gamma Y_t$. Substituting the guess into the goods market clearing condition (93) while imposing the parameter restriction $\delta = 1$ yields

$$C_t = (1 - \Gamma) Y_t.$$

[27] The combination of the assumptions renders the model unrealistic, as already noted by Brock and Mirman (1972) themselves. We do not employ the model for realistic reasons, but because it provides us with an analytical benchmark we can use. This also motivates the choice of our shocks.

Table B.1 Parameter values of the analytical Brock and Mirman (1972) model

Parameter	Value	Description	Parameter	Value	Description
Households			**Exogenous processes**		
β	0.96	Discount factor	A	1.0	Steady state TFP
γ	5	Inverse Frisch	ρ_a	0.9	TFP persistence
ω	1.0	Scale labor disutility	σ_a	{0.0, 0.01, 0.05}	TFP std.
			Z	1.0	Steady state labor prod.
Firms			ρ_z	0.9	Labor prod. persistence
α	0.33	Capital share	σ_z	{0.0, 0.01, 0.05}	Labor prod. std.
δ	1.0	Depreciation rate	μ	1.1	Steady State Markup
			ρ_μ	0.0	Markup persistence
Government			σ_μ	{0.0, 0.01, 0.05}	Markup std.
τ^L	0.0%	Labor tax rate level			
τ^R	0.0%	Capital tax rate level			
τ^C	0.0%	Value-added tax rate level			

NOTE - All parameters in the table are calibrated to a yearly frequency.

We use the two guesses and substitute into the Euler equation (89)

$$\frac{1}{C_t} = \beta \mathbb{E}_t \left[\frac{\frac{\alpha}{\mu_{t+1}} \frac{Y_{t+1}}{K_{t+1}}}{C_{t+1}} \right] \Leftrightarrow \frac{1}{(1-\Gamma)Y_t} = \beta \mathbb{E}_t \left[\frac{\alpha}{\mu_{t+1}(1-\Gamma)K_{t+1}} \right],$$

from which it is straightforward to see that $\Gamma = \frac{\alpha\beta}{\mu}$ given that $\mathbb{E}_t \frac{1}{\mu_{t+1}} = \mu^{-1}$ with $\rho_\mu = 0$. Note that the value-added-tax $(1+\tau^C)$ drops from the Euler equation, since it is constant over time. To obtain the policy function (35) we substitute the guesses with specified Γ into the labor-supply condition:

$$\frac{(1-\tau_t^N)w_t}{(1+\tau^C)C_t} = \omega N_t^\gamma \Leftrightarrow \frac{(1-\tau_t^N)(1-\alpha)}{\mu_t(1+\tau^C)(1-\Gamma)N_t} = \omega N_t^\gamma,$$

from which we obtain expression (35) when solving for N_t. □

With KPR-preferences with log-felicity over consumption the income and substitution effect of wage changes cancel out. Therefore, only shocks to the wage tax rate τ_t^L and the markup μ_t impact the level of equilibrium labor supply.

Calibration Table B.1 illustrates the parameter values that we use for solving the model. We largely use standard values from the literature, but some variables require further explanation. First, we have a steady-state markup μ of 1.5, which is very high. We introduce such a high markup value such as to amplify the effects of markup shocks on the economy. Moreover, we shut down the government by setting all tax rates equal to zero. Finally, we do not only simulate the economy with fixed volatilities of the shocks but allow for different volatility levels. While the model only features three shocks, the shocks can have different volatilities, such that we generate data sets for all three shock combinations with

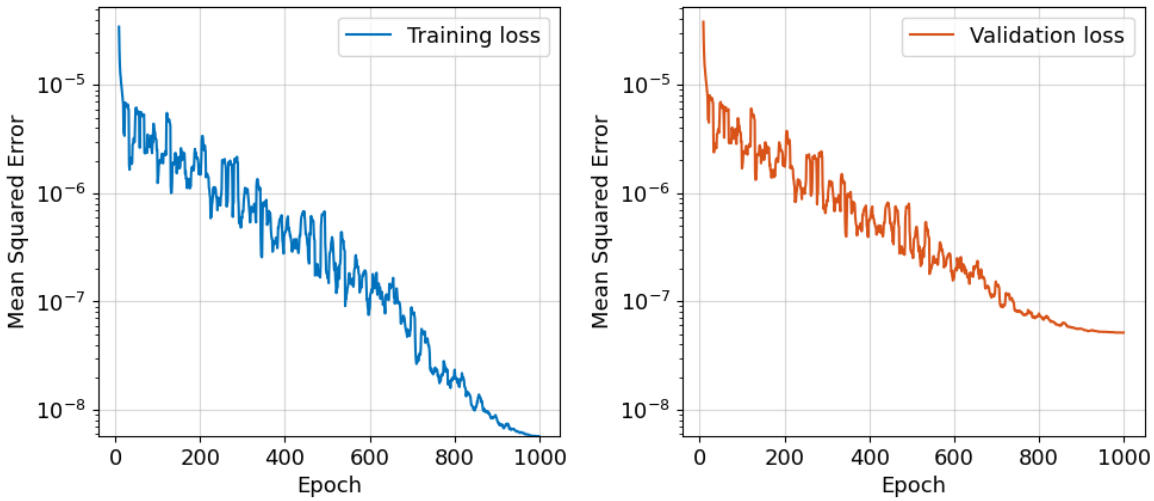

Figure B.1 Training and validation convergence for the analytical variant of the RBC model. The figure shows the loss over the training sample (left) and validation sample (right). An epoch is completed when all the training or validation sample points are utilized. The vertical axis is expressed on a logarithmic scale.

different volatilities. This also challenges the surrogate network since it needs to learn the model dynamics for different shock volatilities.

Neural network and training The network architecture features five hidden layers with 128 neurons each, linear activation functions in the input and output layers, and CELU activations in all hidden layers. We train the network using the AdamW optimizer to minimize the mean squared error between predicted and true values. The learning rate follows a cosine annealing schedule, starting at 10^{-3} and decaying below 10^{-6} over the course of training. The dataset is divided into a training and a validation sample. Figure B.1 shows the mean squared residual for the training and validation samples, which shows that the loss converges to 10^{-7}. While the validation loss is slightly larger, there is no overfitting, as the non-increase in the validation loss highlights.

B.2 Nonlinear version

Table B.2 presents the parameter choices for solving the model. Most parameters align with standard values in the literature but are calibrated at a quarterly frequency. Compared to the model in Section 3.1, we introduce partial depreciation, a capacity utilization choice, a lower Frisch elasticity of labor supply, and capital adjustment costs. Additionally, we incorporate nonlinear capital adjustment costs by allowing ϕ_t to take values $\bar{\phi}$ and $\underline{\phi}$ depending on the capital stock. The other parameters are standard.

The model features strong nonlinearities as the impulse response functions (IRFs) in

Table B.2 Parameter values of the nonlinear medium-sized RBC model

Parameter	Value	Description	Parameter	Value	Description
Households			**Firms**		
σ	1	Risk aversion	α	0.33	Capital share
β	0.99	Discount factor	δ_0	0.025	Depreciation rate
γ	1	Inverse Frisch	δ_1	0.43	Depreciation rate
ω	0.5	Scale labor disutility	δ_2	0.43	Depreciation rate
			κ	2	Cap. adj. cost curvature
			$\bar{\phi}$	2.5	High slope of cap. adj. cost
Exogenous processes			$\underline{\phi}$	0.025	Low slope of cap. adj. cost
A	1.0	Steady state TFP			
ρ_a	0.95	TFP persistence	**Government**		
σ_a	0.01	TFP std.	τ^L	0.0%	Labor tax rate level
ρ_ζ	0.95	Discount factor persistence	τ^R	0.0%	Capital tax rate level
σ_ζ	0.05	Discount factor std.	τ^C	0.0%	Value-added tax rate level
σ_δ	0.004	Depreciation std.			

NOTE - All parameters in the table are calibrated to a quarterly frequency.

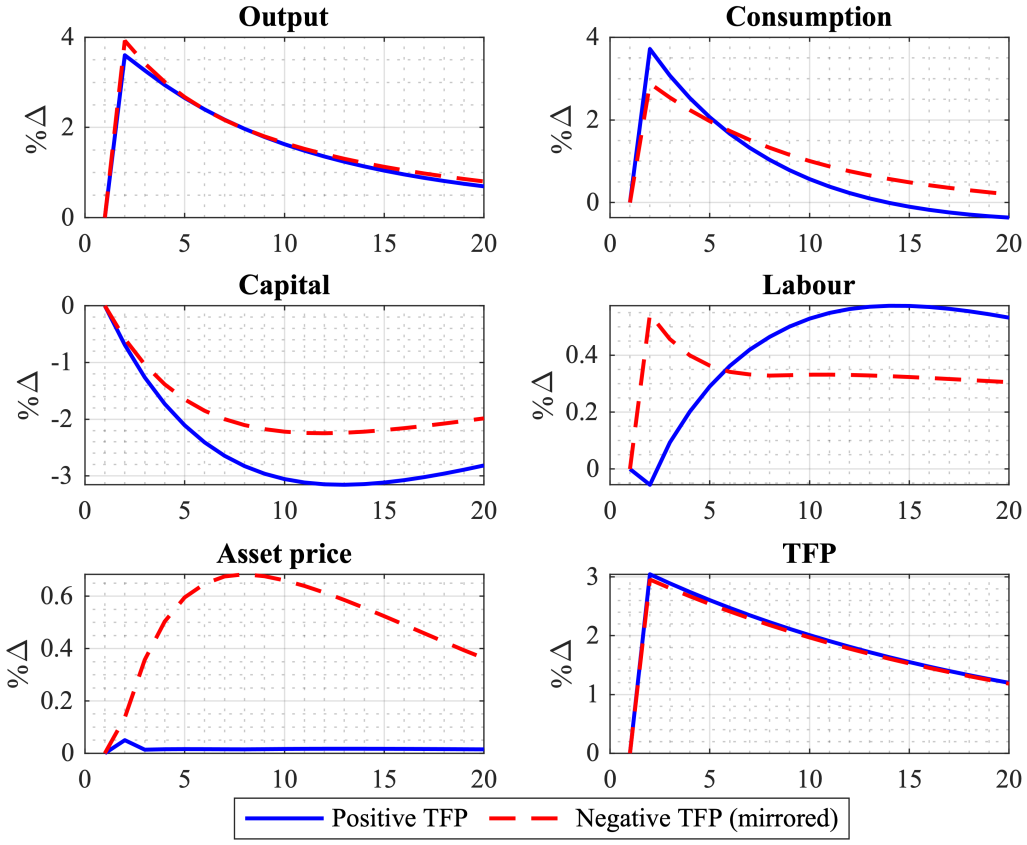

Figure B.2 IRFs and Nonlinear Propagation of the TFP Shock

Figure B.2 highlight. In this simulation, we compare the impact of an expansionary and contractionary three-standard deviation TFP shock. We display the percentage deviation from the stochastic steady state and mirror the IRFs of the negative TFP for easier comparison. The state-dependent investment costs result in strong differences between positive

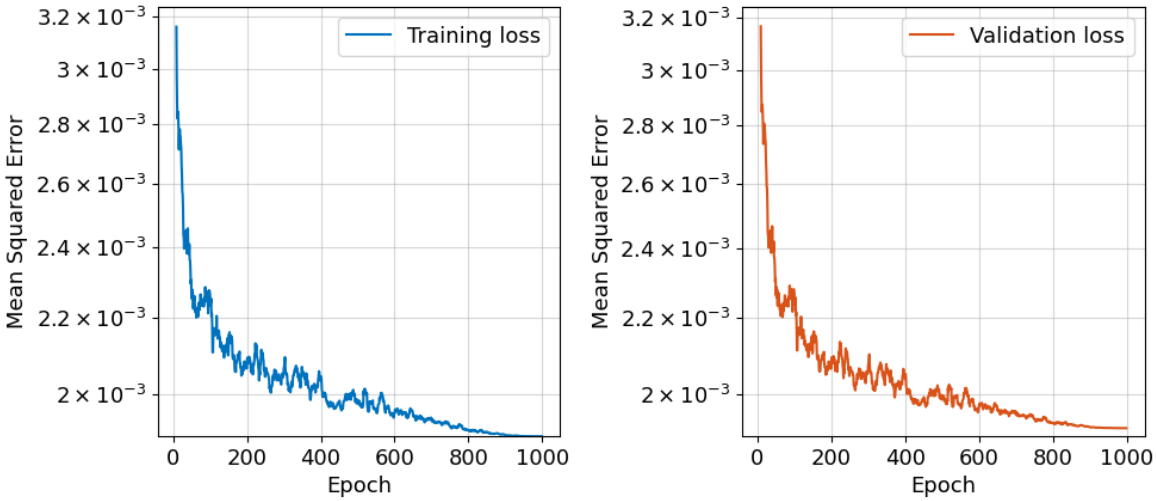

Figure B.3 Training and validation convergence for the non-linear medium scale RBC model. The figure shows the loss over the training sample (left) and validation sample (right). An epoch is completed when all the training or validation sample points are utilized. The vertical axis is expressed on a logarithmic scale.

and negative shocks. We observe a similar behavior when evaluating the preference and discount rate shock. This is an important precondition for our analysis as we want to evaluate the performance of our approach in a highly nonlinear environment.

Neural network and training The network architecture features five hidden layers with 128 neurons each, linear activation functions in the input and output layers, and CELU activations in all hidden layers. The optimizer employed is AdamW, and training minimizes the mean squared error between the predicted and true values. The learning rate follows a cosine annealing schedule, starting at 10^{-3} and decaying to 10^{-10}. The dataset is divided into a training and a validation sample. Figure B.3 shows the mean squared residual for the training and validation samples, which shows that the loss converges to 1.8×10^{-3}. Similarly as before, we do not observe an overfitting.

B.3 Heterogeneous agent version

Solution approach This subsection presents the solution to the heterogeneous agent model following the methodology of Krusell and Smith (1997) and Krusell and Smith (1998).[28] In the consumption-savings problem, households require a prediction of next period's capital given today's state space. With heterogeneous agents, this would typically require households to track the entire distribution of households over the state space, Θ_t

[28] The code is available at `https://github.com/Fabio-Stohler/KS1998`.

as an additional state variable, which renders the problem numerically intractable. To address this, Krusell and Smith demonstrate that households do not need to keep track of the full distribution Θ_t, but a few moments of the distribution suffice to forecast future capital. Specifically, they approximate the law of motion for capital using its mean, allowing households to form expectations based on a simplified perceived law of motion. Let $\vec{A}$, $\vec{\zeta}$, and $\vec{\delta}$ denote the discretized grid of aggregate productivity, discount factor, and depreciation shock with N_A, N_ζ, and N_δ shocks. We generalize on the original paper and assume that the law of motion takes the following state-dependent functional form:

$$\begin{aligned}\ln K_{t+1} = \beta_0 &+ \sum_{i=1}^{N_A} \beta_{A,i}\, \mathbf{1}_{\{A_t = A^i\}} + \sum_{j=1}^{N_\zeta} \beta_{\zeta,j}\, \mathbf{1}_{\{\zeta_t = \zeta^j\}} + \sum_{k=1}^{N_\delta} \beta_{\delta,k}\, \mathbf{1}_{\{\delta_t = \delta^k\}} \\ &+ \beta_K \ln K_t + \sum_{i=1}^{N_A} \gamma_{A,i}\, \mathbf{1}_{\{A_t = A^i\}} \ln K_t + \sum_{j=1}^{N_\zeta} \gamma_{\zeta,j}\, \mathbf{1}_{\{\zeta_t = \zeta^j\}} \ln K_t + \sum_{k=1}^{N_\delta} \gamma_{\delta,k}\, \mathbf{1}_{\{\delta_t = \delta^k\}} \ln K_t\end{aligned} \tag{94}$$

Our extension allows for an arbitrary many realizations of the discretized shocks and allows for both the slope and the intercept to vary with each (discretized) aggregate state value. The solution algorithm consists of an inner and an outer loop. The outer loop iterates until the coefficients in the regression equation (94) converge. Let

$$\boldsymbol{\beta}^n = \left(\beta_0^n,\ \left\{\beta_{A,i}^n\right\}_{i=1}^{N_A},\ \left\{\beta_{\zeta,j}^n\right\}_{j=1}^{N_\zeta},\ \left\{\beta_{\delta,k}^n\right\}_{k=1}^{N_\delta},\ \beta_K^n,\ \left\{\gamma_{A,i}^n\right\}_{i=1}^{N_A},\ \left\{\gamma_{\zeta,j}^n\right\}_{j=1}^{N_\zeta},\ \left\{\gamma_{\delta,k}^n\right\}_{k=1}^{N_\delta}\right)'$$

denote the vector of regression coefficients for the perceived law of motion of iteration n of the algorithm. Convergence is determined by checking whether the coefficients remain unchanged across iterations. If the coefficients changed by less then a small ϵ, the algorithm terminates.[29]

The inner loop iterates until the household problem is globally solved for a given perceived law of motion for capital. To solve the household side, we discretize the space $(k_{it}, h_{it}, K_t, A_t, \zeta_t, \delta_t)$ and use the endogenous grid-point method (EGM) of Carroll (2006) to solve the household problem given the stochastic processes and the perceived law of motion. Household policies are updated iteratively until the (inverse) marginal values of consumption converge. Once the household problem is solved globally, we aggregate and simulate the economy for T periods using the stochastic simulation method of Young (2010). Finally, using the simulated time series of capital and the aggregate states, we estimate the regression equation (94) to update the law of motion.

We illustrate the algorithm as follows. Let $\vec{k}$, $\vec{h}$, denote the discretized vectors of indi-

[29] We also verify that the true law of motion for capital closely matches the perceived law of motion, which is generally the case.

vidual capital holdings, individual productivity, respectively.

1. For each realization of aggregate states $\{K_t, A_t, \zeta_t, \delta_t\}$ compute labor L_t (which depends on the aggregate state), as well as the interest rate r_t and the wage rate w_t as the marginal products of capital and labor. For each realization of the individual state space $\{k_{it}, h_{it}, K_t, A_t, \zeta_t, \delta_t\}$ compute household incomes.

2. Initialize the coefficients for the law of motion (94). Typically, the intercepts (β_0, β_1, β_2, β_3) are set to zero, and the slopes (β_4, β_5, β_6, β_7) to one.

3. Given the coefficients for the law of motion, solve the household problem using EGM[30]:

   (a) Initialize guesses for the policy functions c^0_{it}, k^0_{it} defined on the state space $(k_{it}, h_{it}, K_t, A_t, \zeta_t, \delta_t)$. Create an initial guess for the marginal value function $\frac{\partial V^0_{it}}{\partial k_{it}} = (1+r)\frac{\partial u(c^0_{it})}{\partial c_{it}}$. The superscript denotes the iteration step mm of the EGM algorithm.

   (b) For each realization of the aggregate state today $\{K_t, A_t, \zeta_t, \delta_t\}$ forecast next period's capital stock using the perceived law of motion. Let $\tilde{K}_{t+1}$ denote the forecasted capital stock according to the perceived law of motion. Interpolate the marginal value $\frac{\partial V^{n-1}_{it}}{\partial k_{it}}$ from the exogenous grid $\vec{K}$ onto the perceived value in the next period $\tilde{K}_{t+1}$. Finally, compute the expected marginal value by integrating over the realizations of the aggregate $\{A_t, \zeta_t, \delta_t\}$ and idiosyncratic states $\{h_{it}\}$ and discount the expected value with the discount factor.[31]

   (c) Apply the inverse of the marginal utility function to the interpolated expected marginal value to find the policy function of consumption $\hat{c}_{it}$ on the endogenous grid $\tilde{k}_{it}$.

   (d) Compute the endogenous grid points $\tilde{k}_{it}$ from the budget constraint given the policy $\hat{c}_{it}$.

   (e) Interpolate the consumption policy function $\hat{c}_{it}$ from the endogenous grid $\tilde{k}_{it}$ onto the exogenous grid $\vec{k}$ to obtain an updated policy function c^m_{it}.

   (f) Enforce the borrowing constraint.

[30] There exist numerous resources that go into detail in the illustration of the method. We only highlight the differences that occur due to the presence of aggregate risk. The interested reader might consult Carroll (2006), Barillas and Fernández-Villaverde (2007), Hintermaier and Koeniger (2010), and the appendix of Bayer et al. (2019).

[31] Note that the idiosyncratic risk depends on the realization of the aggregate risk.

(g) Check convergence by verifying for a small ϵ, whether the condition $|u'^{-1}\left(\frac{\partial V_{it}^{m}}{\partial k_{it}}\right) - u'^{-1}\left(\frac{\partial V_{it}^{m-1}}{\partial k_{it}}\right)| < \epsilon$ is true. $|\cdot|$ denotes the Euclidean norm. If the condition is not satisfied, repeat steps (b)–(g).

4. With the converged global policy functions, we aggregate and simulate the economy using stochastic simulation of Young (2010) for T periods:

   (a) We set the initial capital stock to the value from the deterministic steady state of a representative agent economy and denote the capital stock as K_1.[32]

   (b) In period t, we have capital stock K_t, which is generally off-grid. To evaluate the policy functions of the household, we evaluate the individual policy function at K_t by interpolating from the exogenous grid $\vec{K}$ on the current capital stock K_t. Evaluate the policy functions at the current aggregate state $\{A_t, \zeta_t, \delta_t\}$.

   (c) Given household policies evaluated at the state realizations today, we update the household distribution using the stochastic simulation method of Young (2010).

   (d) Repeat steps (b) and (c) for T periods.

5. Discard the first 1000 periods as a burn-in sample. Use the remaining time series to update the perceived law of motion by regressing the logarithm of the capital stock on the aggregate states and the lagged logarithm of the capital stock as in equation (94). Denote the resulting regression coefficients as $\tilde{\boldsymbol{\beta}}^m$

6. Check whether $|\tilde{\boldsymbol{\beta}}^n - \boldsymbol{\beta}^{n-1}| < \epsilon$ for a small ϵ. If the condition is met, stop; otherwise, update the coefficients as $\boldsymbol{\beta}^n = \varphi\tilde{\boldsymbol{\beta}}^n + (1-\varphi)\boldsymbol{\beta}^{n-1}$ with $\varphi \in (0, 0.5)$ and repeat steps (3) - (6).

Note that the description above accounts for all aggregate shocks but also accommodates cases with fewer aggregate shocks by keeping some of them fixed. We apply steps (1) to (6) to each satellite model, generating a dataset that is then used to train the surrogate model.

Calibration Table B.3 reports the parameter values used to solve the satellite models and simulate the corresponding data. For household and firm behavior, we adopt the parameterization from Krusell and Smith (1998), calibrated at an annual frequency. The exogenous shock processes are also specified using standard annual values commonly found in the literature.

[32] We also find a distribution that has the mean of K_1 and initialize the simulation with this distribution.

Table B.3 Parameter values of heterogenous agent model

Parameter	Value	Description	Parameter	Value	Description
Households			**Exogenous processes**		
β	0.95	Discount factor	ρ_h	0.9	Idiosy. risk persistence
σ	1.0	Risk aversion	σ_h	0.15	Idiosy. risk std.
			A	1	Steady State TFP
Firms			ρ_a	0.75	TFP persistence
α	0.36	Capital share	σ_a	0.02	TFP std.
δ	0.1	Steady State depreciation	ζ	1	Steady State Discount fact.
			ρ_ζ	0.75	Discount fact. persistence
Government			σ_ζ	0.02	Discount Fact. std.
τ^L	0.0%	Labor tax rate level	σ_δ	0.01	Depreciation std.
τ^R	0.0%	Capital tax rate level			
τ^C	0.0%	VAT rate level			

NOTE - All parameters in the table are calibrated to a yearly frequency.

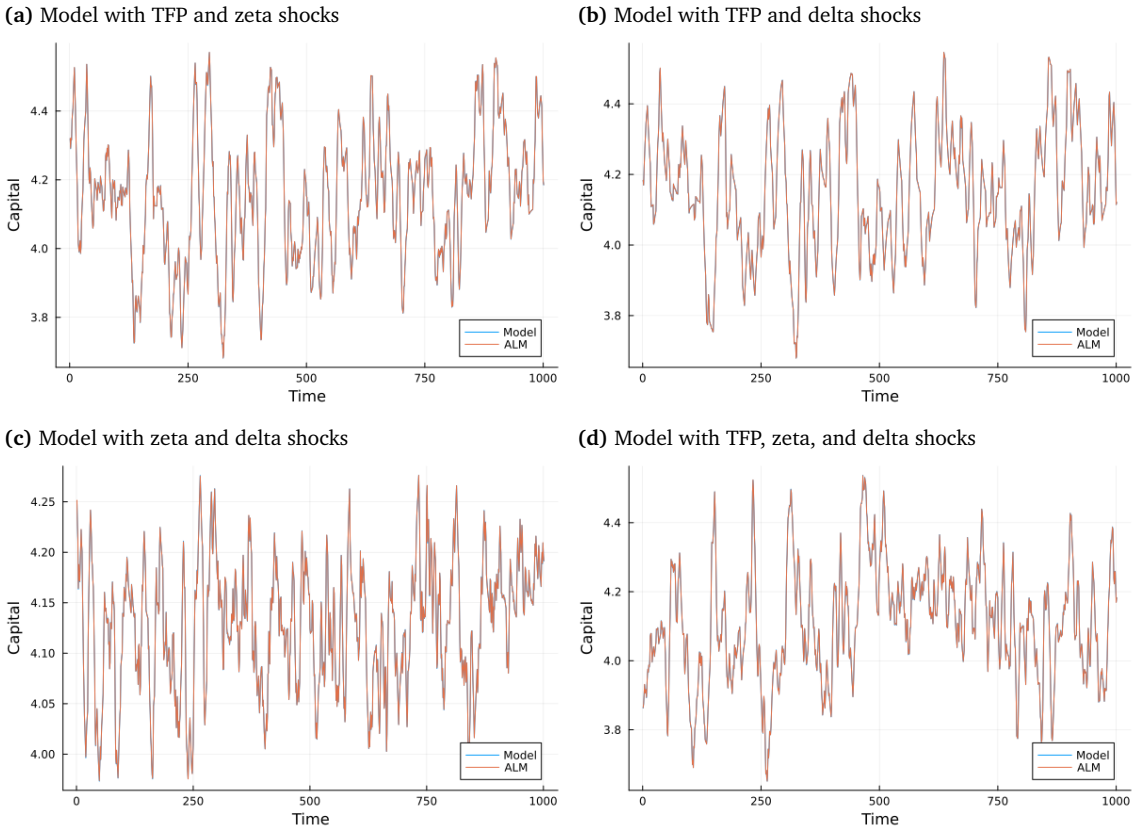

Figure B.4 Model implied series of capital and perceived aggregate law of motion (ALM)

Moreover, figure B.4 illustrates the perceived law of motion of the households in comparison to the true law of motion for capital in the economy. As the plot illustrates, the perceived law of motion and the true law of motion closely align, with the error between the two lines generally being below one percent of the capital stock.

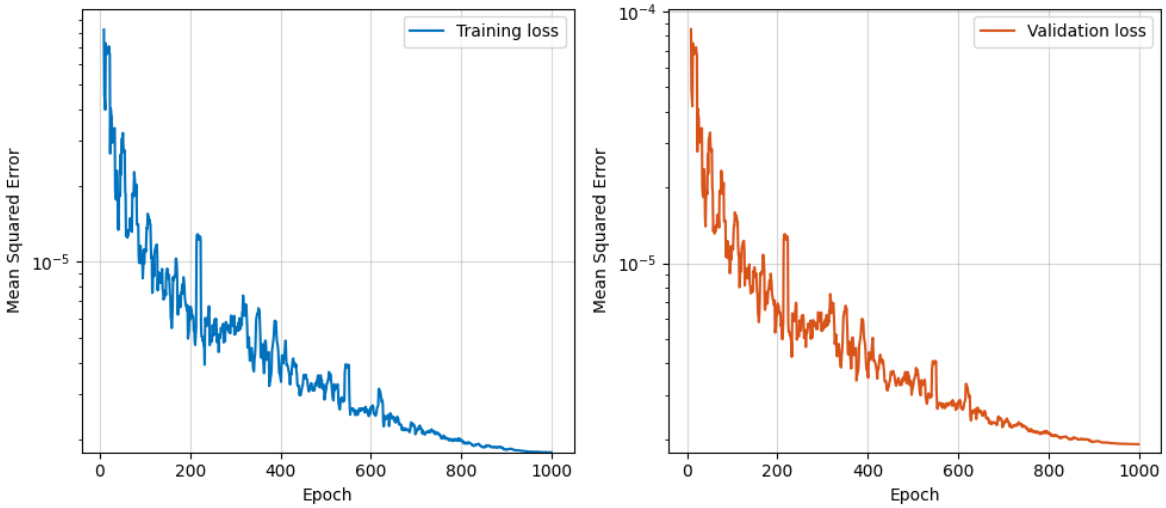

Figure B.5 Training and validation convergence for the RBC model with heterogeneous agents. The figure shows the loss over the training sample (left) and validation sample (right). An epoch is completed when all the training or validation sample points are utilized. The vertical axis is expressed on a logarithmic scale.

Neural network architecture and training The network architecture features five hidden layers with 128 neurons each, linear activation functions in the input and output layers, and CELU activations in all hidden layers. The optimizer employed is AdamW, and training minimizes the mean squared error between the predicted and true values. The learning rate follows a cosine annealing schedule, starting at 10^{-3} and decaying to 10^{-10}. The dataset is divided into a training and a validation sample. Figure shows the mean squared residual for the training and validation samples. The loss converges to 10^{-5} and there is no overfitting.

C Prediction Errors of Additional Endogenous Variables

This section illustrates additional results of of the error distributions generated by our methodology for different endogenous variables.

Figure C.1 illustrates the empirical errors for consumption C_t and labor L_t in the analytical RBC model. The complete model, as well as the generated model achieve very low prediction errors, which are centered around zero. Figure C.2 shows the identical figure, however zoomed out beyond relative errors of 0.3 percent. The figure shows that the complete and the generative models indeed achieve very low prediction errors while the incomplete submodel generates very large relative errors, that reach up to 80 percent.

Figure C.3 shows the fit of the nonlinear RBC model for other endogenous variables. With the model featuring rich nonlinearities due to the state dependency in investment

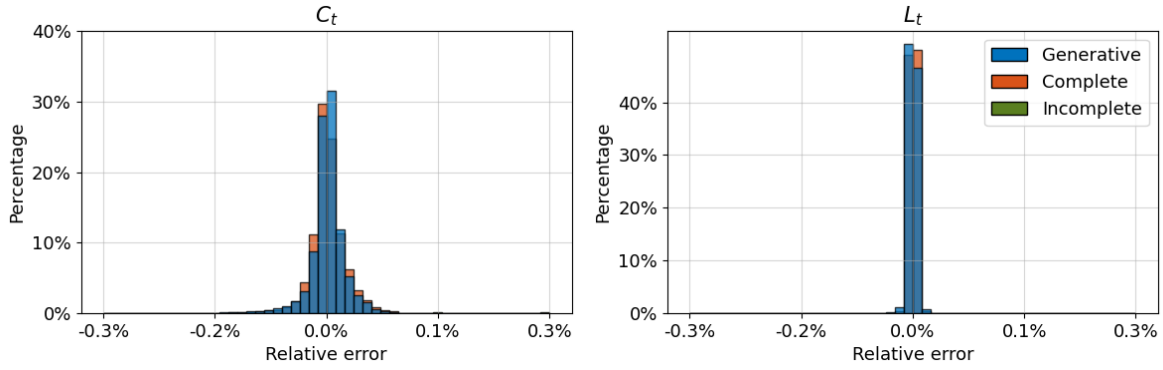

Figure C.1 Approximation error of consumption C_t and investment I_t for our generative modeling approach for the analytical RBC model zoomed in on the histogram. We compare here the simulated value to the true value for 10000 periods for our generative modeling approach (blue). It is compared to using the complete model (orange) and incomplete model (green), for which we train a neural network using data generated from the full analytical solution and from a single submodel, respectively.

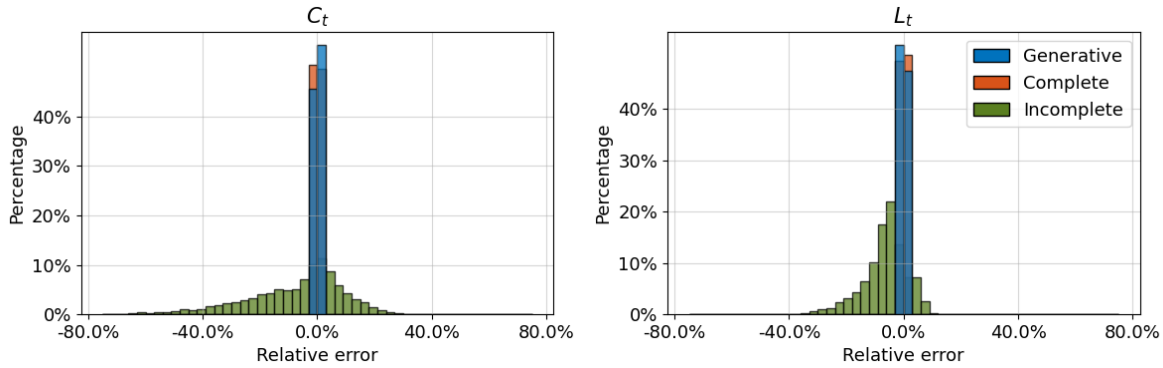

Figure C.2 Approximation error of consumption C_t and investment I_t for our generative modeling approach for the analytical RBC model. We compare here the simulated value to the true value for 10000 periods for our generative modeling approach (blue). It is compared to using the complete model (orange) and the incomplete model (green), for which we train a neural network using data generated from the full analytical solution and from a single submodel, respectively.

adjustment costs, the prediction error is largest for this proof-of-concept; however, it is consistently low. For the incomplete model, we see a left shift of the histogram for some variables due to the nonlinear adjustment costs.

Figure C.4 illustrates the fit of generative economic modeling for investment I_t and aggregate consumption C_t. For both variables, the performance of generative economic modeling is similar to that of the complete model trained on the full dataset and consistently superior to the performance of the incomplete model, trained only on a submodel. The prediction error for investment I_t has a larger variance, as the model features large depreciation shocks, which introduce additional variability in investment.

Finally, figure C.5 illustrates the prediction errors of the individual policy functions in

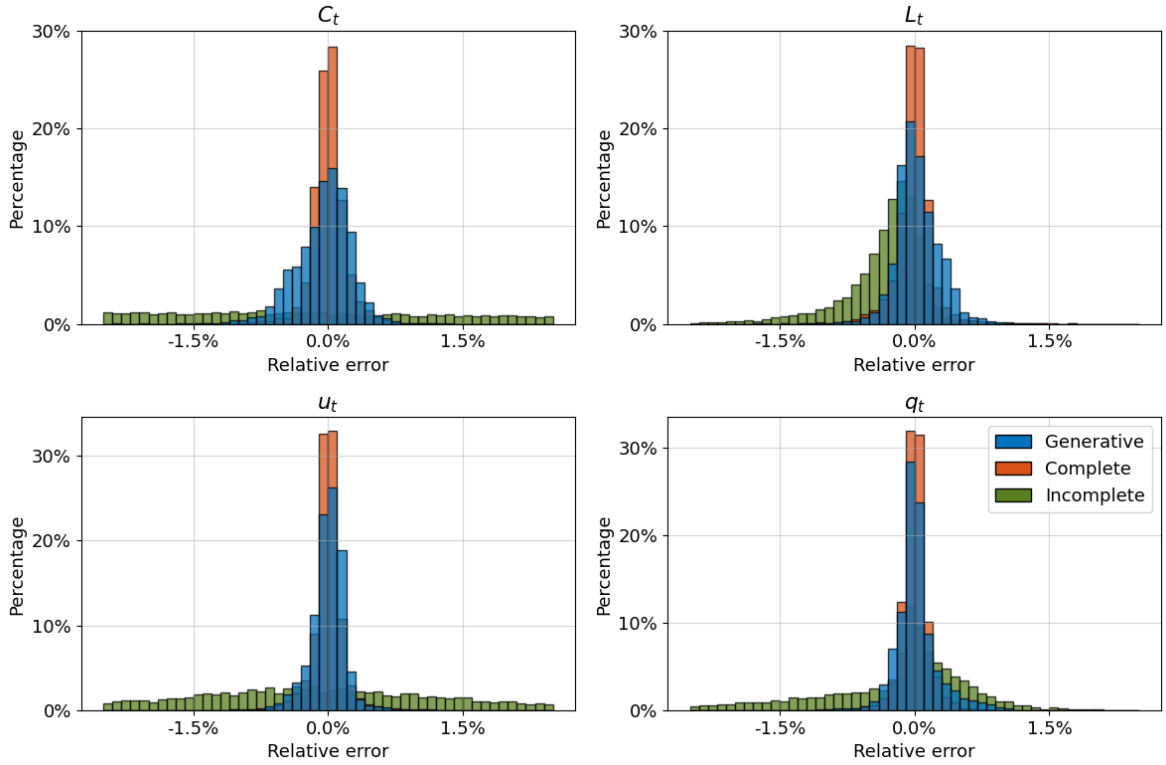

Figure C.3 Approximation error of consumption C_t and investment I_t for our generative modeling approach for the nonlinear RBC model. We compare here the simulated value to the true value for 10000 periods for our generative modeling approach (blue). It is compared to using the complete model (orange) and incomplete model (green), for which we train a neural network using data generated from the full analytical solution and from a single submodel, respectively.

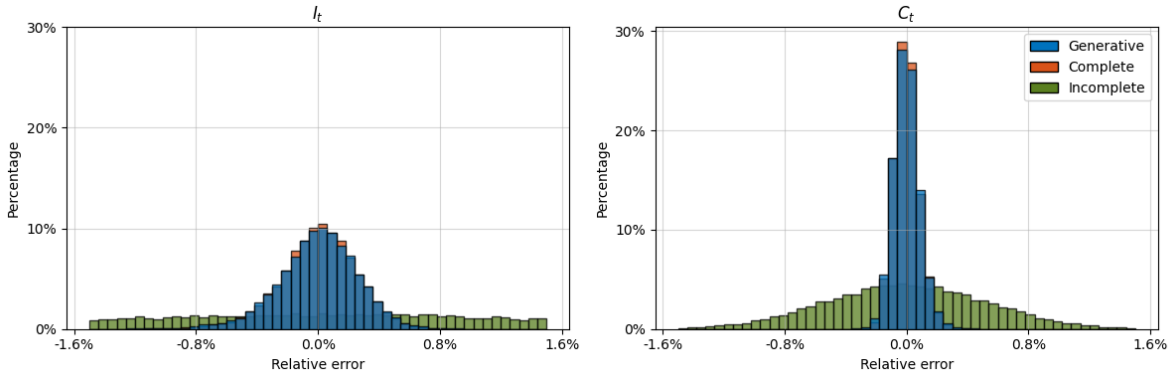

Figure C.4 Approximation error of consumption C_t and investment I_t for our generative modeling approach for the heterogeneous agent model. We compare here the simulated value to the true value for 10000 periods for our generative modeling approach (blue). It is compared to using the complete model (orange) and incomplete model (green), for which we train a neural network using data generated from the full analytical solution and from a single submodel, respectively.

the heterogeneous agent model for nine points on the idiosyncratic state space. The figure illustrates that generative economic modeling achieves comparable errors to the complete

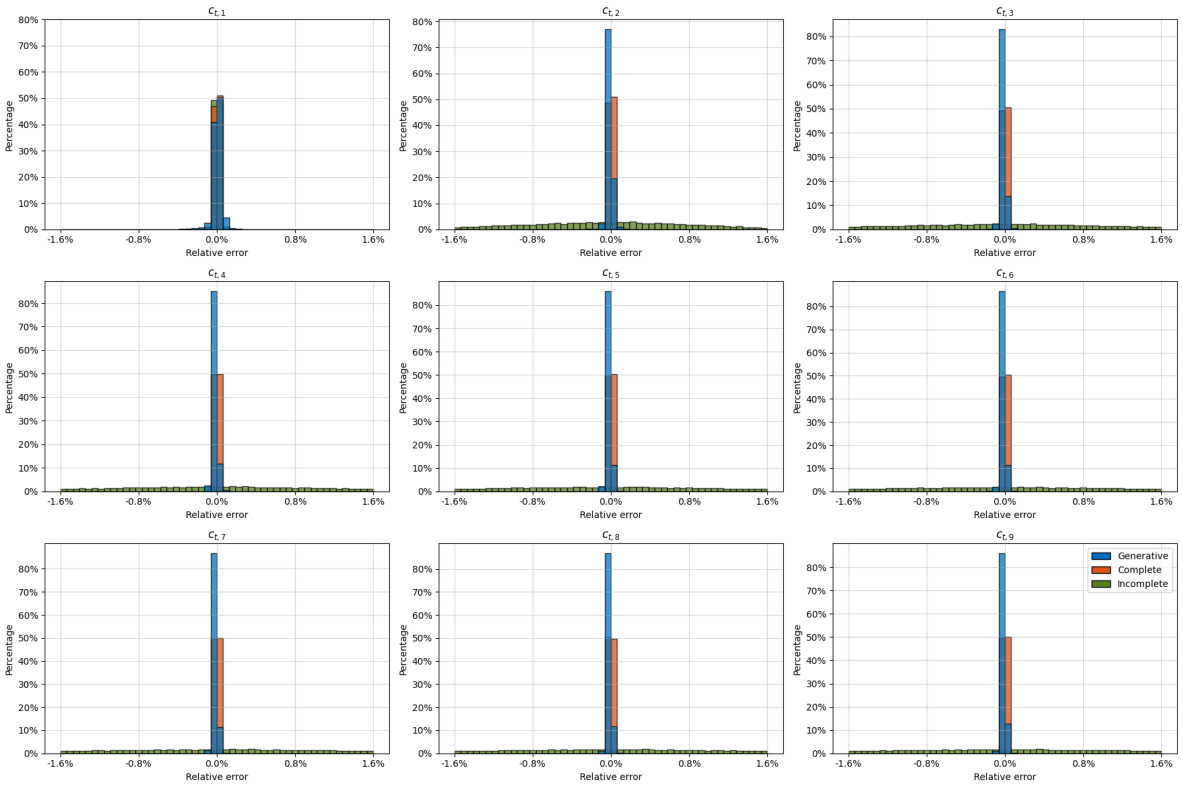

Figure C.5 Approximation error of individual consumption policies for different points at the state space for our generative modeling approach for the heterogeneous agent model. We compare here the simulated value to the true value for 10000 periods for our generative modeling approach (blue). It is compared to using the complete model (orange) and incomplete model (green), for which we train a neural network using data generated from the full analytical solution and from a single submodel, respectively.

model trained on the full data generation process, while achieving substantially lower errors than the incomplete model trained only on the data from one submodel.

Table D.1 Parameter values of HANK model with financial frictions

Parameter	Value	Description	Parameter	Value	Description
Households			**Firms**		
β	0.95	Discount factor	η	20.0	Elasticity of substitution
σ	1.0	Utility parameter	κ	0.4	Slope of the Phillips curve
γ	1.0	Curvature of utility			
ω	0.76	Scale of disutility of labor			
ρ_ϵ	0.9	Persistence of idiosync. shocks			
σ_ϵ	0.25	Std. dev. of idiosyncratic shocks			
Government			**Exogenous processes**		
ϕ_Π	1.5	Taylor rule parameter	ρ_a	0.75	Persistence of TFP shocks
γ_Π	1.107	Reaction of debt to inflation	σ_a	0.0047	Std. dev. of TFP shocks
γ_T	0.5	Reaction of debt to tax revenue	ρ_ζ	0.5	Persistence of disc. factor shocks
ρ_B	0.522	Persistence of the debt rule	σ_ζ	0.012	Std. dev. of disc. factor shocks
b^{ss}	0.9	Steady-state government debt	ρ_λ	0.5	Persistence of financial shocks
Π^{ss}	1.035	Steady-state inflation	σ_λ	0.02	Std. dev. of financial shocks
i^{ss}	1.053	Steady-state nom. interest rate	ρ_ι	0.5	Persistence mon. policy shocks
τ	0.2	Steady-state tax rate	σ_ι	0.01	Std. dev. of mon. policy shocks

NOTE — Values are reported to three decimals where applicable. All parameters in the table are calibrated to a yearly frequency.

D Heterogeneous Agent Model with Financial Frictions

To solve the financial HANK model, we build on the algorithm described in Section B.3. In contrast to the baseline implementation, which relies on a single perceived law of motion, the financial HANK model requires two: one to forecast the forward-looking component of the Phillips curve, and another to nowcast current inflation. A detailed description of the underlying algorithm can be found in the Appendix of Bayer et al. (2019), which served as the main reference for our implementation.

The parameter values used to solve the model are shown in Table D.1. We adopt an annual calibration, which speeds up computation by reducing the discount factor. This choice lowers the computational burden associated with solving the household problem which is the main numerical bottleneck. The calibration thus reflects an annualized version of the model in Bayer et al. (2019). For the exogenous shock processes, we adopt parameter estimates from the similar HANK model in Kase, Melosi and Rottner (2022), while the parameters governing the financial shock are set directly.

Figure D.1 compares the estimated equilibrium laws of motion for current inflation and the forward-looking inflation expectation with their true simulated realizations. As the figure shows, the perceived and true paths are virtually indistinguishable. The mean squared error between the perceived and true laws of motion over the entire simulation horizon is below one percent.

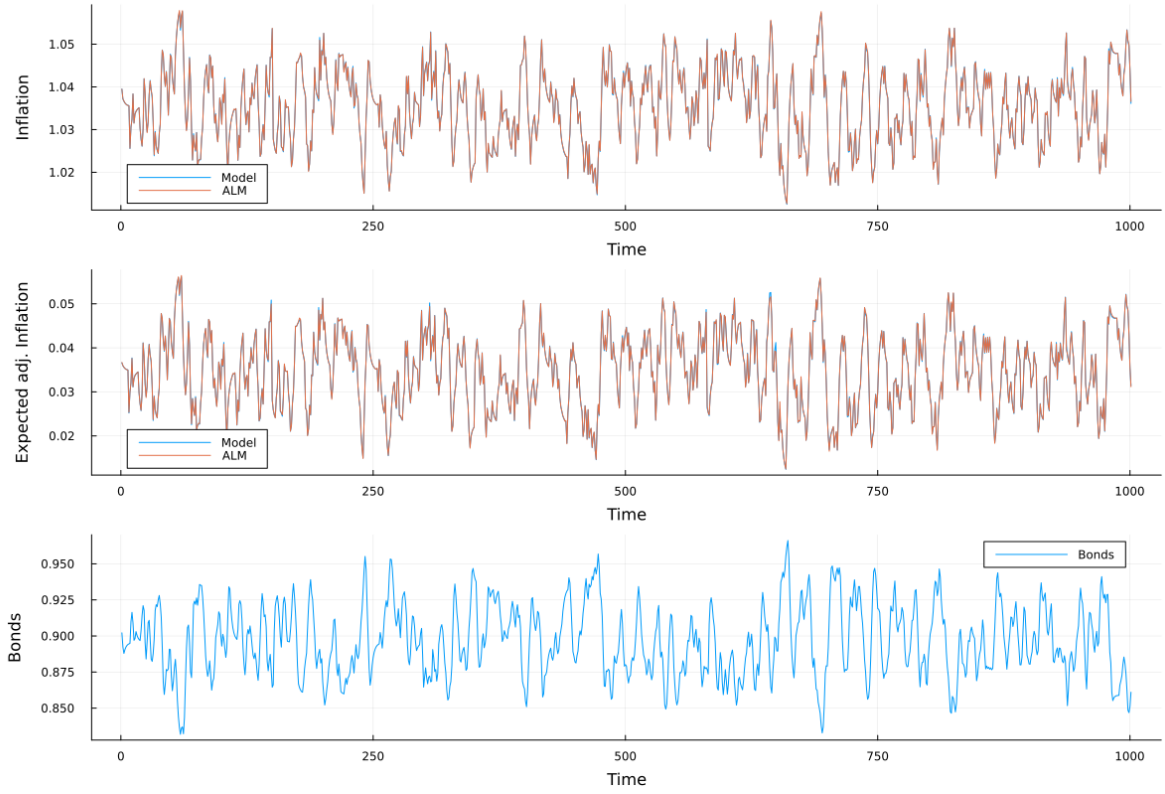

Figure D.1 Perceived Law of Motion and True Law of Motion of Financial HANK model

Previous volumes in this series

1311 November 2025	When bricks meet bytes: does tokenisation fill gaps in traditional real estate markets?	Giulio Cornelli
1310 November 2025	AI agents for cash management in payment systems	Iñaki Aldasoro and Ajit Desai
1309 November 2025	Making suptech work: evidence on the key drivers of adoption	Leonardo Gambacorta, Nico Lauridsen, Samir Kiuhan-Vásquez and Jermy Prenio
1308 November 2025	Environmental factors and capital flows to emerging markets	Jose Aurazo, Rafael Guerra, Pablo Tomasini, Alexandre Tombini and Christian Upper
1307 November 2025	When is less more? Bank arrangements for liquidity vs central bank support	Viral V Acharya, Raghuram Rajan and Zhi Quan (Bill) Shu
1306 November 2025	Big techs, credit, and digital money	Markus K Brunnermeier and Jonathan Payne
1305 November 2025	The asymmetric and heterogeneous pass-through of input prices to firms' expectations and decisions	Fiorella De Fiore, Marco Jacopo Lombardi and Giacomo Mangiante
1304 November 2025	The life experience of central bankers and monetary policy decisions: a cross-country dataset	Carlos Madeira
1303 November 2025	FX debt and optimal exchange rate hedging	Laura Alfaro, Julián Caballero and Bryan Hardy
1302 November 2025	Consumer preferences for a digital euro: insights from a discrete choice experiment in Austria	Helmut Elsinger, Helmut Stix and Martin Summer
1301 November 2025	Competing digital monies	Jon Frost, Jean-Charles Rochet, Hyun Song Shin and Marianne Verdier
1300 October 2025	The aggregate costs of uninsurable business risk	Corina Boar, Denis Gorea and Virgiliu Midrigan
1299 October 2025	Mapping the space of central bankers' ideas	Taejin Park, Fernando Perez-Cruz and Hyun Song Shin

All volumes are available on our website www.bis.org.