J. Econ. Theory 230 (2025) 106106

Contents lists available at ScienceDirect

Journal of Economic Theory

journal homepage: www.elsevier.com/locate/jet

HANK on speed: Robust nonlinear solutions using automatic differentiation

Gregor Boehl

Institute for Macroeconomics and Econometrics, University of Bonn, Adenauerallee 24-42, 53113, Bonn, Germany

ARTICLE INFO

Keywords:
Heterogeneous agents
Computational methods
General equilibrium
Nonlinear systems
JEL:
C63
C32
E52
E47

ABSTRACT

Building on automatic differentiation, I propose a robust and efficient solution method for perfect-foresight transition dynamics in heterogeneous agent models with many aggregate equations. Compared with existing methods, it allows to capture strong nonlinearities, including, e.g., occasionally binding constraints, and dynamics that deviate significantly from the steady state. A powerful and user friendly open-source reference implementation is provided, which efficiently computes nonlinear solutions to the canonical HANK model within seconds, including the transition dynamics of the full distribution. I challenge this method by studying a permanent shift in redistribution policy in a medium-scale two-asset HANK model featuring many aggregate frictions. The results indicate that, as firms seek to deplete their capital stock, the transition path is characterized by a prolonged deflationary episode, the severity of which depends on the interaction between nonlinear constraints, such as the interest rate lower bound and downward nominal wage rigidity.

1. Introduction

This paper develops a method for solving heterogeneous agent models with strong nonlinearities and many aggregate equations – the method of enhanced equilibrium propagations (EP). Unlike existing approaches that rely on linearization and thus discard nonlinear features, this method fully accounts for nonlinearities, including occasionally binding constraints, asymmetric pricing decisions, and severe financial frictions. It provides a robust and efficient way to compute the perfect foresight equilibrium path, even when the economy is far from the steady state.[1] Importantly, while the EP-method is well suited for heterogeneous agent New Keynesian models (HANK), it is equally well applicable to models featuring heterogeneity across firms, banks, or other economic agents.

At the heart of the paper, I develop a modular representation of nonlinear heterogeneous agent models that integrates the disaggregated decisions of agents with an arbitrarily large number of potentially highly nonlinear aggregate equilibrium conditions. Building on this representation, I introduce a numerical method based on Newton's approach, which efficiently solves for the nonlinear perfect foresight equilibrium path. More precisely, I show that the sub-problem of solving the system of linear equations associated with each Newton step can be tackled efficiently using a novel double-iterative procedure based on Jacobian-vector-products

E-mail address: news@gregorboehl.com, gboehl@uni-bonn.de

[1] The *perfect foresight path* is the deterministic equilibrium trajectory in which all future shocks are assumed to be zero (those are also called *MIT*-shocks) and agents correctly anticipate future aggregate variables without uncertainty. It satisfies the dynamic laws of motion during the transition between steady states or after a shock.

https://doi.org/10.1016/j.jet.2025.106106
Received 8 April 2024; Received in revised form 24 September 2025; Accepted 23 October 2025
Available online 29 October 2025

(JVPs), which exploits structural features of the solution of economic models. Thereby, the method leverages *automatic differentiation* (AD)[2] and also allows for solving the full nonlinear transition dynamics of the cross-sectional distribution between steady states.

Over the last decade, heterogeneous agent models have emerged as an important new class of macroeconomic models.[3] They allow to account for the heterogeneity of agents in their wealth, abilities, or other characteristics, thereby permitting economists to better understand the role of different groups of agents for the economy. In parallel to this rise of HANK models, recent work highlights the importance of strong nonlinearities for macroeconomic modeling. Key examples include occasionally binding constraints (e.g., the zero lower bound on nominal rates), downward nominal wage rigidity, and nonlinear labor market frictions.[4] These nonlinearities thus play a central role in the propagation of policies and economic shocks.

Due to the high complexity of heterogeneous agent models most existing solution methods yet either rely on linearized approximations, or only allow solving for nonlinear dynamics in the vicinity of the steady state. This necessarily discards the key nonlinear features of the model and limits their ability to capture the associated economic effects. Finding nonlinear solutions is especially challenging for models with many aggregate equations, which tend to raise the computational costs of using conventional nonlinear methods exponentially. This paper fills this gap by providing an efficient, fully nonlinear solution method that remains robust even when the economy is far from the steady state.

Along with the method, I provide a high-level reference implementation that I propose as a blueprint of best-practices for the provision of codes and numerical routines in economics: the *econpizza* package. The package consequently follows the open-source paradigm and comes with an extensive online documentation.[5] A core concept is the strict separation of economic model (provided by the user), underlying simulation code (provided by the implementation), and the analysis of the results (left to the user). To achieve this, I introduce a generic and standardized modeling syntax for the representation of heterogeneous agent models, which leverages the modular structure of these models: the aggregate equations on one side, and the disaggregated decision problem on the other. The implementation further showcases how to provide generic, reusable code and to adhere to the principles of modern software development. As I argue, this helps to make these methods accessible to a larger group of researchers while allowing and fostering continuous progress in the field.

I apply the proposed methods to a fundamental economic question: the macroeconomic effects of redistributive policy. I propose a medium-scale heterogeneous agent model which features the full set of frictions of contemporary DSGE models. Agents may hold two classes of assets and face idiosyncratic income risk. This gives rise to a precautionary savings motive and a non-trivial distribution of assets. If transfers are financed by labor income taxes, the distortionary effects of these taxes decreases output significantly relative to an equilibrium without transfers. My methodology allows to study the transition dynamics of the complete cross-sectional distribution between the two equilibria. As I show, these dynamics are strongly deflationary in the short and medium term due to the depletion of the capital stock, and their recessionary nature may be intensified by severe nonlinearities such as a binding lower bound on the nominal interest rate or downward nominal wage rigidity.

Literature

The idea of solving economic models in sequence space dates back to Fair and Taylor (1980), which exclusively treats representative agent models. Building on this idea, Laffargue (1990) and Juillard et al. (1996) show that the block tridiagonal structure of the sequence space Jacobian of representative agent models can be exploited to efficiently solve the system of linear equations during each Newton step. Juillard et al. (1998) further show that Newton-based methods are advantageous over first-order algorithms in terms of robustness and speed. This algorithm is, for completeness, detailed in Appendix C. It is also provided in the reference implementation where AD is used to calculate the single Jacobian blocks. Unfortunately, the sequence space Jacobian of *heterogeneous* agent models does not inherit such handy block tridiagonal structure, nor can we safely ex-ante assume its sparsity.[6] For this reason, the EP method combines the knowledge of the steady state sequence space Jacobian with the information readily obtainable via AD to iteratively solve the system of linear equations during each Newton step efficiently.

The sequence space approach to heterogeneous agent models was introduced by Boppart et al. (2018) and further advanced by Auclert et al. (2021). Both methods focus on linearized solutions in the direct neighborhood of the steady state. Boppart et al. (2018) propose a small-scale model which features very few aggregated variables and equations. They show that the underlying

[2] Automatic differentiation is a computational technique for efficiently computing the derivatives of a function without having to write out the derivative by hand or use numerical methods. Section 2 gives a short primer on AD and its virtues.

[3] Important applications featuring heterogeneous households include, e.g., Kaplan et al. (2018), Gornemann et al. (2016), McKay et al. (2016), Auclert and Rognlie (2017, 2018), Ahn et al. (2018), Auclert (2019), De Ferra et al. (2020), Hagedorn et al. (2019), Krueger et al. (2015), Bayer et al. (2020, 2023) and Achdou et al. (2022). Heterogeneous firms are prominently featured in, e.g., Golosov and Lucas (2007), Nakamura and Steinsson (2010) and Khan and Thomas (2013).

[4] E.g., see (Gust et al., 2017) for the role of the interest rate lower bound on the empirical dynamics and (Lindé and Trabandt, 2018) for the effects of nonlinearities on fiscal multipliers. Petrosky-Nadeau et al. (2018) document that nonlinear labor search frictions can induce endogenous disasters in otherwise standard models. Klenow and Kryvtsov (2008) study the role of asymmetric state-dependent heterogeneous firm pricing on inflation dynamics.

[5] The documentation can be found at https://econpizza.readthedocs.io.

[6] Indeed, when adding the distribution and agents' decisions as variables to the root finding problem, the block tridiagonal structure would persist. However, this would render the problem prohibitively large.

heterogeneous agent model can be solved by iterating on the trajectory of these aggregate variables without having to keep track of the disaggregated variables. The authors use the nonlinear impulse responses in the immediate neighborhood of the steady state to generate general linear impulse responses. Building on this, Auclert et al. (2021) provide an elegant and efficient method for calculating the steady state sequence space Jacobian (SSJ) which provides linear impulse response functions for models with many aggregated variables. Auclert et al. (2021) also use the steady state sequence space Jacobian in the context of a Newton method to find the nonlinear perfect foresight solution in the neighborhood of the steady state.[7] Section 5.3 contains an in-depth comparison of the SSJ approach with the EP method.

An alternative approach for solving heterogeneous agent models is based on the state-space representation and goes back to Reiter (2009). This approach as well returns impulse responses to the linearized model. Since the disaggregated state space of heterogeneous agent models may generally be very large, such state-space representation usually makes a state-space reduction necessary. Corresponding reduction routines are given, e.g., by Algan et al. (2008), Winberry (2018), Ahn et al. (2018), or Bayer et al. (2020), where they are also applied to the Bayesian estimation of linearized heterogeneous agent models. Additionally, Reiter (2023) provides a method that also allows for second-order perturbation solutions. These methods allow for very general functional relationships between the distribution and the aggregated economy. In contrast, the sequence space approach of Auclert et al. (2021) does not require approximation or compression of distribution but requires the existence of sufficiently good linear approximation of these functional forms. Other than that, the EP method does neither require any of the functional forms of the model to be linear, nor any approximation or compression of the distribution. Centrally, the method also allows to find the fully nonlinear perfect foresight solution even if the trajectory is very far from the steady state and is robust even to strong nonlinearities due to annealing the true Jacobian during the Newton iterations. It further allows to retain the nonlinear perfect foresight transition of the complete cross-sectional distribution.

By applying automatic differentiation to solve economic models, this paper also adds to a very recent branch of the literature which introduces machine learning tools for quantitative economics. Examples include (Scheidegger and Bilionis, 2019; Maliar et al., 2021; Kahou et al., 2021; Bianchi et al., 2021; Azinovic et al., 2022; Fernández-Villaverde et al., 2023) and, for the context of the ZLB, Fernández-Villaverde et al. (2025). These papers use deep learning networks to solve nonlinear high-dimensional macroeconomic models including aggregate uncertainty.[8] Notably, solving this type of models was deemed impossible only a few years ago and the current progress in this area is very impressive. This literature documents that it is well possible to successfully apply deep learning to solve economic models, but that these methods (and their convergence properties) are not yet well understood, may take a long time to train and require additional expert knowledge. As machine learning techniques are currently very actively researched across many fields, it is likely that they are the path forward to circumvent the curse of dimensionality and to solve complex nonlinear models even with aggregate uncertainty. This paper can be seen as an intermediary step contributing to this overarching goal.[9]

My method is based on iteratively solving the system of linear equations of each Newton step. A class of related numerical methods are the well-known Krylov subspace methods, see e.g. the generalized minimal residual method (GMRES, Saad and Schultz, 1986) and the biconjugate gradient stabilized method (Van der Vorst, 1992, BiCGSTAB). Relative to these methods the algorithm introduced here has a considerably lower computational and memory overhead and is thus much faster.[10] However, while the conditions for convergence of my method are generic to the type of nonlinear systems found in economic models, they may not be generalizable to be used outside this subclass.

The rest of this paper is structured as follows. Section 2 presents the main methodological contributions. Section 3 introduces a general class of HANK models, which serves as the framework for applying the proposed method in this paper. In Section 4 the method is used to analyze the dynamic effects of a change in government redistribution. Section 5 evaluates the method's performance through benchmarking, including a thorough comparison with its competitors. Finally, Section 6 provides concluding remarks.

2. Solving nonlinear heterogeneous agent models

This section presents the main methodological innovations. I first provide a general aggregate representation of heterogeneous agent models, and show how it can be used to find the equilibrium dynamics. Then I review the principles of automatic differentiation (AD), which are important for understanding the innovations of the main method. After that I present the core iterative procedure for finding dynamic equilibrium transition paths. Finally, I show how the steady state and the steady state Jacobian, the latter being an important ingredient to the main method, can be calculated efficiently using AD.

[7] The strategy to use the same Jacobian for each subsequent iteration is known as the *chord method*. While it may provide good results for systems with mild nonlinearities and close to the steady state, it often does not converge for more complicated models or when simulating dynamics further away from the steady state.

[8] In the context of the ZLB, aggregate uncertainty may play an important role as agents foresee the significant negative effects of a potentially binding lower bound. Approximations of these effects, such as Lin and Peruffo (2024), exist.

[9] Similarly, the online appendix of Achdou et al. (2022) also applies automatic differentiation in the context of a Newton method. Other than in the method introduced here, they calculate the full Jacobian matrix which is very costly for larger models.

[10] E.g. GMRES requires an Arnoldi iteration during each step, which can be computationally costly, and needs to store the results from previous iterations.

2.1. An aggregate representation of heterogeneous agent models

Let $(x_t)_{t\geq 0} \in \mathbb{X} \subset \mathbb{R}^n$ be the *aggregate* variables in period t including aggregate shocks and denote the *disaggregated* state space of heterogeneous agents by $\mathbb{S} \subset \mathbb{R}^{n_s}$.[11] A large class of heterogeneous agent models can be cast in the form

$$(a_t, w_t) = W(w_{t+1}, x_{t-1}, x_t, x_{t+1}), \tag{1}$$

$$d_t = D(a_t, d_{t-1}), \tag{2}$$

$$\mathbf{0} = f(x_{t-1}, x_t, x_{t+1}, d_t, a_t), \tag{3}$$

where $(w_t, a_t, d_t)_{t\geq 0}$ are time-t functions defined on $\mathbb{S}$ as follows. $w_t : \mathbb{S} \to \mathbb{W} \subset \mathbb{R}^{n_w}$ denotes agents' recursive valuations of $\mathbb{S}$, $a_t : \mathbb{S} \to \mathbb{A} \subset \mathbb{R}^{n_a}$ are their actions (or functions thereof) for a given state, and $d_t : \mathbb{S} \to [0,1]$ is the distribution of agents across $\mathbb{S}$.[12] These objects are specified in detail below.

$W(\cdot)$ is a recursive representation which maps the agents' valuations of future objects together with aggregate variables into their current valuations and actions. Often, w_t are *marginal* values as required when using variants of the endogenous grid method (EGM) of Carroll (2006), which usually are more efficient computationally.[13] The object a_t are the decisions implied by solving the Bellman equation. Since $\mathbb{S}$ is usually discretized on a grid S_g with suitably chosen grid size g, (w_t, a_t, d_t) in practice are grid-based representations of the underlying functions where w_t is a $n_w \times g$ matrix, a_t is $n_a \times g$ and d_t is the distribution over S_g represented by the g-dimensional unit hypercube $[0,1]^g$ with $\sum_j d_{jt} = 1$. Note that a_t is defined on S_g but not directly related to the distribution d_t over S_g. However, the above representation is without loss of generality as it also includes cases where W is a more complicated recursion, e.g. tracing default probabilities over time in models where agents can default on their debt. Notably, the only type of models not captured by this are models in which the agents' actions depend on the *full* distribution, i.e. where W also is a function of d_t or d_{t-1}.

The *evolution* of the distribution, $D(\cdot)$, then maps the agents' current decisions a_t and the last period's distribution d_{t-1} into the current distribution d_t. Importantly, $D(\cdot) : (d_{t-1}, a_t) \to d_t$ is a function general to all models and does not need to be provided by the user. If $\mathbb{S}$ is discretized on a grid, d_t can be constructed e.g. by using the lottery method of Young (2010). The function $f(\cdot) = z_t$ with $f : \mathbb{R}^n \to \mathbb{R}^n$ contains the n nonlinear equations describing the law-of-motion of the aggregate economy, formulated such that each equation (i.e. each residual contained in the vector z_t) must equal zero in equilibrium, i.e. $z_t = \mathbf{0}$. Future or past aggregate variables beyond $t+1$ or before $t-1$ can easily be included by introducing auxiliary variables. In the following it is assumed that W, D, and f are differentiable and that this property is retained throughout the discretization and interpolation routines.

Omitting the expectations operator on $(t+1)$-objects implies to abstract from aggregate uncertainty and focus on the perfect foresight path. For an initial state $(x_0, d_0) \in (\mathbb{X}, [0,1]^g)$, an equilibrium consists of $\langle W, D, f\rangle$ satisfied by sequences of $\{x_t, w_t, a_t, d_t\}_{t=1}^{\infty}$ for all $t = 1, 2, \ldots$ periods. Note that the above specification nests representative agent models if (w_t, a_t, d_t) are empty sets. This is because the agents intertemporal decision can then be represented by an Euler Equation and the model can be formulated in terms of the n aggregate variables x_t and the n aggregate equations in $f : \mathbb{X} \to \mathbb{R}^n$.

This representation generalizes the specification of Auclert et al. (2021) in two regards: first, the effect of idiosyncratic variables on the aggregated economy (i.e., the mapping from (w_t, a_t, d_t) in f) can take arbitrary functional forms and is not restricted to be linear in the distribution and actions. In particular, it is not necessary to explicitly cast idiosyncratic variables into aggregate output variables before supplying them to f. Instead, the distribution and the agents' actions enter f directly.[14] Second, the aggregate economy does not require a representation as a directed acyclic graph.[15]

2.2. A truncated representation of the dynamic equilibrium

Take (x_0, d_0) as given and fix a terminal period T sufficiently far in the future. Assume that (x_T, w_T) in period T are known, e.g. because the system in T is ϵ-close to a steady state. Starting with w_T and a guess for the sequence of aggregated variables $\{x_t\}_{t=1}^{T-1}$, the function $W(\cdot)$ can be iterated backwards in time, thereby providing the sequence of decisions $\{a_t\}_{t=1}^{T-1}$. Then starting with d_0, this sequence can be used to iterate the function $D(\cdot)$ forwards in time until T, resulting in the sequence of distributions $\{d_t\}_{t=1}^{T-1}$. Using boldface notation for time-sequences, $\mathbf{x} = \{x_t\}_{t=1}^{T-1}$, let backwards and forwards iteration and aggregation be represented by the functions F_a, F_d and F_x as in

$$F_a : \mathbf{x} \underset{W}{\to} \mathbf{a}, \tag{4}$$

$$F_d : \mathbf{a} \underset{D}{\to} \mathbf{d}, \tag{5}$$

[11] Any aggregate shock $\varepsilon^{\natural}$ can be included in the set of aggregate variables by shifting the timing of $\varepsilon_t^{\natural}$ one period backwards to $\varepsilon_{t-1}^{\natural}$ and adding an auxiliary equation $\varepsilon_t^{\natural} = 0$. Impulses can then be simulated by setting $\varepsilon_{t-1}^{\natural} \neq 0$. Expected shocks can be treated equivalently.

[12] Without loss of generality, there could be several distributions. I will here use the singular term for the sake of simplicity.

[13] A generalization of EGM for multiple dimensions and portfolio choice models is given by Hintermaier and Koeniger (2010). For the two-asset HANK it is necessary to track the marginal values of liquid and illiquid assets and, hence, $w_t \in S_g^2$.

[14] To be precise, the appendix of Auclert et al. (2021) provides a extension for this case which uses the Jacobians of D and f w.r.t. d_{ss}. While this is conceptionally solid, these Jacobians are very large in practice and their evaluation is very costly.

[15] The DAG representation required by SSJ has the advantage that it directly allows to reduce the state space of the model. However, finding a DAG representation may be cumbersome while, as documented in Section 5.2, a smaller state space does not necessarily reduce computation times by much.

$$F_x : (\mathbf{x}, \mathbf{d}, \mathbf{a}) \underset{f}{\to} \mathbf{z}, \tag{6}$$

where $\mathbf{z} = \{z_t\}_{t=1}^{T-1} = \{f(x_{t-1}, x_t, x_{t+1}, d_t, a_t)\}_{t=1}^{T-1}$ is the sequence of residuals from the aggregated equations. In other words, the functions F_a, F_d and F_x can be constructed by repeatedly applying the functions W, D and f. It is then straightforward to define a function $F : \mathbb{R}^{n(T-1)} \to \mathbb{R}^{n(T-1)}$ by

$$F(\mathbf{x}) = F_x(\mathbf{x}, F_a(\mathbf{x}), F_d(F_a(\mathbf{x}))) = \mathbf{z}, \tag{7}$$

which is, thus, defined in terms of the sequence of aggregate variables only.[16] It follows that a perfect foresight equilibrium trajectory $\mathbf{x}^*$ solves

$$F(\mathbf{x}^*) = \mathbf{0}, \tag{8}$$

implying that a fully nonlinear period-T truncated solution to the model in Eqs. (1) to (3) can be expressed as a $(T-1 \times n)$-dimensional root finding problem. A solution to this type of problems can be found using Newton's method. Starting with an initial guess on the equilibrium trajectory $\mathbf{x}_0$, Newton's method is given by the iteration

$$\mathbf{x}_{i+1} = \mathbf{x}_i - J(\mathbf{x}_i)^{-1} F(\mathbf{x}_i), \tag{9}$$

until $||\mathbf{x}_{i+1} - \mathbf{x}_i|| < \epsilon$, where $J(\mathbf{x}_i)$ is the Jacobian matrix of F evaluated at $\mathbf{x}_i$ and ϵ is a given (very small) stopping criterion. Note that $x_{i+1} = x_i$ once $F(x_i) = \mathbf{0}$. While Newton's method is known for quadratic convergence, applying it directly to our problem is usually impractical (if not impossible) because the calculation of the Jacobian and its inverse is prohibitively expensive. To circumvent this problem, I below present an iterative method to solve the linear system of equations associated with $J(\mathbf{x}_i)^{-1} F(\mathbf{x}_i)$ in Eq. (9) very efficiently using automatic differentiation.

2.3. A primer on automatic differentiation

Automatic differentiation (AD) is a computational technique for efficiently computing derivatives of a function. It works by using the chain rule of calculus to build up the derivative of a function from the derivatives of its constituent parts. While it is a very powerful technique, it is sometimes mistaken that AD can magically and at almost zero computational costs provide the Jacobian of any multivariate function. This, importantly, is not the case.

AD knows two distinct modes, both of which are leveraged in the presented solution method: *forward mode* and *reverse mode*. Forward accumulation is accomplished by augmenting the algebra of real numbers and obtaining a new arithmetic: the algebra of *dual numbers*. An additional component is added to every number to represent the derivative of a function at the number, and all arithmetic operators are extended for the augmented algebra. A dual number a is given by

$$a = b + c \cdot \epsilon, \tag{10}$$

such that $\epsilon > 0$ and $\epsilon^2 = 0$. The algebra of dual numbers has hence some similarities to the algebra of complex numbers, with the difference that $\epsilon^2 = 0 \neq -1$.

Using the vector of dual numbers $\mathbf{a} = \mathbf{b} + \mathbf{c}\epsilon$ the algebra can be extended to multivariate analytic functions such that

$$g(\mathbf{a}) = g(\mathbf{b} + \mathbf{c}\epsilon) = g(\mathbf{b}) + J(\mathbf{b})\mathbf{c}\epsilon, \tag{11}$$

where g is a function and $J(\mathbf{b})$ its Jacobian evaluated at $\mathbf{b}$. This implies that AD – via dual numbers – allows to efficiently calculate Jacobian-vector products (JVPs) such as $J(\mathbf{b})\mathbf{c}$ just by a single forward-pass. It does however by no way imply that the Jacobian itself is cheap to obtain. To see this, denote the JVP as $J(\mathbf{b})\mathbf{c} = \Lambda(\mathbf{b}, \mathbf{c})$. Assuming $g : \mathbb{R}^m \to \mathbb{R}^n$, the Jacobian at $\mathbf{b}$ can be calculated using AD by

$$J(\mathbf{y}) = J(\mathbf{y})\mathbf{I}_n = J(\mathbf{y})\begin{bmatrix}\mathbf{e}_1^\intercal & \mathbf{e}_2^\intercal & \cdots & \mathbf{e}_n^\intercal\end{bmatrix} = \begin{bmatrix}\Lambda(\mathbf{y}, \mathbf{e}_1^\intercal) & \Lambda(\mathbf{y}, \mathbf{e}_2^\intercal) & \cdots & \Lambda(\mathbf{y}, \mathbf{e}_n^\intercal)\end{bmatrix}, \tag{12}$$

where $\mathbf{I}_n$ is the n-dimensional identity matrix and $\mathbf{e}_i$ is the ith vector in the standard basis of $\mathbb{R}^n$. This implies that calculating the Jacobian of a function with domain $\mathbb{R}^n$ requires exactly n evaluations of f, which is only one evaluation less than needed for the one-sided finite difference approximation of the Jacobian. Notably for our application, this makes the evaluation of objects such as the Jacobian of f w.r.t. d_t (as e.g. suggested in the appendix of Auclert et al. (2021) to generalize their model specification) prohibitively expensive for many applications.[17] Thus and so far, the only clear advantage of AD over finite difference methods is precision.

In *reverse mode* automatic differentiation, the computational graph of the function is traversed in reverse order, and the derivative of each node is computed by using the derivatives of its outputs. The derivative of each node is then used to update the derivative of its inputs. The process continues until the derivatives of the inputs are computed. Importantly, this allows to cheaply evaluate the

[16] In other words that for a representative agent problem F and F_x are equivalent while for heterogeneous agents, F is a composition of F_x, F_a and F_d.

[17] The finite difference approximation of a JVP is given by $\Lambda(\mathbf{y}, \mathbf{z}) \approx \frac{f(\mathbf{y}+\sigma\mathbf{z}) - f(\mathbf{y})}{\sigma}$ where σ is the step size. The evaluation of a JVP using dual numbers requires one evaluation of f versus two evaluations when using a finite difference approximation. Note that the dual number evaluation of f comes with a computational overhead.

vector-Jacobian product (VJP) $\mathbf{c}^\intercal J(\mathbf{b}) = \Gamma(\mathbf{b}, \mathbf{c})$ at a single pass of the function. Continuing to assume that the codomain of g is $\mathbb{R}^m$, the Jacobian can therefore also be evaluated using AD by

$$J(\mathbf{y}) = \mathbf{I}_m J(\mathbf{y}) = \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \vdots \\ \mathbf{e}_m \end{bmatrix} J(\mathbf{y}) = \begin{bmatrix} \Gamma(\mathbf{y}, \mathbf{e}_1) \\ \Gamma(\mathbf{y}, \mathbf{e}_2) \\ \vdots \\ \Gamma(\mathbf{y}, \mathbf{e}_m) \end{bmatrix}, \tag{13}$$

which requires m evaluations. The use of reverse mode AD over finite differences is hence beneficial either for evaluating VJPs at low costs or for calculating J if $m < n$, whereas forward mode is beneficial if $m > n$. The next two subsections show how to apply these insights efficiently to solve macroeconomic heterogeneous agent models.

2.4. An extended Newton's method based on JVPs

Summarizing the last subsection, the particular strength of AD is to evaluate JVPs, VJPs, and Jacobians of functions with either very small domain or codomain. Unfortunately, the latter is clearly not the case for the function F from Eq. (7) – Newton's method – which comprises a square Jacobian of size $n(T-1) \times n(T-1)$. Calculating a *single* Jacobian for the two-asset HANK model and a truncation horizon of $T = 300$ takes more than 5 min on a standard laptop, thereby rendering the calculation of a single transition path prohibitively costly.[18] Thus, AD alone cannot solve the problem of finding nonlinear transition paths for heterogeneous agent models.

Rather than actually calculating and inverting the Jacobian matrix for Newton's method in Eq. (9), we may seek to solve the system of linear equations

$$J(\mathbf{x}_i)(\mathbf{x}_{i+1} - \mathbf{x}_i) = -F(\mathbf{x}_i), \tag{14}$$

i.e. during each iteration we are looking for an $\mathbf{y} = \mathbf{x}_i - \mathbf{x}_{i+1}$ such that $\Lambda(\mathbf{x}_i, \mathbf{y}) = F(\mathbf{x}_i)$. Denote by $\bar{\mathbf{x}}$ the sequence of steady state variables and by $\bar{J} = J(\bar{\mathbf{x}})$ the steady state sequence space Jacobian. Then

$$J(\mathbf{x}_i)\mathbf{y} = F(\mathbf{x}_i), \tag{15}$$

$$(J(\mathbf{x}_i) - \alpha^{-1}\bar{J} + \alpha^{-1}\bar{J})\mathbf{y} = F(\mathbf{x}_i), \tag{16}$$

$$\alpha^{-1}\bar{J}\mathbf{y} = F(\mathbf{x}_i) - (J(\mathbf{x}_i) - \alpha^{-1}\bar{J})\mathbf{y}, \tag{17}$$

$$\mathbf{y} = \mathbf{y} + \alpha\bar{J}^{-1}(F(\mathbf{x}_i) - J(\mathbf{x}_i)\mathbf{y}), \tag{18}$$

where $\alpha > 0$ is a scalar damping factor discussed further below.

This last equation can be used as the starting point for an iterative procedure. Proposition 1 provides a first convergence result for the neighborhood of the steady state.

Proposition 1 (convergence near the steady state)**.** *Given an initial guess $\mathbf{y}_0$ and fixing $\alpha = 1$, the iterative scheme*

$$\mathbf{y}_{j+1} = \mathbf{y}_j + \alpha\bar{J}^{-1}(F(\mathbf{x}_i) - \Lambda(\mathbf{x}_i, \mathbf{y}_j)), \tag{19}$$

converges to

$$\lim_{j\to\infty} \mathbf{y}_j = J(\mathbf{x}_i)^{-1}F(\mathbf{x}_i) \tag{20}$$

if $\mathbf{x}^*$ is sufficiently close to $\bar{\mathbf{x}}$ and $\bar{J}$ and $J(\mathbf{x}_i)$ are invertible.

Proof. Given invertibility of $\bar{J}$ it is clearly the case that if $\mathbf{y}_{j+1} = \mathbf{y}_j$, then $F(\mathbf{x}_i) = \Lambda(\mathbf{x}_i, \mathbf{y}_j) = J(\mathbf{x}_i)\mathbf{y}_j$. This means we have to prove convergence of $\mathbf{y}_j$ in Eq. (19). It is well known that the iterative procedure

$$\mathbf{y}_{j+1} = \mathbf{c} + A\mathbf{y}_j \tag{21}$$

converges for a square matrix A if the spectral radius $\rho(A)$ of A is less than unity, i.e. if the modulus of all eigenvalues of A lie within the unit circle.

Define $\Sigma_i = J(\mathbf{x}_i) - \bar{J}$ to be the deviation of the Jacobian at iteration i from the steady state Jacobian. For $\alpha = 1$ we have that

$$\mathbf{y}_{j+1} = \mathbf{y}_j + \bar{J}^{-1}(F(\mathbf{x}_i) - \Lambda(\mathbf{x}_i, \mathbf{y}_j)), \tag{22}$$

$$= \bar{J}^{-1}F(\mathbf{x}_i) + \left(\mathbf{I} - \bar{J}^{-1}J(\mathbf{x}_i)\right)\mathbf{y}_j, \tag{23}$$

$$= \bar{J}^{-1}F(\mathbf{x}_i) - \bar{J}^{-1}\Sigma_i\mathbf{y}_j, \tag{24}$$

and convergence thus depends on the spectral radius $\rho(\bar{J}^{-1}\Sigma_i)$ of the second term.

In an ϵ-close neighborhood of the steady state we have, for very small ϵ and any norm $||\cdot||$, $||\mathbf{x}_i - \bar{\mathbf{x}}|| < \epsilon$ and thus $||J(\mathbf{x}_i) - \bar{J}|| = ||\Sigma_i|| < \epsilon$. Since the determinant of a matrix equals the product of its eigenvalues it follows that because $\epsilon \ll 1$, the determinant $\det(\Sigma_i) < \epsilon$ is also very small. Recall that for any square matrices B and C it holds that $\det(BC) = \det(B)\det(C)$ and we thus have

$$\det(\bar{J}^{-1}\Sigma_i) = \det(\bar{J}^{-1})\det(\Sigma_i) < \epsilon \implies \rho(\bar{J}^{-1}\Sigma_i) < 1. \tag{25}$$

□

[18] Achdou et al. (2022) discuss this case in their online appendix but rule it out as being too costly in practice.

The above result is useful because it holds as long as $\rho(\bar{J}^{-1}\Sigma_i) < 1$, which may not only be true in the direct neighborhood of $\bar{x}$. However, since we are in particular interested in those cases where the deviation from the steady state is large, we can go one step further and refine the iterative procedure by adding a damping factor α_j to be determined iteratively. For this, the following Lemma 1 will be useful, which allows us to arrive at the central Proposition 2.

Lemma 1 (properties of the Rayleigh quotient). *The Rayleigh quotient of a real matrix M and vector $\mathbf{z}$ is given by*

$$R(M,\mathbf{z}) = \frac{\mathbf{z}^\intercal M \mathbf{z}}{\mathbf{z}^\intercal \mathbf{z}}. \tag{26}$$

It holds that

i) If $\mathbf{v}$ is an eigenvector of M with associated eigenvalue λ, then

$$R(M,\mathbf{v}) = \frac{\mathbf{v}^\intercal M \mathbf{v}}{\mathbf{v}^\intercal \mathbf{v}} = \frac{\mathbf{v}^\intercal \lambda \mathbf{v}}{\mathbf{v}^\intercal \mathbf{v}} = \lambda. \tag{27}$$

ii) For any iterative procedure $\mathbf{y}_{j+1} = \mathbf{z} + M\mathbf{y}_j$ with square matrix M, $\mathbf{y}_j$ grows (or shrinks) along the eigenvector associated with the eigenvalue of M with largest magnitude (Mises and Pollaczek-Geiringer, 1929). Together with i) this implies

$$\lim_{j\to\infty} R(M,\mathbf{y}_j) = \rho(M). \tag{28}$$

iii) For a square matrix M and a vector $\mathbf{z}$ with $||\mathbf{z}|| > 0$ it holds that

$$|R(M,\mathbf{z})| \in [0, \sigma_{\max}], \tag{29}$$

where $\sigma_{\max}$ is the largest singular value of M.

Proof. It is well known that

$$R(M^\intercal M,\mathbf{z}) \in [\sigma_{\min}^2, \sigma_{\max}^2], \tag{30}$$

with $\sigma_{\min}$ as the respective smallest singular value of M. The result follows immediately if we can show that

$$R(M^\intercal M,\mathbf{z}) \geq R(M,\mathbf{z})^2. \tag{31}$$

Define $\mathbf{w} = M\mathbf{z}$. Then the above is equivalent to

$$\frac{\mathbf{z}^\intercal M^\intercal M \mathbf{z}}{\mathbf{z}^\intercal \mathbf{z}} - \frac{\mathbf{z}^\intercal M \mathbf{z}\mathbf{z}^\intercal M \mathbf{z}}{(\mathbf{z}^\intercal \mathbf{z})^2} = \frac{\mathbf{w}^\intercal \mathbf{w}}{\mathbf{z}^\intercal \mathbf{z}} - \frac{\mathbf{z}^\intercal \mathbf{w}\mathbf{z}^\intercal \mathbf{w}}{(\mathbf{z}^\intercal \mathbf{z})^2} \geq 0, \tag{32}$$

$$\mathbf{w}^\intercal \mathbf{w} - \frac{\mathbf{w}^\intercal \mathbf{z}\mathbf{z}^\intercal \mathbf{w}}{\mathbf{z}^\intercal \mathbf{z}} \geq 0, \tag{33}$$

$$\mathbf{w}^\intercal \left(\mathbf{I} - \frac{\mathbf{z}\mathbf{z}^\intercal}{\mathbf{z}^\intercal \mathbf{z}}\right)\mathbf{w} \geq 0, \tag{34}$$

which uses the fact that $R(M,\mathbf{z}) = R(M^\intercal,\mathbf{z})$. It is further that

$$\mathbf{w}^\intercal \mathbf{w} - \frac{\mathbf{w}^\intercal \mathbf{z}\mathbf{z}^\intercal \mathbf{w}}{\mathbf{z}^\intercal \mathbf{z}} = \mathbf{w}^\intercal \mathbf{w}\left(1 - \frac{(\mathbf{w}^\intercal \mathbf{z})^2}{(\mathbf{z}^\intercal \mathbf{z})(\mathbf{w}^\intercal \mathbf{w})}\right). \tag{35}$$

Since from the Cauchy-Schwarz inequality we have

$$\left(\mathbf{w}^\intercal \mathbf{z}\right)^2 \leq (\mathbf{z}^\intercal \mathbf{z})(\mathbf{w}^\intercal \mathbf{w}), \tag{36}$$

and thus

$$\frac{(\mathbf{w}^\intercal \mathbf{z})^2}{\mathbf{z}^\intercal \mathbf{z}\mathbf{w}^\intercal \mathbf{w}} < 1, \tag{37}$$

it follows that

$$\mathbf{w}^\intercal \left(\mathbf{I} - \frac{\mathbf{z}\mathbf{z}^\intercal}{\mathbf{z}^\intercal \mathbf{z}}\right)\mathbf{w} \geq 0. \tag{38}$$

□

Proposition 2 (convergence in general). *Given an initial guess $\mathbf{y}_0$, a scaling parameter $\gamma \in (1,2)$ and initializing $\alpha_0 = 1$, the iterative scheme*

$$\mathbf{y}_{j+1} = \mathbf{y}_j + \alpha_j \bar{J}^{-1}(F(\mathbf{x}_i) - \Lambda(\mathbf{x}_i,\mathbf{y}_j)), \tag{39}$$

$$\alpha_j = \min\left\{\alpha_{j-1}, \gamma \middle/ \left|R\left(\bar{J}^{-1}J(\mathbf{x}_i),\mathbf{y}_j\right)\right|\right\}, \tag{40}$$

with the Rayleigh quotient $R(\cdot) = \frac{\mathbf{y}_j^\intercal \bar{J}^{-1}\Lambda(\mathbf{x}_i,\mathbf{y}_j)}{\mathbf{y}_j^\intercal \mathbf{y}_j}$, converges to

$$\lim_{j\to\infty} \mathbf{y}_j = J(\mathbf{x}_i)^{-1}F(\mathbf{x}_i) \tag{41}$$

if all generalized eigenvalues of $\bar{J}$ and $J(\mathbf{x}_i)$ are real, positive and finite.

Proof. Positivity and finiteness of the generalized eigenvalues imply that $\bar{J}$ and $J(\mathbf{x}_i)$ are nonsingular. We again have to prove convergence of $\mathbf{y}_j$ in Eq. (39) for which $\rho(A_j) < 1$ given $A_j = \mathbf{I} - \alpha_j B$ with $B = \bar{J}^{-1} J(\mathbf{x}_i)$ is a sufficient condition. Denote by $\lambda_k(A_j)$ the (unordered) kth eigenvalue of A_j and define $\lambda_k(B)$ respectively. It holds that

$$\lambda_k(A_j) = 1 - \alpha_j \lambda_k(B), \tag{42}$$

$$|\lambda_k(A_j)| = \sqrt{(1 - \alpha_j \Re(\lambda_k(B)))^2 + (\alpha_j \Im(\lambda_k(B)))^2}, \tag{43}$$

$$= \sqrt{1 - 2\alpha_j \Re(\lambda_k(B)) + \alpha_j^2 |\lambda_k(B)|^2}. \tag{44}$$

Imposing $|\lambda_k(A_j)| < 1$ for all k requires

$$\alpha_j < \frac{2\Re(\lambda_k(B))}{|\lambda_k(B)|^2} \quad \forall k = 1, 2, \dots, n(T-1), \tag{45}$$

which is an upper bound on α_j. Under the given assumption that all eigenvalues of B are real and positive, this reduces to

$$\alpha_j < \frac{2}{\rho(B)}, \tag{46}$$

where the spectral radius $\rho(B)$ of B is unknown. Eq. (40) defines the recursion

$$\alpha_j = \min\left\{\alpha_{j-1}, \gamma / \left|R(B, \mathbf{y}_j)\right|\right\}, \tag{47}$$

which uses the Rayleigh quotient

$$R(B, \mathbf{y}_j) = \frac{\mathbf{y}_j^\intercal B \mathbf{y}_j}{\mathbf{y}_j^\intercal \mathbf{y}_j} = \frac{\mathbf{y}_j^\intercal \bar{J}^{-1} \Lambda(\mathbf{x}_i, \mathbf{y}_j)}{\mathbf{y}_j^\intercal \mathbf{y}_j}. \tag{48}$$

from Lemma 1 to also iteratively approximate $\rho(B)$ with each iteration on $\mathbf{y}_j$.

The remaining task is to show the convergence of α_j. Since from Lemma 1 we know that $|R(B, \mathbf{y}_j)| \leq \sigma_{\max}(B)$ it follows that α_j is bounded by

$$\alpha_j \in \left[\frac{\gamma}{\sigma_{\max}(B)}, 1\right], \tag{49}$$

and in the limit it holds

$$\lim_{j\to\infty} \alpha_j \in \left[\frac{\gamma}{\sigma_{\max}(B)}, \min\left\{1, \frac{\gamma}{\rho(B)}\right\}\right], \tag{50}$$

that is, α_j and thereby $A_j = \mathbf{I} - \alpha_j B$ converges with given bounds. As the eigenvalues of B are positive and real, we have that

$$\lim_{j\to\infty} \rho(A_j) = \max\left\{\lim_{j\to\infty}\left(1 - \alpha_j \lambda_{\min}(B)\right), \lim_{j\to\infty}\left(\alpha_j \rho(B) - 1\right)\right\} \tag{51}$$

with

$$\lim_{j\to\infty}\left(1 - \alpha_j \lambda_{\min}(B)\right) \leq 1 - \gamma \frac{\lambda_{\min}(B)}{\sigma_{\max}(B)} < 1 \tag{52}$$

and

$$\lim_{j\to\infty}\left(\alpha_j \rho(B) - 1\right) \in \left\{\begin{array}{ll} \left[\gamma \frac{\rho(B)}{\sigma_{\max}(B)} - 1, \gamma - 1\right] & \text{if } \rho(B) \geq \gamma \\ \left[\gamma \frac{\rho(B)}{\sigma_{\max}(B)} - 1, \rho(B) - 1\right] & \text{if } \rho(B) < \gamma \end{array}\right\} < 1, \tag{53}$$

where in the second line $\rho(B) < \gamma$ implies $\rho(B) - 1 < \gamma - 1 < 1$. Thus, $\lim_{j\to\infty} \rho(A_j) < 1$ and $\mathbf{y}_j$ converges to a solution to the linear system of equations in (14). □

Importantly, Proposition 2 eliminates the requirement that $\mathbf{x}^*$ must be sufficiently close to $\bar{\mathbf{x}}$ from Proposition 1 and yields a very general tool to solve the class of problems at hand. To achieve this, note that the proposition contains *two* simultaneous iterative procedures: the procedure of solving for the best linear improvement $\mathbf{x}_i$ to the solution of F associated with each Newton step, and, again for each Newton step, the procedure of finding a feasible damping factor α to guarantee convergence of the former. Stated differently, the second procedure iterates on the preconditioning matrix instead of the solution. Both procedures operate at the same time, which is unproblematic since the underlying problem *within* each Newton step is linear. Intuitively, for each new Newton step we set α_j to equal the inverse of the approximation of the spectral radius of B (scaled by γ) and thereby dampen the spectral radius of A to lie inside the unit circle. Under the given assumptions, this guarantees convergence to the solution of the associated system of linear equations.

Numerically, the beauty in Proposition 2 lies in the fact that both, $\mathbf{y}_j$ and α_j can be calculated by just one AD forward-pass on F. Further, $F(\mathbf{x}_i) - \Lambda(\mathbf{x}_i, \mathbf{y}_j)$ is a vector and $\bar{J}^{-1}(F(\mathbf{x}_i) - \Lambda(\mathbf{x}_i, \mathbf{y}_j)$ can hence be evaluated by the LU-factorization of $\bar{J}$, which only needs to be calculated once for a given model. All other operations are trivial vector space calculations.

The two conditions for Proposition 2 – positivity and realness of the generalized eigenvalues – are sufficient, but not necessary conditions. Importantly, this implies that the iterative procedure may converge even if these conditions are not satisfied. The condition that the generalized eigenvalues are real is a rather weak condition, which is related to the fact that the Rayleigh approximation cannot distinguish between the real and imaginary part of the largest eigenvalue. If the eigenvalues are complex, Eq. (45) is still likely to be satisfied whenever $\Re(\lambda_k(B)) > 0$ because, in practice, $\alpha_j \leq \frac{\gamma}{\rho(B)}$ is only an upper bound (c.f. Eq. (49)).[19]

The value of $\gamma \in (1, 2)$ determines the speed of convergence and can be chosen freely within the given bounds: a value close to 2 will shift the largest eigenvalue of B close to 2, resulting in a spectral radius $\rho(A)$ of A close to one. A value of α_j close to 0 will shift the *smallest* eigenvalue of B close to 0 (c.f. Eq. (51)), which also results in $\rho(A)$ close to one. Since convergence speed depends on the magnitude of $\rho(A)$ (smaller in magnitude is better) an optimal value will lie somewhere in the mid-range. While a sufficiently small γ could compensate for the rare cases in which $\frac{2\Re(\lambda_k(B))}{|\lambda_k(B)|^2} \ll \frac{2}{\rho(B)}$ for complex eigenvalues, a value of $\gamma = 1.5$ has in practice proven very reliable for a wide range of applications.

2.5. Finding the steady state and its Jacobian

Continue to denote steady state objects by a bar, i.e. $\bar{x}$, $\bar{d}$, $\bar{a}$ and $\bar{\iota w}$. In the common cases when the steady state is not unique some variables must be fixed ex-ante via an additional set of restrictions $x = b(\mathfrak{x})$, where $\mathfrak{x}$ is the subset of x necessary to evaluate $\bar{x}$.[20] The steady state must satisfy

$$\bar{x} = b(\bar{\mathfrak{x}}), \tag{55}$$

$$(\bar{a}, \bar{\iota w}) = W(\bar{\iota w}, \bar{x}, \bar{x}, \bar{x}), \tag{56}$$

$$\bar{d} = D(\bar{a}, \bar{d}), \tag{57}$$

$$\mathbf{0} = f(\bar{x}, \bar{x}, \bar{x}, \bar{d}, \bar{a}). \tag{58}$$

For a given guess on $\mathfrak{x}$ calculate the corresponding guess on $\bar{x}$ using $b(\cdot)$. $\bar{\iota w}$ can then be found by iterating on Eq. (56) until convergence. Denote the function that does so as $\bar{\iota w} = \bar{W}(\bar{x})$. Given $\bar{x}$ and $\bar{\iota w}$, the steady-state distribution $\bar{d}$ can as well be found by iterating on Eq. (57) until convergence (or, alternatively, via the unit-eigenvector). Denote this solver as $\bar{d} = \bar{D}(\bar{a})$ and define $\bar{f}$ equivalently for $f(\cdot)$. Combining those three, $\bar{\mathfrak{x}}$ must satisfy $H(\bar{\mathfrak{x}}) = \mathbf{0}$ with H defined as

$$H(\bar{\mathfrak{x}}) = \bar{f}\left(b(\bar{\mathfrak{x}}), \bar{D}\left(b(\bar{\mathfrak{x}}), \bar{W}(b(\bar{\mathfrak{x}}))\right), \bar{W}(b(\bar{\mathfrak{x}}))\right). \tag{59}$$

Since $\mathfrak{x} \in \mathbb{R}^m$ with m small, the complete Jacobian of H can be calculated efficiently using forward mode automatic differentiation and, starting with some guess $\mathfrak{x}_i$ on $\bar{\mathfrak{x}}$, the root of H can be found using a modified Newton's method

$$\mathfrak{x}_{i+1} = \mathfrak{x}_i - J_H(\mathfrak{x}_i)^+ H(\mathfrak{x}_i), \tag{60}$$

where $J_H(\mathfrak{x}_i)^+$ denotes the Moore-Penrose inverse. Using the latter is necessary because $J_H(\mathfrak{x}_i)$ typically does not have full rank since the codomain of f is $\mathbb{R}^n$ and $n \leq m$. Proofs of convergence for the modified Newton procedure are, e.g., given by Ben-Israel (1965). Despite not requiring any manual input, the procedure turns out to be very robust even for relatively bad initial guesses. In particular, it is usually not necessary to manually provide some of the steady state relationships as an additional input to the dynamic system of the economic model.

Given $\bar{x}$, the steady state Jacobian $\bar{J} = J(\bar{\mathbf{x}}) = J(\{\bar{x}\}_0^T)$ of F can also be found using AD. To be clear on notation, let $J_{x\to y}$ be the matrix whose (i, j)th entry is $J_{ij} = \frac{\partial z_i}{\partial x_j}$. The (i, j)th entry of the steady state Jacobian is then for $i > 1$ given by

$$\bar{J}_{ij} = \frac{\partial z_i}{\partial x_j} = \frac{\partial f_i}{\partial x_j} + \frac{\partial f_i}{\partial a_i}\frac{\partial a_i}{\partial x_j} + \frac{\partial f_i}{\partial d_i}\frac{\partial d_i}{\partial x_j} = \frac{\partial f_i}{\partial x_j} + \frac{\partial f_i}{\partial a_i}\frac{\partial a_i}{\partial x_j} + \frac{\partial f_i}{\partial d_i}\left(\frac{\partial d_i}{\partial a_i}\frac{\partial a_i}{\partial x_j} + \frac{\partial d_i}{\partial d_{i-1}}\frac{\partial d_{i-1}}{\partial x_j}\right) \tag{61}$$

$$= \frac{\partial f_i}{\partial x_j} + \frac{\partial f_i}{\partial a_i}\frac{\partial a_i}{\partial x_j} + \frac{\partial f_i}{\partial d_i}\sum_{k=0}^{i}\frac{\partial d_i}{\partial d_k}\frac{\partial d_k}{\partial a_k}\frac{\partial a_k}{\partial x_j} = \frac{\partial f_i}{\partial x_j} + \sum_{k=0}^{i}\frac{\partial z_i}{\partial a_k}\frac{\partial a_k}{\partial x_j}, \tag{62}$$

which is a n-by-n matrix.

[19] A path to ensure that both conditions are guaranteed to be satisfied is to pre-multiply $F(\mathbf{x}_i)$ and $\Lambda(\mathbf{x}_i, \mathbf{y}_j)$ in Eqs. (39) and (40) by $J(\mathbf{x}_i)^\intercal(\bar{J}^{-1})^\intercal\bar{J}^{-1}$ instead of $\bar{J}^{-1}$. These expressions can efficiently be calculated using AD via

$$J(\mathbf{x}_i)^\intercal(\bar{J}^{-1})^\intercal\bar{J}^{-1}F(\mathbf{x}_i) = \Gamma\left(\mathbf{x}_i, \left((\bar{J}^{-1})^\intercal\bar{J}^{-1}F(\mathbf{x}_i)\right)^\intercal\right)^\intercal \tag{54}$$

and ensures that the central matrix in the iterative procedure is positive definite. However, convergence would be relatively slow because the largest and smallest eigenvalues of $J(\mathbf{x}_i)^\intercal(\bar{J}^{-1})^\intercal\bar{J}^{-1}J(\mathbf{x}_i)$ are $\sigma_{\min}(B)^2$ and $\sigma_{\max}(B)^2$ which implies eigenvalues of the iterative procedure with modulus considerably closer to unity.

[20] A typical example for New-Keynesian models is that the steady state inflation needs to be fixed ex-ante since it is a policy choice variable of the central bank.

Recall that the function $F_a(\cdot)$ from Eq. (4) is defined on the complete sequence $\mathbf{x}$. We can calculate the Jacobian with respect to x_{T-1} by stacking the JVPs of the transpose of the last n vectors of the standard basis of $\mathbb{R}^{(T-1)n}$:

$$\bar{J}_{x_{t-1}\to\mathbf{a}} = \begin{bmatrix} \Lambda_{F_a}\left(\bar{\mathbf{x}}, \mathbf{e}^\intercal_{(T-1)}\right)^\intercal \\ \Lambda_{F_a}\left(\bar{\mathbf{x}}, \mathbf{e}^\intercal_{(T-1)2}\right)^\intercal \\ \vdots \\ \Lambda_{F_a}\left(\bar{\mathbf{x}}, \mathbf{e}^\intercal_{(T-1)(n-1)}\right)^\intercal \\ \Lambda_{F_a}\left(\bar{\mathbf{x}}, \mathbf{e}^\intercal_{(T-1)n}\right)^\intercal \end{bmatrix}^\intercal = \begin{bmatrix} \frac{\partial a_0}{\partial x_{T-1}} \\ \frac{\partial a_1}{\partial x_{T-1}} \\ \vdots \\ \frac{\partial a_{T-2}}{\partial x_{T-1}} \\ \frac{\partial a_{T-1}}{\partial x_{T-1}} \end{bmatrix} = \begin{bmatrix} \frac{\partial a_0}{\partial x_{T-1}} \\ \frac{\partial a_0}{\partial x_{T-2}} \\ \vdots \\ \frac{\partial a_0}{\partial x_1} \\ \frac{\partial a_0}{\partial x_0} \end{bmatrix} = \bar{J}_{\mathbf{x}\to a_0}, \tag{63}$$

where the equivalence in the second step holds because in the steady state we have $\frac{\partial a_0}{\partial x_k} = \frac{\partial a_l}{\partial x_{k+l}}$ for any l. Note that the calculation of this object requires only n evaluations of $F_a(\cdot)$.

Similarly, the Jacobian $\bar{J}_{\mathbf{a}\to z_{T-1}}$ of z_{T-1} (i.e. the last output element of F_x) w.r.t. the sequence $\mathbf{a}$ can be evaluated by using reverse mode automatic differentiation:

$$\bar{J}_{\mathbf{a}\to z_{T-1}} = \begin{bmatrix} \Gamma_{F_x\circ F_d}\left(\bar{\mathbf{a}}, \mathbf{e}_{(T-1)}\right) \\ \Gamma_{F_x\circ F_d}\left(\bar{\mathbf{a}}, \mathbf{e}_{(T-1)2}\right) \\ \vdots \\ \Gamma_{F_x\circ F_d}\left(\bar{\mathbf{a}}, \mathbf{e}_{(T-1)(n-1)}\right) \\ \Gamma_{F_x\circ F_d}\left(\bar{\mathbf{a}}, \mathbf{e}_{(T-1)n}\right) \end{bmatrix} = \begin{bmatrix} \frac{\partial z_{T-1}}{\partial a_0} \\ \frac{\partial z_{T-1}}{\partial a_1} \\ \vdots \\ \frac{\partial z_{T-1}}{\partial a_{T-2}} \\ \frac{\partial z_{T-1}}{\partial a_{T-1}} \end{bmatrix}^\intercal = \begin{bmatrix} \frac{\partial z_{T-1}}{\partial a_0} \\ \frac{\partial z_{T-2}}{\partial a_0} \\ \vdots \\ \frac{\partial z_1}{\partial a_0} \\ \frac{\partial z_0}{\partial a_0} \end{bmatrix}^\intercal = \bar{J}_{a_0\to\mathbf{z}}, \tag{64}$$

which, as above, uses the fact that $\frac{\partial z_k}{\partial a_0} = \frac{\partial z_{k+l}}{\partial a_l}$ in steady state. Note again that independently of the complexity of the functions F_x and F_d this requires only n evaluations of $F_x \circ F_d$.

Finally, initialize a helper matrix $\hat{J}$ with the tensor (outer) product $J(\bar{\mathbf{x}})_{\mathbf{a}\to z_{T-1}} \otimes J(\bar{\mathbf{x}})_{x_{t-1}\to\mathbf{a}}$ and add $\frac{\partial f}{\partial x_t}$ to $\hat{J}_{T-1,T-1}$, $\frac{\partial f}{\partial x_{t+1}}$ to $\hat{J}_{T-1,T-2}$ and $\frac{\partial f}{\partial x_{t-1}}$ to $\hat{J}_{T-2,T-1}$. Then

$$\hat{J} = \begin{bmatrix} \frac{\partial z_{T-1}}{\partial a_0}\frac{\partial a_0}{\partial x_{T-1}} & \cdots & \frac{\partial z_{T-1}}{\partial a_0}\frac{\partial a_0}{\partial x_1} & \frac{\partial z_{T-1}}{\partial a_0}\frac{\partial a_0}{\partial x_0} \\ \vdots & \ddots & \vdots & \vdots \\ \frac{\partial z_1}{\partial a_0}\frac{\partial a_0}{\partial x_{T-1}} & \cdots & \frac{\partial z_1}{\partial a_0}\frac{\partial a_0}{\partial x_1} & \frac{\partial f}{\partial x_{t-1}} + \frac{\partial z_1}{\partial a_0}\frac{\partial a_0}{\partial x_0} \\ \frac{\partial z_0}{\partial a_0}\frac{\partial a_0}{\partial x_{T-1}} & \cdots & \frac{\partial f}{\partial x_{t+1}} + \frac{\partial z_0}{\partial a_0}\frac{\partial a_0}{\partial x_1} & \frac{\partial f}{\partial x_t} + \frac{\partial z_0}{\partial a_0}\frac{\partial a_0}{\partial x_0} \end{bmatrix} = \begin{bmatrix} \frac{\partial z_{T-1}}{\partial x_{T-1}} & \cdots & \frac{\partial z_{T-1}}{\partial x_0} & \\ \vdots & \ddots & \vdots & \vdots \\ \frac{\partial z_1}{\partial x_{T-1}} & \cdots & \frac{\partial f}{\partial x_t} + \frac{\partial z_0}{\partial x_0} & \end{bmatrix}, \tag{65}$$

and $\bar{J}$ can be expressed as the recursion over its block components

$$\bar{J}_{ij} = \bar{J}_{i-1,j-1} + \hat{J}_{T-i,T-j} = \sum_{k=0}^{\min\{i,j\}} \hat{J}_{T-i+k,T-j+k}, \tag{66}$$

which corresponds to the expression in Eq. (61) and therefore yields $\bar{J}$ for only n evaluations of F_a, F_d and F_x each. Since for most applications $\bar{J}$ is sparse, a sparse implementation of the incomplete LU decomposition can be used to pre-calculate $\bar{J}^{-1}$. For any model, this only needs to be done once.

3. A class of medium scale heterogeneous agent models

This paper focuses on a generic class of heterogeneous agents models, which nests many prominent models from the literature. They differ in the degree of heterogeneity of households and the complexity of the aggregate economy and include the HANK model with one asset, the HANK model where households have access to a liquid and an illiquid asset (two-asset HANK), the medium-scale model with a single representative household (RANK), and even the three-equation canonical New Keynesian model. For the sake of brevity I here only sketch the setup of the two-asset HANK model and those aggregate equations that are non-standard. The more standard aggregate relationships and the definition of the dynamic equilibrium are redirected to Appendix A together with the RANK model and the one-asset-HANK model.

The benchmark two-asset HANK model combines the two-asset HANK model (e.g. Auclert et al., 2021) with the standard medium-scale DSGE model in the tradition of Smets and Wouters (2007).[21] This results in a very rich dynamic model with many aggregate state variables that demonstrates the potential of the method.[22] The model hence contains a disaggregated part with the households decisions and the distribution dynamics, and the aggregated part of a medium-scale DSGE model. Households are heterogeneous with respect to three dimensions: their skills and their liquid and illiquid asset holdings. The aggregate part of the model features, next to a host of state and jump variables, eight aggregate shocks. The modular setup of the method presented in Sections 2 (and Section 5, for the implementation) allows to specify both parts separately.

[21] This goes beyond the work of Bayer et al. (2020) as I additionally add various forms of inertia that are present in Smets and Wouters (2007), such as, e.g., inflation and wage indexation.

[22] Arguably, one would only employ a model with such a degree of feature-richness when desiring to bring it to the data using Bayesian methods. For the analysis of policy interventions, such as presented in Section 4, a smaller model is preferable.

3.1. Recursive valuations $W(\cdot)$

In the disaggregated part of the model, households can hold liquid bonds b_{it} and illiquid assets a_{it}, the latter pay higher returns but are subject to convex portfolio adjustment costs. Households face idiosyncratic labor income risk e_{it} and a borrowing constraint on both assets. They wish to accumulate net worth for the purpose of consumption smoothing and to insure against the associated idiosyncratic income risk. Their Bellman equation is given by

$$V_t(e_{it}, b_{i,t-1}, a_{i,t-1}) = \max_{c_{it}, b_{it}, a_{it}} \left\{ \frac{c_{it}^{1-\sigma_c}}{1-\sigma_c} - \chi \frac{n_t^{1+\sigma_l}}{1+\sigma_l} + \beta_t E_t V_{t+1}(e_{i,t+1}, b_{it}, a_{it}) \right\} \tag{67}$$

such that

$$c_{it} + a_{it} + b_{it} = (1-\tau_t)\mathfrak{w}_t e_{it} n_t + (1+r_t^a)a_{i,t-1} + (1+r_t^b)b_{i,t-1} - \Phi_t(a_{it}, a_{i,t-1}) + T_t, \tag{68}$$

$$a_{it} \geq 0, \tag{69}$$

$$b_{it} \geq \bar{b}, \tag{70}$$

where β_t is the discount factor, c_{it} is the consumption of household i, n_t denotes the labor supply set by labor unions, and $\mathfrak{w}_t$ is the economy-wide wage, while r_t^a and r_t^b are the rates of return on the two types of assets. τ_t the level of labor taxation and T_t is an exogenous government lump-sum transfer specified further below. e_{it}, the household-specific labor productivity, follows an AR(1) process in logs,

$$\log e_{it} = \rho_e \log e_{i,t-1} + \epsilon_{it}^e. \tag{71}$$

$\Phi_t(\cdot)$ is the function specifying portfolio adjustment costs for the illiquid asset

$$\Phi_t(a_{it}, a_{i,t-1}) = \frac{\chi_1}{\chi_2} \left| \frac{a_{it} - (1+r_t^a)a_{i,t-1}}{(1+r_t^a)a_{i,t-1} + \chi_0} \right|^{\chi_2} [(1+r_t^a)a_{i,t-1} + \chi_0], \tag{72}$$

with $\chi_0, \chi_1 > 0$ and $\chi_2 > 1$.

In terms of the general representation of heterogeneous agent models presented in Section 2, Eqs. (67) to (72) represent the function $W(\cdot)$. $\mathbb{S} \subset \mathbb{R}^3$ is the space spanning over the domains of the two types of assets a_{it} and b_{it} and the domain of the household-specific productivity level e_{it}. The decisions a_t are the choices of c_{it}, a_{it} and b_{it}, whereas w_t corresponds to the marginal values of these choices for each node on the grid S_g. The recursive problem takes wages $\mathfrak{w}_t$, labor n_t, taxes τ_t, transfers T_t and the return rates r_t^a and r_t^b as given, which are all part of the set of aggregate variables x_t.

This setup reduces to a one-asset HANK model after setting χ_0 to zero, which eliminates all portfolio adjustment costs, making assets a_{it} and b_{it} perfect substitutes (see Eqs. (A.36) to (A.36) in Appendix A). For the aggregate economy this would imply that no arbitrage requires $r_t^a = r_t^b$. The RANK model is nested when removing the borrowing constraints given by Eqs. (69) and (70). In this case agents can perfectly insure against idiosyncratic income risk and would thus make exactly the same decision. Their consumption-savings decision is then represented by a conventional Euler equation (c.f. Eqs. (A.1) and (A.2) in Appendix A).

3.2. The aggregate economy $f(\cdot)$

The aggregate economy features all the bells and whistles of the medium-scale workhorse model of Smets and Wouters (2007). The setup of firms includes capital accumulation, investment adjustment costs, capital utilisation costs and price indexation. Likewise, the wages are determined by labor unions which feature wage stickiness and wage indexation. To avoid linear approximations, the design of the price and wage Phillips curves follows a (Rotemberg, 1982) setup as in Gust et al. (2012) rather than the (Calvo, 1983) price setting of Smets and Wouters (2007). The conventional parts of the model, including in particular the firm side and exogenous shocks, are presented in detail in Appendix A.1.

Households sell labor services to a labor union, which in turn supplies labor input n_t in a perfectly competitive labor market. Wage adjustment costs then give rise to a wage Phillips curve,

$$\psi_{\mathfrak{w}} \left(\frac{\pi_t^{\mathfrak{w}}}{\tilde{\pi}_t^{\mathfrak{w}}} - 1 \right) \frac{\pi_t^{\mathfrak{w}}}{\tilde{\pi}_t^{\mathfrak{w}}} = \psi_{\mathfrak{w}} \beta_t E_t \left\{ \left(\frac{\pi_{t+1}^{\mathfrak{w}}}{\tilde{\pi}_{t+1}^{\mathfrak{w}}} - 1 \right) \frac{\pi_{t+1}^{\mathfrak{w}}}{\tilde{\pi}_{t+1}^{\mathfrak{w}}} \right\} + \frac{\mu_t^{\mathfrak{w}}}{\mu_t^{w} - 1} \left(\chi n_t^{1+\sigma_l} - \frac{(1-\tau_t)\mathfrak{w}_t n_t}{\mu_t^{\mathfrak{w}}} \int e_{it} c_{it}^{-\sigma} di \right), \tag{73}$$

where $\mu_t^{\mathfrak{w}}$ is the exogenous wage markup, $\psi_{\mathfrak{w}}$ the parameter governing wage adjustment costs, and χ as well as σ_l are preference parameters. $\pi_t^{\mathfrak{w}} = \frac{\mathfrak{w}_t^n}{\mathfrak{w}_{t-1}^n} \pi_t$ denotes wage inflation and wage indexation $\tilde{\pi}_t^{\mathfrak{w}}$ is given by

$$\ln \tilde{\pi}_t^{\mathfrak{w}} = \omega_{\mathfrak{w}} \ln \bar{\pi}^{\mathfrak{w}} + (1-\omega_{\mathfrak{w}}) \ln \pi_{t-1}^{\mathfrak{w}}, \tag{74}$$

with variables with bars, such as $\bar{\pi}^{\mathfrak{w}}$, denoting steady state values.

Note that wage indexation can be eliminated by setting $\omega_{\mathfrak{w}} = 1$ while wage stickiness can be removed completely by setting wage adjustment costs $\psi_{\mathfrak{w}} = 0$. Further, wages can be subject to downward nominal wage rigidity governed by rigidity parameter $\iota_{\mathfrak{w}}$,

$$\mathfrak{w}_t = \max \left\{ \iota_{\mathfrak{w}} \frac{\mathfrak{w}_{t-1}}{\pi_t}, \mathfrak{w}_t^n \right\}, \tag{75}$$

which can be removed by setting $\iota_{\mathfrak{w}} = 0$.

Dividends are given by

$$\Pi_t = y_t - \mathfrak{w}_t n_t - i_t - \frac{\psi}{2}\left(\frac{\pi_t}{\bar{\pi}_t} - 1\right)^2 y_t, \tag{76}$$

where y_t is aggregate output and i_t investment in the capital stock. The central bank sets the policy rate R_t following a conventional monetary policy rule with interest rate inertia,

$$\ln R_t^n = \rho \ln R_{t-1}^n + (1-\rho)\left(\ln R_t^* + \phi_\pi\left[\ln \pi_t - \ln \bar{\pi}\right] + \phi_y\left[\ln y_t - \ln \bar{y}\right]\right) + \ln \upsilon_t, \tag{77}$$

that may be subject to the zero lower bound on nominal interest rates (ZLB) as in

$$R_t = \max\left\{1,\ R_t^n\right\}. \tag{78}$$

I assume that the government is running a balanced budget, which implies

$$\tau_t \mathfrak{w}_t n_t = \left(\frac{R_{t-1}}{\pi_t} - 1\right) b^g + g_t + T_t, \tag{79}$$

where b^g is the amount of outstanding government debt (assumed to be constant), and government transfers T_t are an exogenous policy decision which is assumed to follow an AR(1) process in logs

$$\ln T_t = (1-\rho_T)\ln \bar{T} + \rho_T T_{t-1} + \varepsilon_t^T. \tag{80}$$

Importantly, the government taxes labor only and adjusts the labor tax rate from period to period to run a balanced budget. The amount of equity s_t is determined by the no arbitrage condition

$$\frac{R_t}{E_t \pi_{t+1}} = \frac{E_t\left\{\Pi_{t+1} + s_{t+1}\right\}}{s_t}, \tag{81}$$

and market clearing requires

$$A_t = \int a_{it} di, \tag{82}$$

$$B_t = \int b_{it} di, \tag{83}$$

$$C_t = \int c_{it} di, \tag{84}$$

$$y_t = C_t + g_t + i_t + \psi_t + \zeta B_t, \tag{85}$$

where ζ is the liquidity premium, and where the aggregate asset holdings A_t and B_t together must contain all outstanding equity and government debt,

$$A_t + B_t = s_t + b^g. \tag{86}$$

Ex-post returns on liquid assets are subject to surprise inflation and the liquidity premium,

$$R_t^b = \frac{R_{t-1}}{\pi_t} - \zeta, \tag{87}$$

and returns to illiquid assets are given by

$$R_t^a = \frac{\Pi_t + s_t}{A_{t-1}} + \frac{A_{t-1} - s_{t-1}}{A_{t-1}} \frac{R_{t-1}}{\pi_t}. \tag{88}$$

The calibration – for the aggregate and disaggregate part of the model – is quite conventional and anchored around the values reported in Boehl (2022), where a similar model is estimated under inclusion of the households' preference parameters. To allow for transition dynamics from one steady state to another, only parameters are fixed ex-ante but no steady state values. The inflation target is set to 2 % annually and the initial level of transfers is zero. Appendix A contains further details on the calibration as well as the full set of equilibrium conditions, which constitute the aggregate equations in $f(\cdot)$, and the list of aggregate variables contained in x_t.

4. Nonlinear transition dynamics

A central feature of models with heterogeneous households is that they allow to study the dynamic effects of government transfers, taxes, and redistribution over the cross-sectional distribution of households and on business cycle aggregates. This section applies the EP method to examine the nonlinear transition dynamics of – permanent or transitory – changes in government redistribution policy. I first discuss the dynamic responses of a change in steady state transfers. I then study the effects of two severe nonlinearities on the transition dynamics, the first being downward nominal wage rigidity and the second being the zero lower bound on nominal interest rates. Finally, I analyze the role of the distribution of wealth for the transmission of transfer shocks.

To provide a serious challenge to the EP method, simulations in this section are based on the two-asset HANK model from Section 3 including all the aggregate bells and whistles introduced there. However, while the macroeconomic effects of redistribution are a fascinating study subject, the focus of this paper remains the presentation and illustration of the EP method. This section can thus not yield as an exhaustive and terminal discussion of the subject of redistribution vs. efficiency.

Table 1
Difference of the steady states with redistribution relative to the old steady state without redistribution.

relative change		relative change	
output	−11.1 %	top-10 % share assets	+8.1 %
wages	−1.8 %	top-10 % share bonds	+11.8 %
labor hours	−9.4 %	top-10 % share consumption	−3.7 %
capital	−17.5 %	assets	−21.6 %
consumption	−11.6 %	bonds	−43.5 %
interest rate	+0.27 %	equity	−26.8 %
dividends	−7.6 %		
	tax rate	+20.2 %	

4.1. A permanent increase in government transfer funded by a proportional labor tax

Assume a permanent increase in government transfers from zero to 10 % of (old) GDP. After announcement, the government gradually increases the volume of transfers with an autocorrelation of $\rho_T = 0.8$. Since the government is running a balanced budget, the labor tax rate must adjust simultaneously and the policy thus redistributes income of high labor income earners to those with low labor income. Importantly, this permanent shift in transfers and taxes implies the transition from one steady state to another.

Before looking at the dynamics it is useful to compare the two steady states in detail. Fig. 1 shows the stationary distribution of assets in the steady state without redistribution. The distribution is bimodal: the majority of agents holds a fair amount of assets and bonds, where agents with many bonds tend to hold fewer assets and vice versa. Roughly one-third (34.3 %) of all agents do not hold any bonds but yet a considerable volume of assets. These are the famous wealthy hand-to-mouth households of Kaplan et al. (2018). These households have experienced a series of negative income shocks and thus depleted their stock of liquid bonds. Since they only hold illiquid bonds and liquidations of these bonds are subject to portfolio adjustment cost, they face limited insurance and have a higher marginal propensity to consume out of income.

Table 1 presents the redistributive steady state relative to the steady state without redistribution and highlights a strong trade-off between redistribution and efficiency. The results suggest that an additional 20 % labor tax rate is necessary to finance the new permanent transfers. Since higher labor taxes reduce after-tax wages, they weaken incentives to work, leading to a nearly 10 % decline in labor supply. With lower labor input, firms find it optimal to reduce their capital stock, which falls by almost 20 %. This contraction in both labor and capital leads to a more than 10 % decline in total output, reducing firm profits and resulting in lower dividends and devaluations in equity.

The new system of government transfers provides income insurance, reducing the need for households to self-insure by holding liquid assets. As a result, the demand for liquid bonds falls by nearly 50 %. Since poorer households face lower precautionary savings

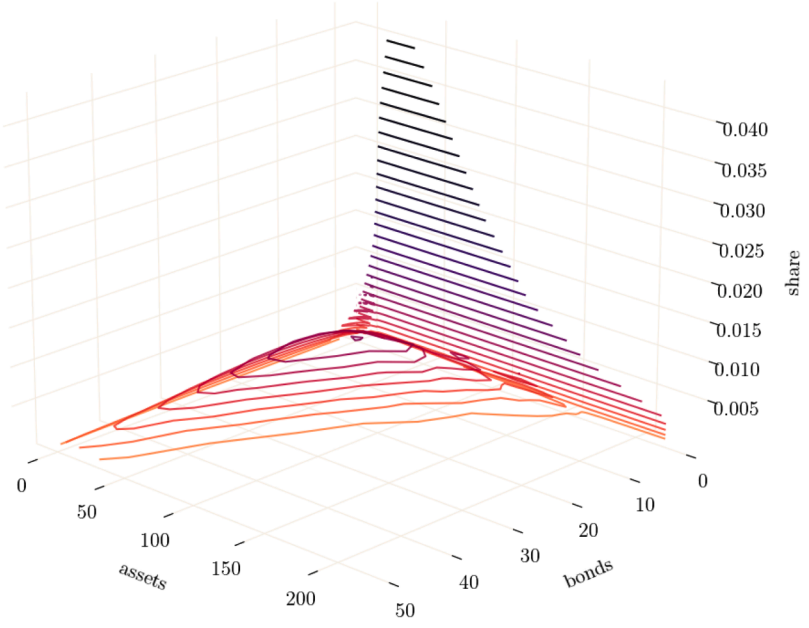

Fig. 1. The stationary distribution of the heterogeneous agent New Keynesian model with two-assets. Coloring corresponds to the share of households represented on the vertical axis. Note that for the sake of clearer display, quantities are given as shares of nodes on a log-grid (rather than true densities), meaning that shares for larger values on the grid may seem overrepresented.

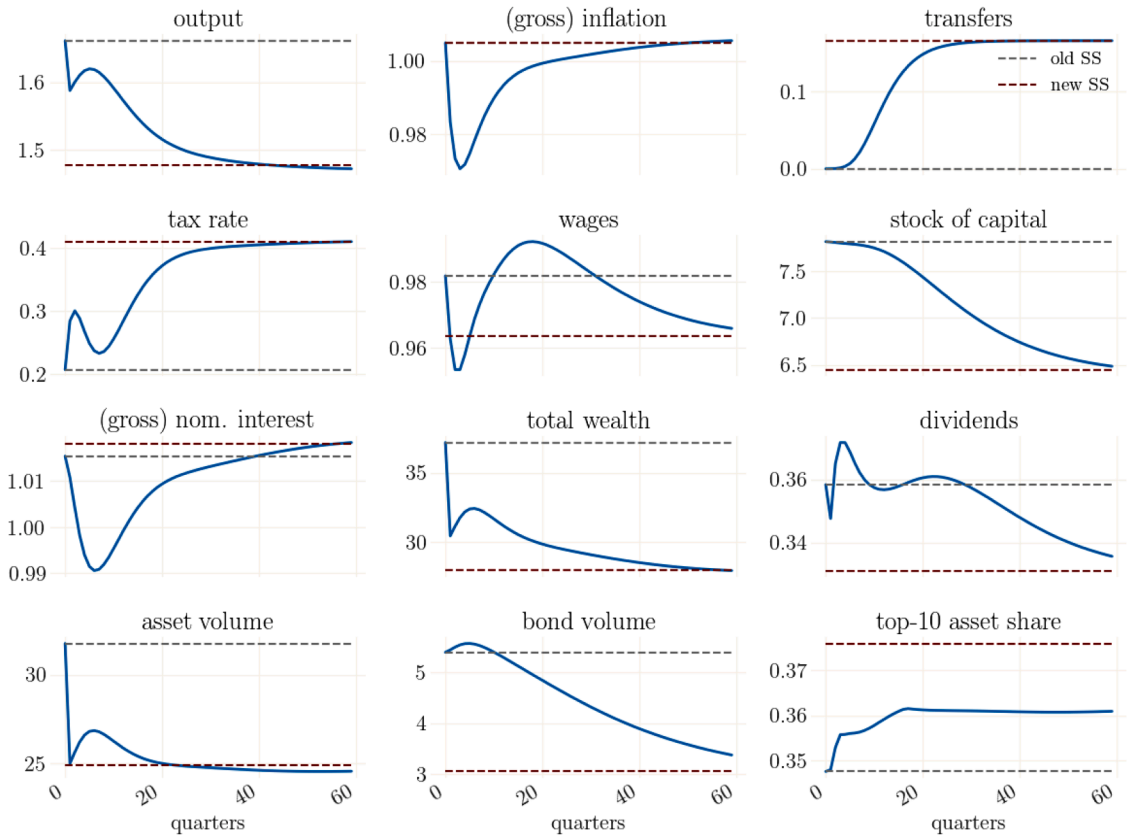

Fig. 2. Nonlinear transition dynamics for a permanent increase in government transfers. All measures are given in levels and coloring corresponds to the level of wealth. Interest and inflation rates are given in quarterly gross-rates. The dashed gray and red lines represent the old steady state without and with redistribution, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

motives, inequality in asset holdings-measured by the top 10 percentile share-increases for both liquid and illiquid assets. However, because redistribution raises transfer income among the poorest households, consumption inequality declines slightly. Thus, while the increase in labor taxation reduces efficiency, the redistribution mechanism significantly impacts household consumption patterns, benefiting liquidity-constrained agents.

The magnitude of these very negative effects may, at first, seem surprising. A common intuition in HANK models suggests that transfers could have positive demand effects, as some households have high marginal propensities to consume. However, in this model, the dominant force driving the results is the labor supply response, which is governed by the wage Phillips curve in Eq. (73).

In its core form, and absent any nominal wage rigidities (i.e., for $\psi_w = 0$ and $\mu_t^w = 1$), the collective wage-setting condition simplifies to

$$\chi n_t^{\sigma_l} = (1 - \tau_t) w_t \int e_{it} c_{it}^{-\sigma} di. \tag{89}$$

This equation shows that labor supply depends on the after-tax wage, which is reduced by τ_t, creating a direct distortion. Additionally, the aggregate marginal utility of consumption $\int e_{it} c_{it}^{-\sigma} di$ amplifies this effect: when less productive households increase consumption most, the decline in their marginal utility of consumption further discourages labor supply.

Since steady-state wages are pinned down by firms' marginal costs, labor supply must fall sharply in response to an increase in τ_t. Even if wealth effects were eliminated using Greenwood-Hercowitz-Huffman (GHH) preferences (Greenwood et al., 1988), the tax distortion would still depress labor supply. This means that even if the increase in transfers were financed by debt rather than higher labor taxes, the contractionary effects on labor and output would persist.

Finally, as shown in Table 1, the new steady state features a higher real interest rate. With higher lump-sum transfers, households reduce their demand for self-insurance and offload bonds and assets. In general equilibrium, this implies that bond prices fall, pushing up rates of return to increase compensation for holding assets.

Let us next turn to the dynamic responses that follow after implementing such redistributive policy, i.e., the transition dynamics shown in Fig. 2. Most centrally, after the announcement of the new policy, firms gradually lower their capital stock in response to the expected long-run decline in labor supply. Due to the presence of capital adjustment costs, this process is smooth rather than

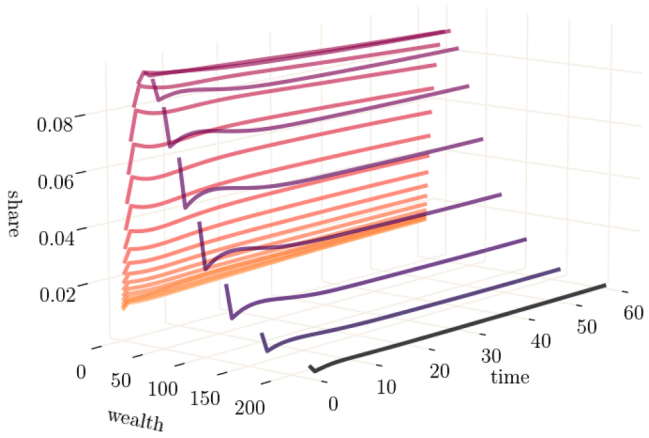

Fig. 3. Nonlinear transition dynamics of the distribution of illiquid asset for a permanent increase in government transfers. Each line presents the transition over time of the share associated with one grid node, meaning that the shares of larger grid values are overrepresented.

instantaneous. As firms reduce investment, labor demand declines, leading to lower wages and employment. At the same time, the capital-to-output ratio temporarily increases, which lowers the marginal productivity of capital and reduces firms' marginal costs. This decline in marginal costs triggers disinflation.

Monetary policy responds to the fall in inflation by lowering the nominal interest rate to stimulate consumption. However, in anticipation of higher steady-state real interest rates, the central bank adjusts its target rate upward, while the inflation target is fixed. Due to interest rate smoothing, this adjustment occurs gradually rather than abruptly. As the increase in transfers gains momentum, the government must finance these expenditures by raising labor taxes. This leads to higher pre-tax wages, which push against the disinflationary pressure and gradually bring inflation back toward its target. The speed of this process is further slowed by price and wage inertia.

As firms seek to reduce their capital stock, they temporarily maintain elevated dividend payouts. However, the expectation of lower future profits and capital accumulation results in a sharp one-time drop in equity prices, causing a significant devaluation of assets. While the main adjustment toward the new fiscal regime is largely completed within 20 quarters (five years), the economy continues to exhibit high persistence. In particular, the full adjustment of capital stock and wages takes roughly 60 quarters (15 years) to fully materialize.

Fig. 3 shows the transition dynamics between the two steady states of the distribution of assets. Wealthy households experience the announcement of the new policy as a strong negative news shock, leading to an immediate and sharp devaluation of their asset holdings. The large-scale asset devaluation destroys a considerable amount of wealth in the economy, primarily affecting the upper percentiles of the distribution. Consequently, the decline in the new steady-state supply of assets, as documented in Table 1, is reflected over time in a lower tail mass in the distribution of assets, as shown in Fig. 3. Notably, the convergence of the asset distribution is considerably slower than the adjustment of aggregate macroeconomic variables. This can also be observed when comparing the convergence speed of inequality dynamics in asset holdings, measured by the top-10-percentile share in Fig. 2, with the faster adjustment of any aggregate variable.

4.2. The role of the ZLB and DNWR during the transition

The presence of strong nonlinearities, such as the zero lower bound (ZLB) on nominal interest rates and downward nominal wage rigidity (DNWR), can have significant effects on transition dynamics. At the ZLB, the central bank is reluctant to set the nominal interest rate below zero to prevent households from hoarding cash, which would weaken monetary policy transmission. DNWR implies that firms, either due to regulatory constraints or incentive considerations, are unable to lower *nominal* wages beyond a certain threshold. Both constraints pose severe challenges to numerical solution techniques, as the resulting dynamics can deviate strongly from those implied by a linear approximation of the system. These features, therefore, provide excellent test cases to illustrate the potential of the EP method.

Fig. 4 again shows the transition dynamics following the announced gradual and permanent increase in transfers, as discussed in the previous subsection. The gray dashed line represents the baseline case, identical to Fig. 2, where neither the ZLB nor DNWR constraints are active. The blue line illustrates the case in which the ZLB is binding, while the orange line represents the scenario where nominal wages are downwardly rigid (DNWR).

The dynamic effects of the ZLB are qualitatively similar to those observed in representative agent models with a binding ZLB constraint: a shock has caused deflationary dynamics and the central bank wishes to stimulate the economy. Once the ZLB binds, the central bank can not lower interest rates further and real rates rise, leading to a decline in aggregate demand. Households respond by reducing consumption, as their marginal utility of consumption rises in the absence of monetary accommodation. With weaker demand, firms face lower marginal costs, leading to an even deeper decline in inflation. This disinflation further increases ex-post

real interest rates, exacerbating the initial contraction in consumption and output. The effects on inequality – measured by the top percentiles of asset holdings – are particularly stark: both inequality measures double in the initial spike relative to the scenario without the ZLB. This outcome is driven by the fact that only wealthier households benefit from the relatively higher real rates, while those with low net worth suffer the most from the lack of monetary stimulus and reduced labor demand.

The orange line in Fig. 4 shows the transition when nominal wages are downwardly rigid (DNWR). The calibration assumes that wages can decline by a maximum of 2% per quarter, consistent with empirical estimates. Since nominal wages cannot adjust freely, real wages remain elevated in the early phases of the transition, preventing firms from fully reducing labor costs. This wage rigidity limits the initial disinflationary response, as higher-than-equilibrium wages sustain some degree of labor income and demand. However, since firms are unable to lower wages sufficiently, they compensate by cutting employment more aggressively, leading to a stronger contraction in the labor market. While the presence of DNWR does slow down the adjustment process and introduces moderate inefficiencies, its overall impact on output is relatively mild compared to the severe contraction observed under the ZLB scenario. This highlights the asymmetry in how these two nonlinearities shape the transition: while DNWR primarily delays labor market adjustments, the ZLB fundamentally alters the transmission of monetary policy, amplifying the contraction in aggregate demand with significant effects for the overall economy.

4.3. *The role of the distribution for the transition of shocks*

As a final exercise, let us focus on the role of the distribution of wealth in the transmission of transitory shocks, i.e., one-time shocks without persistence. To isolate this effect, I simplify the setup drastically by abstracting from capital (i.e., $\alpha = 0$), labor unions ($\psi_w = 0$, $\mu_t^w = 1$), and price and wage indexation ($\omega_p = \omega_w = 1$). Additionally, I set the interest rate smoothing parameter ρ to zero. In this simplified environment, the model collapses to the two-asset HANK counterpart of the canonical three-equation model, and dummyTXdummy– all persistence in response to economic shocks arises solely from temporary shifts in the distribution of wealth.

Consider a one-time transfer equal to 5% of GDP. As before, in the baseline scenario the government operates under a balanced budget, requiring that the increase in transfers be offset by an equivalent increase in labor taxes. This setup makes the shock a pure redistribution shock, directly shifting income from high-income households to low-income households. Fig. 5 illustrates the impulse responses to this redistribution.

When taxes rise, households experience a decline in after-tax labor income, prompting a reduction in labor supply. To compensate for this decline, firms increase pre-tax wages, as they need to attract workers despite the tax wedge. Thus, the increase in labor taxes again acts as a distortionary wedge, driving a contraction in hours worked and total output.

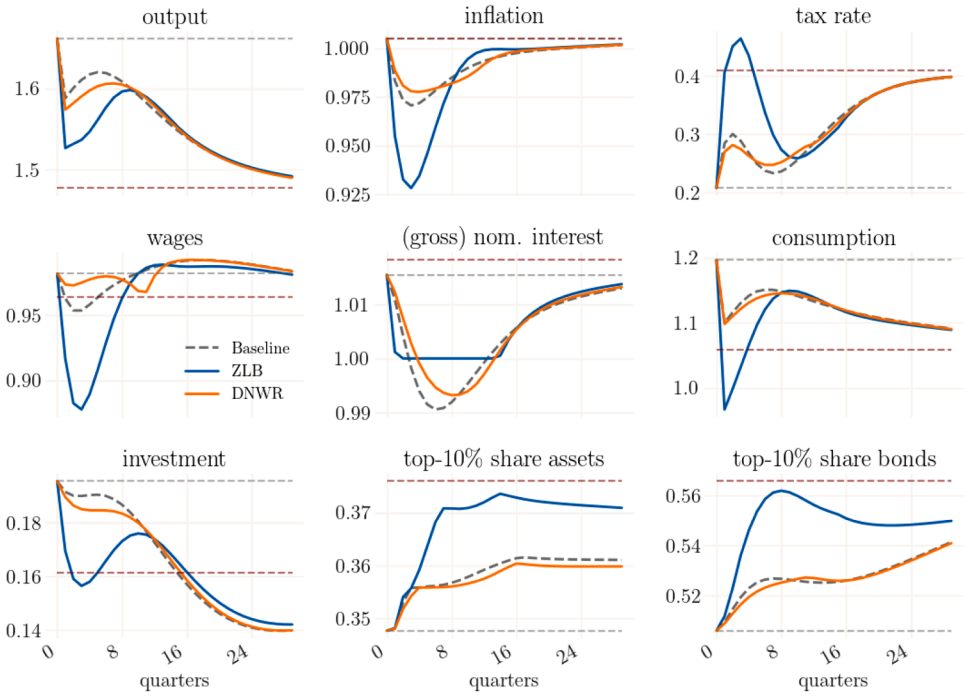

Fig. 4. Nonlinear transition dynamics for a permanent increase in government transfers. The blue and orange lines show the transition dynamics with an active ZLB and DNWR, respectively. Both are inactive for the dashed black line. All measures are given in levels. Interest and inflation rates are given in quarterly gross-rates. The dashed gray and red lines represent the old steady state without and with redistribution, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

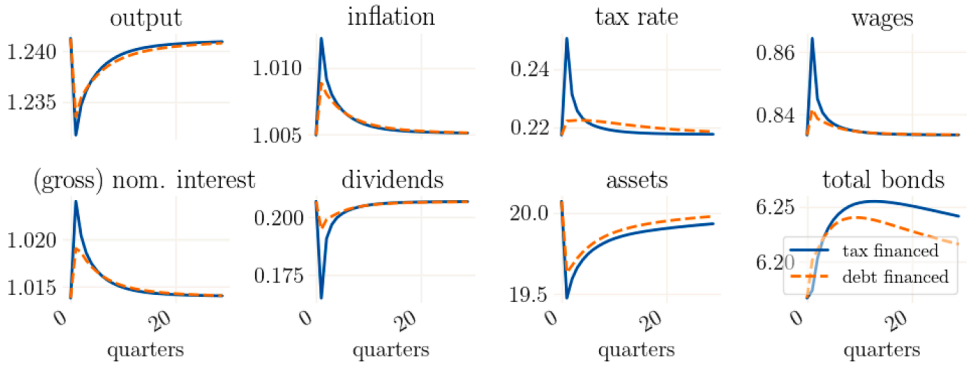

Fig. 5. Nonlinear impulse responses to a one-time increase in government transfers. All measures are given in levels. Interest and inflation rates are given in quarterly gross-rates. The solid blue line depicts simulations in which the government runs a balanced budget, while the dashed orange line shows simulations where the transfer shock is debt-financed in the medium run. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Additionally, the rise in real wages generates inflationary pressure, pushing up prices. This contrasts with previous results, where disinflation was driven by capital stock depletion; in this case, with no capital adjustment channel, the inflationary effect dominates. The central bank responds immediately by raising nominal (and, thereby, real) interest rates to counteract inflationary pressures. This policy response has two effects. First, the higher interest rate increases government debt-servicing costs, requiring further tax increases to finance interest payments. Second, the rise in the real interest rate encourages households to save more, reinforcing the initial contraction in aggregate demand. As dividends fall due to lower firm revenues and rising costs, the return on equity declines, making bonds more attractive relative to assets. Consequently, bond demand increases while asset demand declines, leading to an overall reduction in aggregate wealth.

Since the shock is transitory, its impact on wealth inequality is relatively minor. Households with significant asset holdings suffer small wealth losses due to lower dividends, while less productive households, which do not hold liquid assets, save a portion of their transfer. As a result, inequality in asset holdings rises slightly (with the top-10 % holding a greater share, not shown in the figure), while inequality in bond holdings falls slightly (as lower-income households accumulate bonds). Following the shock, households gradually deplete their excess savings, bringing the economy back to its steady state.

To assess the contribution of distortionary taxation to the observed contraction in output, I extend the model by allowing for government debt accumulation and gradual repayment through future labor tax increases. For this, Eq. (79) is replaced by

$$\tau_t \mathfrak{w}_t n_t + b_t^g = \frac{R_{t-1}}{\pi_t} b_{t-1}^g + g_t + T_t, \tag{90}$$

which introduces government debt b_t^g as a dynamic variable rather than requiring immediate fiscal balance. Taxes now follow a smoothed law of motion,

$$\tau_t = (1 - \rho_\tau)\tau_t^* + \rho_\tau \tau_{t-1}, \tag{91}$$

where τ_t^* represents the long-run tax rate necessary to balance the budget. It is given by

$$\tau_t \mathfrak{w}_t n_t = \left(\frac{R_{t-1}}{\pi_t} - 1 \right) b_{t-1}^g + g_t + T_t. \tag{92}$$

For the tax smoothing parameter, I set $\rho_\tau = 0.9$, ensuring that tax adjustments occur gradually over time.

The corresponding simulations are depicted by the dashed orange line in Fig. 5. These results confirm that the inflationary response is primarily driven by labor tax increases, as debt financing significantly dampens the inflationary impact. However, despite the moderation in inflation, the negative output effect remains nearly unchanged, suggesting that the contraction in output is driven primarily by supply-side distortions rather than a reduction in aggregate demand. Interestingly, total bond demand (including government bonds) is higher under the balanced-budget scenario than under debt-financing. This reflects the fact that equities become less attractive relative to bonds, as firms reduce expected future dividends in response to lower revenues and higher costs.

5. Design and benchmarks

The methods used in macroeconomics in general, and in particular the methods presented in this paper, have reached a degree of complexity that not only requires expert knowledge on numerical methods but also on computational and programming tools. Two implications follow from this insight: First, for a methodological contribution to be useful for the general economics community it is important to provide high-level reference implementations that do not require expert computational knowledge. Such a high-level implementation is a program with strong abstraction from the computational details of the implementation. Second, it calls for a design of software that is comprehensible and extensible, but yet allows for reproducible research that can be conducted efficiently.

This section discusses these design and implementation requirements in detail. The methodological contributions of this paper are implemented in the *econpizza* software package, which I propose as a blueprint to address these issues. Together with the package, I suggest a generic high-level syntax to express heterogeneous agent models, which is presented thereafter. I then give speed benchmarks for the reference implementation. Concrete details on the econpizza software – including guides and tutorials – are redirected to the extensive online documentation of the package.[23] Finally, I provide a comparison of EP with other recently proposed methods.

5.1. Design choices, open-source software, and the importance of reusable code

The reference implementation addresses five requirements, which I regard as central for the future progress of macroeconomic research:

i.) **Strict separation between the input representing the economic model, the libraries with solution routines, and routines for economic analysis.** Frequently used numerical routines should be organized into software libraries, which are then reusable *across* models. The use of such standardized and reusable packages – instead of relying on large blobs of user-written routines – has the potential to largely reduce the complexity of individual codes.[24] Consequently, it is necessary to strictly separate the model from the solution routines. To this end, the econpizza package not only is a high-level library for solving heterogeneous agent models, but also implements a syntax to generically represent heterogeneous agent models, which is discussed further below.

ii.) **Adherence to the open-source paradigm.** With the rising complexity of numerical methods, their performance increasingly depends on the quality of their implementation. This means that software libraries should improve over time, must allow for corrections or usability enhancements, and should adapt to computational advances. Although tempting, it is not a fruitful approach for PhD students and young scholars to write simulation programs from scratch but, rather, to build and elaborate on an existing codebase. To allow this, the *econpizza* is publicly developed on the version control platform GitHub, which allows users to suggest changes to the code, point out potential bugs, or to propose new functionalities.[25] Such version control systems play a central role in modern software development and are also widely used in, e.g., physics or engineering.

iii.) **Integration in a modern software development workflow.** Another objective is to maintain a clean, working, and well-documented code base. To satisfy this requirements, for the reference implementation tools for versioning (see above), automated unit testing, automatic code linting, and automated module documentations are employed. Automated unit tests ensure that, after any changes, the publicly available code produces economically correct results.[26] Linting is an automated procedure to format code such that it adheres to official coding style guidelines. This greatly improves readability of code and thus supports extensibility and reproducibility.[27] Automated module documentation consists of auto-generated webpages that are automatically generated from the documentation strings of classes and functions, which help to increase the transparency and traceability of the software.[28]

iv.) **Maintenance of a high-level information flow between the user and the software.** In practice, low-level routines may – for various reasons not provide the expected results. As only a cursory understanding of the underlying implementation can be expected from the user, an insufficient information flow bears the risk of unintentional misuse and false results. It is thus fatal if internal errors are not sufficiently propagated and communicated. For this reason it is crucial to implement reliable checks and informative warning and error messages for the underlying routines and potential pitfalls.

v.) **Use of a modular programming language that fully supports the functional programming paradigm.** It is highly challenging to write (and maintain) code libraries written in languages with limited support for functional programming.[29] This complicates writing and sharing function libraries that are reusable across frameworks and models because the libraries quickly becomes intractable in size and functions are inflexible. Furthermore, a programming language should integrate well with versioning systems (such as, e.g., GitHub) and feature a straightforward packaging system.[30] For these reasons, the community has recently

[23] The online documentation can be found at https://econpizza.readthedocs.io.

[24] A different dimension of this problem is that macroeconomic research is currently struggling with issues of insufficient replicability of numerical work. It can not be assumed that journal referees have the time budget or the ability to check and verify the large amounts of codes that are necessary for a single contemporary research project. A reduction in complexity can thus increase the transparency of macroeconomic research.

[25] This is the fundamental concept behind open-source software. It is hence somewhat counterproductive, in the sense of points i and ii, to publish a blob of inseparable model, simulation, and analysis codes on a private website where they can not evolve over time.

[26] Unit testing is implemented through pytest, a widely used testing framework in Python, running in GitHub Actions. The latter is a free GitHub service that automatically employs the tests on a remote server whenever updates are pushed to the GitHub repository.

[27] In econpizza, automated linting is implemented through the autopep8 package (which enforces the PEP 8 style guide) and pre-commit hooks. Such hooks are running automatically before code is uploaded to GitHub.

[28] Automated documentation for econpizza is implemented through Sphinx, a documentation generator widely used by the Python community, and employed on Read the Docs, which is an open-sourced free software documentation hosting platform used by many open-source projects.

[29] For example, the Matlab software widely used in economics restricts the number of function definition per file to one, and function definitions do not allow for default arguments. A default argument is an argument to a function that a programmer is not required to specify because a default value is provided.

[30] A packaging system allows to easily install additional modules/packages which provide specific functions and classes. Python and R, and more recently also Julia, have very rich ecosystem of packages for a large variety of applications, the large majority of which are well-tested and developed in the public domain. The econpizza package can be directly installed via the official Python repositories.

Table 2
Speed benchmarks for the three models provided in Appendix A. All numbers are in seconds. The HANK2 model without capital is the small-scale HANK model with two assets as described in Section 4.3.

	tentative (T = 150)		full (T = 300)	
Model	once	subsequent	once	subsequent
RANK	0.742	0.142	0.893	0.282
HANK1	3.546	1.071	7.153	3.092
HANK2 (no capital)	8.156	1.278	25.776	5.766
HANK2	10.355	1.076	35.858	4.763

started to adapt free and open-source languages such as Python and Julia. In particular, Python is highly flexible, simple to use, and is well integrated into modern development workflows.[31] The reference implementation is written in Python using the JAX framework, which provides just-in-time compilation and automatic differentiation. Additional details on JAX are given further below.

A fundamental design principle (cf. Point i. above) of the econpizza package is to separate the input of the economic model (provided by the user), the underlying solution routines (the software package), and the economic analysis of the simulation outcomes. The reference implementation provides a simple syntax for expressing heterogeneous agent models, which is based on the widely used YAML format.[32] Details can be found in Appendix B.

5.2. Speed benchmarks

The implementation in the econpizza package heavily leverages on just-in-time ("jit") compilation via the Python framework JAX. JAX is an open-source machine learning framework developed by Google which supports high-level automatic differentiation and jiting while providing the same syntax as NumPy, which is the primary Python library for numerical computing. The execution speed of JAX-jitted functions is on par with execution speed of compiled code from languages as Fortran or C.

Table 2 provides speed benchmarks for the four baseline models presented in Appendix A.[33] Note that RANK models are not solved using the method from Section 2 but rely on a simpler procedure that exploits the block tridiagonal structure of the sequence space Jacobian, which is outlined in Appendix C. Using this well-known procedure helps to quantify the computational overload implied by introducing heterogeneous agents. The simulations listed under "tentative" use a truncation horizon of 150 (vs. 300 for "full") and a slightly smaller grid (50 grid points for the asset grid of the HANK1 model and 10 and 20 grid points for the liquid and illiquid assets in the HANK2 models) than reported in Appendix A. The simulation results from these tentative simulations are very similar to the results obtained when using the full grid as specified in Appendix A, and the full truncation horizon. Yet, while researchers often use such short horizons while studying their models, they often want to use longer truncation horizons in the final versions of their project to avoid numerical inaccuracies.

The table documents a large speed difference between the first simulation ("once") and each successive simulation ("subsequent"). Subsequent simulations allow to use different shocks, initial conditions, or parameters that do not alter the steady state. These simulations are much faster for two reasons. First, for each new steady state the steady state sequence space Jacobian must be recomputed including its LU decomposition. The computational load of the Jacobian and the LU decomposition vary substantially with the length of the truncation horizon. Second, the model functions must be compiled (which is done automatically by JAX). Both steps are not necessary if the steady state remains unaltered when the objects can simply be reused.

Runtimes of the RANK model are much faster since solving the RANK model does not require the calculation of the steady state Jacobian and its LU decomposition. Additionally, automatic differentiation does not have to traverse through the value functions and the distributions. Comparing the HANK1 model to the HANK2 model, calculation and compilation times roughly double while the number of grid points remains roughly the same. Further, calculation and compilation times of the medium-scale HANK2 model with capital compared to the small-scale HANK2 model without capital are only marginally larger although the complexity of the aggregate system of equations of the medium-scale model is considerably larger than the complexity of the small scale model. Since the disaggregated problem is exactly the same for both models, this suggests that the complexity of the aggregated system of equations is of second order importance for the computational complexity.

[31] Python is an object-oriented general purpose language used in a large field of applications. It is the de-facto industry standard in data science and machine learning and supported by major big-tech firms. Python is free and open source with no limiting licences or additional costs, has a huge active user base and its design simplicity allows for a high code quality.

[32] YAML ("Yet Another Markup Language") and is a standardized human-readable data-serialization language. The format is similar to XML but has a minimal syntax in order to be easily usable. It is useful to provide data input in a clear and simple way across programming languages, and is widely used in applications that require a high level human-computer interaction, such as configuration files or data storage.

[33] All benchmarks are done on a standard laptop with 8 Intel(R) Core(TM) i7-8650U CPUs (1.90GHz). The package does not make explicit use of parallel computing.

Table 3
Comparison of largest possible shocks for EP and SSJ of all shocks in the two-asset HANK model including calculation times. Calculation times are given in seconds. The shocks are, in the given order: discount factor shock, monetary policy, investment technology, price markup, wage markup, technology, government spending, transfers. A "moderate shock" is chosen to half the size of the largest possible shock with SSJ. "peak output" is given in percentage deviation from the steady state. For both methods, a horizon of $T = 200$ is assumed.

shock type	disc. fct.	mon. pol.	inv. tech.	price MU	wage MU	tech.	gov. spend.	transfers
	moderate shock*							
calc. time SSJ	15.70	15.33	14.61	15.38	14.13	14.96	14.84	15.42
calc. time EP	1.11	1.07	1.83	1.01	3.79	1.03	1.03	1.84
	largest possible shock with SSJ							
shock size	0.012	0.007	0.174	0.029	0.276	0.025	0.244	1.303
peak output (%)	−1.5	−1.2	1.5	−0.5	−2.6	1.2	2.0	−1.8
calc. time SSJ	16.71	16.12	15.34	15.58	15.67	15.38	15.27	15.47
calc. time EP	2.47	2.57	6.78	2.95	12.97	2.20	2.97	5.74
	largest possible shock with EP							
shock size	0.259	0.095	1.349	1.080	1.191	0.460	1.147	2.678
peak output (%)	−18.2	−9.9	14.4	−10.3	−8.0	31.2	21.2	−5.4
calc. time EP	10.61	14.50	28.00	27.01	38.52	10.03	29.88	33.37
	linear approximations (all shocks)							
calc. time SSJ				9.23				
calc. time EP				6.46				

5.3. Comparison with other methods

The three main competitors to the EP method (and software package) are (Reiter, 2009), (Bayer et al., 2020), and (Auclert et al., 2021), of which the first two share some similarities as they are based on the state space representation. In contrast, the sequence space Jacobian (SSJ) approach of Auclert et al. (2021) and EP are based on the sequence space representation. I first briefly discuss the advantages and disadvantages of the state space approach, and then compare SSJ and EP in detail. Importantly, the three methods cited above are usually only applied in the context of linearized dynamics. As documented in the respective papers, their performance is overall comparable.

The advantage of the state-space approach is – when going beyond linear approximations – that it enables the application of second- and higher-order perturbation techniques. This is particularly beneficial as it allows, relative to SSJ and EP, to account for aggregate uncertainty. However, the reliance on perturbation methods also introduces key limitations. First, since perturbation inherently involves a local approximation around a steady state, the state-space approach is not well-suited for solving the nonlinear perfect foresight path, which is particularly relevant if one is interested in highly nonlinear models with large deviations from steady state, or strong nonlinearities such as, e.g., occasionally binding constraints. Second, these methods typically incorporate algorithms for dimensionality reduction of the distribution and value functions to make computation feasible. While this dimensionality reduction is necessary for tractability, it introduces an additional source of approximation errors. This highlights the trade-offs between accuracy in handling aggregate uncertainty and the ability to fully capture nonlinear dynamics in deterministic settings.

Of these competitors, only the SSJ approach by Auclert et al. (2021) allows for basic nonlinear transition dynamics. The remainder of this subsection thus compares the versatility and performance of SSJ versus the EP method. Similar to the EP method, nonlinear SSJ is based on a sequence-space approach where a Newton method iterates on the expected perfect foresight equilibrium path up to a truncated horizon. However, unlike the EP method, SSJ computes each Newton iteration using the Jacobian of the steady-state sequence rather than an approximation of the true Jacobian. This technique, known as the *Chord method*, is known to perform well in the vicinity of the steady state but often fails when equilibrium dynamics exhibit significant nonlinearities such as, e.g., the zero lower bound. In contrast, the EP method is able to deal with such strong nonlinearities, even if they are far away from the steady state.

To compare the two methods, I replicate the two-asset HANK model with the features of the medium-scale model within the SSJ framework. Since the underlying model structure remains identical, both methods yield the same dynamic responses up to the desired level of numerical precision. However, I find that the Chord method fails to converge for equilibrium paths involving the zero lower bound (ZLB) on nominal interest rates or downward nominal wage rigidity. This is unsurprising, as these constraints introduce strong nonlinearities that can substantially alter dynamic responses to shocks.

To quantify the extent to which both methods handle standard shocks in a nonlinear setup *without* ZLB or downward nominal wage rigidity, I implement an iterative procedure to determine the largest feasible shock each method can solve. Table 3 presents the results for a horizon of 200 and excluding the calculation of the steady state.[34] I compare the performance of all eight economic

[34] The steady state routine provided in Section 2 significantly outperforms any other method and is additionally able to find the steady state even if is under- or overdetermined.

shocks (columns of the table) for different magnitudes of shocks. A moderate shock is chosen such that it can be easily handled by both methods. The simulations suggest that for these shocks, EP offers a speed improvement by a factor of ten. I then determine, for each shock type, the largest shock for which SSJ can find a solution. For these shocks the speed improvement of EP over SSJ remains significant. I then determine the largest shock for which EP can find a solution. When then comparing largest possible shock sizes, the simulations suggest that EP is capable of solving for significantly larger shocks, with the maximum solvable shock approximately one order of magnitude larger in terms of the output variation.[35] The last row compares calculation times for purely linear impulse responses, for which EP beats SSJ by about 1.5 in terms of calculation time.

Thus, to summarize, to a user interested in purely linear solutions neither of the four discussed methods provides a significant advantage, although EP offers some improvements. If the user is interested in aggregate uncertainty but not in nonlinear dynamics, the second-order method of the state space approach is the method of choice. Otherwise, these benchmarking results demonstrate that EP achieves significant computational gains while maintaining robustness in solving nonlinear heterogeneous agent models. Compared to existing approaches, it efficiently handles strong nonlinearities and occasionally binding constraints, where alternative methods, such as the SSJ based solver, often fail to converge. Moreover, the method exhibits superior scalability, solving complex models with many aggregate equations in a fraction of the time required by conventional techniques.

6. Conclusion

This paper introduces an iterative method to find nonlinear solutions to macroeconomic models with heterogeneous agents. The method is based on Newton iterations and leverages the technique of automatic differentiation. I provide an easy-to-use reference implementation and suggest a series of central requirements of such an implementation. These are, among others, the consequent separation of economic model, solution code, and analysis, the generation of reusable code (across models), and adherence to the open-source philosophy. The solution method is applied to study the nonlinear transition dynamics of a gradual but permanent change in government redistribution. The overall effects of such policy, both in the short and in the long run, are contractionary and can be aggravated by strong nonlinearities such as a lower bound on interest rates or downward nominal wage rigidity.

CRediT authorship contribution statement

Gregor Boehl: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Data availability

All codes can be found at https://github.com/gboehl/econpizza

Declaration of competing interest

I am a postdoctoral researcher at the University of Bonn. My research on this project was financially supported by the Deutsche Forschungsgemeinschaft (DFG) under CRC-TR 224 (projects C01 an C05) and under project number 441,540,692 at the University of Bonn. I declare that I have no relevant or material financial interest that relate to the research described in this paper

Acknowledgements

I am grateful to Christian Bayer, Flora Budianto, Marten Hillebrand, Keith Kuester, Alexander Meyer-Gohde, Vasudeva Ramaswamy, Michael Reiter, two anonymous referees, and participants of the 2024 T2M conference and the IMFS workshop on Numerical Methods in Macroeconomics for discussions and helpful comments on the contents of this paper. I further thank the editor, Guillermo Ordonez, for providing meaningful guidance during the review process. I gratefully acknowledge financial support by the Deutsche Forschungsgemeinschaft (DFG) under CRC-TR 224 (projects C01 and C05) and under project number 441540692.

Appendix A. Models

This section presents the three main models used throughout the paper and which are provided with the package. A fourth model, applied in Section 4, is the two-asset-model without capital, which is a special case of the two-asset-model presented below. Importantly, all four models can be seen as special cases of the general medium scale two-asset HANK model. Besides some minor deviations, the three models share the same variables, which are summarized in Table A.1, and parameters, which are given in Table A.2.

[35] Both methods can be improved through various hand-tailored adjustments, such as decreasing the Newton method's step size or increasing the maximum number of allowed iterations. Additionally, EP performs nearly ten times faster than SSJ when handling moderately strong shocks, highlighting its superior computational efficiency. Finally, as the table suggests, due to the computational efficiency of the accumulation of the steady state Jacobian proposed in Section 2.5, EP offers a noteworthy improvement in computation speed for linear solutions relative to SSJ.

Table A.1
Legend of variable names.

C_t	consumption	mc_t	marginal costs
y_t	output	r_t^a	return to illiquid assets
π_t	inflation	r_t^b	return to liquid assets
$\tilde{\pi}_t$	inflation inertia	$\mathfrak{w}_t^n$	notional wage
Π_t	firms profit	$\mathfrak{w}_t$	wage
s_t	equity	$\pi_t^{\mathfrak{w}}$	wage inflation
n_t	labor hours	$\tilde{\pi}_t^{\mathfrak{w}}$	wage inflation inertia
z_t	technology	A_t	total illiquid assets
k_t	capital stock	B_t	total liquid assets
q_t	price of capital	β_t	discount factor
i_t	investment	μ_t	price markup
R_t	nominal interest rate (post ZLB)	$\mu_t^{\mathfrak{w}}$	wage markup
R_t^n	notional interest rate	ϵ_t^i	investment technology
R_t^*	interest rate target	MPK_t	marginal productivity of capital
τ_t	tax rate	u_t	capital utilisation
g_t	government spending	T_t	lump-sum transfers

A.1. A medium scale RANK model

This model is based on Gust et al. (2012), which is an early working paper version of Gust et al. (2017). The reader is delegated to this reference for extensive microfoundations of the different parts of the model. Relative to this reference the model contains a series of simplifications, e.g. no growth in steady state. It features all the bells and whistles of the medium-scale workhorse model of Smets and Wouters (2007) but uses Rotemberg pricing instead of Calvo pricing. The model also forms the basis for the aggregate equations of the two HANK models, where the disaggregated household sector replaces the household block.

Households

$$\Lambda_t = \frac{1}{c_t - hc_{t-1}} - \frac{h\beta_t}{E_t c_{t+1} - hc_t} \tag{A.1}$$

$$\Lambda_t = \beta_t \epsilon_t^{\Lambda} E_t \left\{ \Lambda_{t+1} \frac{R_t}{\pi_{t+1}} \right\} \tag{A.2}$$

$$d_t + c_t + \tau_t + \frac{\psi_{\mathfrak{w}}}{2}\left(\frac{\pi_t^{\mathfrak{w}}}{\tilde{\pi}_t^{\mathfrak{w}}} - 1\right)^2 = \mathfrak{w}_t n_t + R_{t-1}/\pi_t d_{t-1} + \Pi_t + \Pi_t^b \tag{A.3}$$

The last equation is the households budget constraint which not necessary for the aggregate dynamics due to Walras' law.

Labor unions

$$\psi_{\mathfrak{w}}\left(\frac{\pi_t^{\mathfrak{w}}}{\tilde{\pi}_t^{\mathfrak{w}}} - 1\right)\frac{\pi_t^{\mathfrak{w}}}{\tilde{\pi}_t^{\mathfrak{w}}} = \psi_{\mathfrak{w}}\beta_t E_t\left\{\left(\frac{\pi_{t+1}^{\mathfrak{w}}}{\tilde{\pi}_{t+1}^{\mathfrak{w}}} - 1\right)\frac{\pi_{t+1}^{\mathfrak{w}}}{\tilde{\pi}_{t+1}^{\mathfrak{w}}}\right\} + n_t \frac{\mu_t^{\mathfrak{w}}}{\mu_t^{\mathfrak{w}} - 1}\left(\chi n_t^{\sigma_l} - \Lambda_t \mathfrak{w}_t/\mu_t^{\mathfrak{w}}\right) \tag{A.4}$$

$$\pi_t^{\mathfrak{w}} = \frac{\mathfrak{w}_t^n}{\mathfrak{w}_{t-1}^n}\pi_t \tag{A.5}$$

$$\ln \tilde{\pi}_t^{\mathfrak{w}} = \omega_{\mathfrak{w}} \ln \bar{\pi}^{\omega} + (1 - \omega_{\mathfrak{w}}) \ln \pi_{t-1}^{\mathfrak{w}} \tag{A.6}$$

$$\mathfrak{w}_t = \max\left\{\iota_{\mathfrak{w}} \frac{\mathfrak{w}_{t-1}}{\pi_t}, \mathfrak{w}_t^n\right\} \tag{A.7}$$

Firms

$$\psi\left(\frac{\pi_t}{\tilde{\pi}_t} - 1\right)\frac{\pi_t}{\tilde{\pi}_t} = \frac{1}{1 - \mu_t} + \frac{\mu_t}{\mu_t - 1} mc_t + \psi E_t\left\{\beta_{t+1}\frac{\Lambda_{t+1}}{\Lambda_t}\left(\frac{\pi_{t+1}}{\tilde{\pi}_t} - 1\right)\frac{\pi_{t+1}}{\tilde{\pi}_t}\frac{y_{t+1}^f}{y_t^f}\right\} \tag{A.8}$$

$$\ln \tilde{\pi}_t = \omega \ln \bar{\pi} + (1 - \omega) \ln \pi_{t-1} \tag{A.9}$$

$$y_t^f = (u_t k_{t-1})^{\alpha} (z_t n_t)^{1-\alpha} \tag{A.10}$$

$$k_t = (1 - \delta)k_{t-1} + \epsilon_t^i\left(1 - \frac{\psi_i}{2}\left(\frac{i_t}{i_{t-1}} - 1\right)^2\right)i_t \tag{A.11}$$

$$1 = q_t \epsilon_t^i\left(1 - \frac{\psi_i}{2}\left(\frac{i_t}{i_{t-1}} - 1\right)^2 - \psi_i\left(\frac{i_t}{i_{t-1}} - 1\right)\frac{i_t}{i_{t-1}}\right) + \beta_t \epsilon_{t+1}^i \frac{\Lambda_{t+1}}{\Lambda_t} q_{t+1} \psi_i\left(\frac{i_{t+1}}{i_t} - 1\right)\left(\frac{i_{t+1}}{i_t}\right)^2 \tag{A.12}$$

$$q_{t-1}\frac{R_t}{\pi_{t+1}} = MPK_t u_t + (1-\delta)q_t - C(u_{t-1}) \tag{A.13}$$

$$\mathfrak{w}_t = (1-\alpha)mc_t\frac{y_t^f}{n_t} \tag{A.14}$$

$$MPK_t = \alpha mc_t\frac{y_t^f}{(u_t k_{t-1})} \tag{A.15}$$

$$C(u_t) = \bar{MPK}(u_t - 1) + \frac{1}{2}\frac{\psi_u}{1-\psi_u}(u_t-1)^2 \tag{A.16}$$

$$\psi_u(u_t - 1) = (1-\psi_u)(MPK_t - \bar{MPK}) \tag{A.17}$$

$$\Pi_t = \left(1 - mc_t - \frac{\psi_p}{2}\left(\frac{\pi_t}{\tilde{\pi}_t} - 1\right)^2\right)y_t^f - \left(1 - q_t\left(1 - \frac{\psi_i}{2}\left(\frac{i_t}{i_{t-1}} - 1\right)^2\right)\right)i_t \tag{A.18}$$

Financial sector

$$d_t = q_t^b b_t + q_t k_t \tag{A.19}$$

$$R_t = \frac{(1+\kappa q_{t+1}^b)}{q_t^b} \tag{A.20}$$

$$\Pi_t^b = \left((1+\kappa q_t^b)b_{t-1} + R_{t-1}q_{t-1}k_{t-1} - R_{t-1}d_{t-1}\right)/\pi_t \tag{A.21}$$

Government

$$q_t^b b_t + \tau_t = g_t + \frac{(1+\kappa q_t^b)}{\pi_t}b_{t-1} \tag{A.22}$$

$$b_t = \frac{\bar{y}}{\bar{q}^b} \tag{A.23}$$

$$\ln R_t^n = \rho \ln R_{t-1}^n + (1-\rho)\left(\ln R_t^* + \phi_\pi\left[\ln \pi_t - \ln \bar{\pi}\right] + \phi_y\left[\ln y_t - \ln \bar{y}\right]\right) + \ln \upsilon_t \tag{A.24}$$

$$R_t = \max\left\{1,\ R_t^n\right\} \tag{A.25}$$

Clearing conditions

$$c_t + i_t + g_t + C(u_t)k_{t-1} + \frac{\psi_{\mathfrak{w}}}{2}\left(\frac{\pi_t^{\mathfrak{w}}}{\tilde{\pi}_t^{\mathfrak{w}}} - 1\right)^2 = \left(1 - \frac{\psi_p}{2}\left(\frac{\pi_t}{\tilde{\pi}_t} - 1\right)^2\right)y_t^f \tag{A.26}$$

$$c_t + i_t + g_t = y_t \tag{A.27}$$

Exogenous processes

$$\ln \beta_t = (1-\rho_\beta)\ln \bar{\beta} + \rho_\beta \ln \beta_{t-1} + \varepsilon_t^\beta \tag{A.28}$$

$$\ln z_t = (1-\rho_z)\ln \bar{z} + \rho_z \ln z_{t-1} + \varepsilon_t^z \tag{A.29}$$

$$\ln \mu_t^{\mathfrak{w}} = (1-\rho_{\mathfrak{w}})\ln \bar{\mu}^w + \rho_{\mathfrak{w}} \ln \mu_{t-1}^{\mathfrak{w}} + \varepsilon_t^w \tag{A.30}$$

$$\ln \mu_t = (1-\rho_p)\ln \bar{\mu} + \rho_p \ln \mu_{t-1} + \varepsilon_t^p \tag{A.31}$$

$$\ln g_t = (1-\rho_g)\ln(0.2\bar{y}) + \rho_g \ln g_{t-1} + \varepsilon_t^g \tag{A.32}$$

$$\ln \epsilon_t^i = \rho_i \ln \epsilon_{t-1}^i + \varepsilon_t^i \tag{A.33}$$

$$\ln \epsilon_t^\Lambda = \rho_\Lambda \ln \epsilon_{t-1}^\Lambda + \varepsilon_t^\Lambda \tag{A.34}$$

$$\ln R_t^* = (1-\rho_r)\ln \bar{R} + \rho_r \ln R_{t-1}^* + \varepsilon_t^r \tag{A.35}$$

Parameters

The parameters are set as in Table A.2.

Equilibrium

The competitive equilibrium of the RANK model is a set of sequences of

1. Allocations $\{c_t, y_t, mc_t, k_t, i_t, n_t, u_t, b_t, \tau_t, MPK_t, C(u_t), y_t^f, \Pi_t, R_t^*, \mathfrak{w}_t^n\}$
2. Prices $\{\Lambda_t, R_t, R_t^n, \mathfrak{w}, q_t, \Pi_t^b, q_t^b, \pi_t, \tilde{\pi}_t, \tilde{\pi}_t^{\mathfrak{w}}, \pi_t^{\mathfrak{w}}\}$

Table A.2
Joint parameters.

Parameter		Value	Target
σ_l	inverse Frisch elasticity of labour supply	2	
χ	weight on the disutility of labour	–	$\bar{n} = 0.33$
β	steady state discount factor	0.995	
θ	elasticity of substitution	6	
$\theta_{\mathfrak{w}}$	elasticity of substitution for wages	11	
κ	decay parameter for coupon payments of perpetual bonds	0.975	
δ	depreciation rate	0.025	
h	habit formation parameter	0.74	
ψ_i	parameter on the costs of investment adjustment	5.6	
ψ_p	parameter on the costs of price adjustment	60	
$\psi_{\mathfrak{w}}$	parameter on the costs of wage adjustment	96	
ψ_u	parameter on the capital utilisation costs	0.8	
α	capital income share	0.33	
π^*	inflation target	$1.02^{0.25}$	
$\phi_p i$	Monetary policy rule coefficient on inflation	1.5	
ϕ_y	Monetary policy rule coefficient on output	0.1	
ρ	persistence in (notional) nominal interest rate	0.8	
ω_p	coefficient on steady state inflation in price indexation	0.44	
$\omega_{\mathfrak{w}}$	coefficient on steady state wage inflation in wage indexation	0.66	
$\iota_{\mathfrak{w}}$	degree of downward nominal wage rigidity	1	
ρ_β	persistence of discount factor shock	0.9	
ρ_z	persistence of technology shocks	0.9	
ρ_p	persistence of price MU shock	0.9	
$\rho_{\mathfrak{w}}$	persistence of wage MU shock	0.9	
ρ_g	persistence of government spending shock	0.9	
ρ_i	persistence of MEI shock	0.9	
ρ_r	persistence of MP shock	0.9	
ρ_u	persistence of wage MU shock	0.9	

3. Shock processes $\{\beta_t, z_t, \epsilon_t^i, \epsilon_t^\Lambda, \mu_t, \mu_t^{\mathfrak{w}}, g_t\}$
4. Exogenous shocks $\{\varepsilon_t^\beta, \varepsilon_t^z, \varepsilon_t^w, \varepsilon_t^p, \varepsilon_t^g, \varepsilon_t^i, \varepsilon_t^\Lambda, \varepsilon_t^r\}$

that satisfy

- The households' optimality conditions Eqs. (A.1) and (A.2)
- The wage setting decision by the labor union Eqs. (A.4) to (A.7)
- The firms optimality conditions Eqs. (A.8) to (A.17)
- The definitions for the financial sector Eqs. (A.19) to (A.21)
- The Government's policy responses Eqs. (A.22) to (A.25)
- Market clearing conditions Eqs. (A.26) and (A.27)
- The law of motion for exogenous processes Eqs. (A.28) to (A.35)

A.2. A small scale HANK model with one asset

In this variant, households can hold one type of assets, a_{it}, face idiosyncratic income risk and a borrowing constraint. They have GHH preferences with the composite good $x_{i,t}$, and the Bellman equation is given by

$$V_t(e_{it}, a_{i,t-1}) = \max_{c_{it}, n_{it}, a_{it}} \left\{ \frac{x_{it}^{1-\sigma_c}}{1-\sigma_c} + \beta \mathbb{E}_t \left[V_{t+1}(e_{i,t+1}, a_{it}) | e \right] \right\} \tag{A.36}$$

$$x_{it} = c_{it} - e_{it} \frac{n_{it}^{1+\sigma_l}}{1+\sigma_l} \tag{A.37}$$

$$c_{it} + a_{it} = \frac{R_{t-1}}{\pi_t} a_{i,t-1} + \mathfrak{w}_t e_{it} n_{it} - \tau_t \bar{\tau}(e_{it}) + \Pi_t \bar{\Pi}(e_{it}) \tag{A.38}$$

$$a_{it} \geq 0 \tag{A.39}$$

where e_{it} is i's household-specific productivity which follows an AR(1) process in logs as in Eq. (71). $\bar{\tau}(e)$ and $\bar{\Pi}(e)$ are skill-specific incidence rules for taxes and dividends.

The aggregate model is as in Appendix A.1 but parameters are chosen such that labor is the only production factor (i.e. no capital accumulation with $\alpha = 0$) and such that there are no price inertia in the Phillips curve ($\omega = 0$). Dividends are given by Eq. (76). The

Table A.3
Parameters specific to the one-asset-HANK model.

Parameter		Value
σ_c	intertemporal elasticity of substitution	2
$\bar{\beta}$	discount factor	0.98
b_g	bond supply	5.6
α	capital factor share	0
ω_p	coefficient on steady state inflation in price indexation	1
$\bar{a}$	borrowing constraint	0
σ_e	standard error of earnings	0.6
ρ_e	autocorrelation of earnings	0.966
n_e	points for Markov chain of e	4
n_a	points for asset grid	50

government is running a balanced budget with

$$\tau_t = \left(\frac{R_{t-1}}{\pi_t} - 1\right) b_g + g_t, \tag{A.40}$$

where taxes are collected in a lump-sum fashion instead of acting as labor taxes. I further abstract from labor unions and thus, due to GHH preferences, labor supply simplifies to

$$n_t^{\sigma_l} = \mathfrak{w}_t. \tag{A.41}$$

Markets clear with

$$\int c_{it} di = C_t = \left(1 - \frac{\psi}{2}\left(\frac{\pi_t}{\bar{\pi}} - 1\right)^2\right) y_t + g_t, \tag{A.42}$$

$$\int a_{it} di = b_g, \tag{A.43}$$

and the parameters specific to this model are given in Table A.3.

A.3. A medium scale HANK model with two assets

The two-asset HANK model shares many of the aggregate features with the representative agent model in Appendix A.1 and is presented in Section 3. A central difference is the setup of households. Based on the endogenous grid method of Carroll (2006), the appendix of Auclert et al. (2021) describes an efficient algorithm to solve the two-asset household problem with convex adjustment costs. All equations that are not stated in Section 3 are as in the RANK model, including the exogenous processes from Eqs. (A.28)–(A.35). Parameters specific to the two-asset HANK model are given in Table A.4.

The competitive equilibrium of the medium scale HANK model with two assets is a set of sequences of

1. Allocations $\{C_t, A_t, B_t, y_t, mc_t, k_t, i_t n_t, u_t, \tau_t, MPK_t, C(u_t), y_t^f, \Pi_t, R_t^*, \mathfrak{w}_t^n, s_t\}$
2. Prices $\{\Lambda_t, R_t, R_t^n, r_t^a, r_t^b, \mathfrak{w}, q_t, \Pi_t^b, q_t^b, \pi_t, \tilde{\pi}_t, \tilde{\pi}_t^{\mathfrak{w}}, \pi_t^{\mathfrak{w}}\}$
3. Distributions across a_{it} and b_{it}
4. Shock processes $\{\beta_t, z_t, \epsilon_t^i, \epsilon_t^\Lambda, \mu_t, \mu_t^{\mathfrak{w}}, g_t, T_t\}$
5. Exogenous shocks $\{\varepsilon_t^\beta, \varepsilon_t^z, \varepsilon_t^w, \varepsilon_t^p, \varepsilon_t^g, \varepsilon_t^i, \varepsilon_t^\Lambda, \varepsilon_t^r, \varepsilon_t^T\}$

that satisfy

- The solution to the households' recursive problem given in Eqs. (67) to (70)
- The wage setting decision by the labor union in Eqs. (73) and (A.5) to (A.7)
- The firms optimality conditions Eqs. (A.8) to (A.17)
- The financial sector in Eqs. (81), (87) and
- The Government's policy responses Eqs. (79) and (A.23) to (A.25)
- Market clearing conditions Eqs. (A.26) and (A.27) and (82) to (86)
- The laws of motion for exogenous processes Eqs. (A.28) to (A.35) and (80)

Appendix B. A generic syntax to express heterogeneous agent models

Using the formalization from Section 2, the necessary user input to describe a heterogeneous agent model can be reduced to two elements: the EGM step manifesting in the function $W(\cdot)$ from Eq. (1) and the n aggregate equations in $f(\cdot)$ from Eq. (3). In contrast, it is typically not necessary to explicitly specify the mapping from agents' decisions to the distribution, $D(\cdot)$, from Eq. (2) since this function is generic and standard routinen such as, e.g., the lottery method of Young (2010) can be used.

Table A.4
Parameters specific to the two-asset HANK model.

Parameter		Value
σ_c	intertemporal elasticity of substitution	2
σ_l	inverse Frisch elasticity of labour supply	2.9
χ	weight on the disutility of labour	0.5
ψ_p	parameter on the costs of price adjustment	60
ψ_w	parameter on the costs of wage adjustment	96
$\psi_a 0$	parameter on portfolio adjustment no.1	0.25
$\psi_a 1$	parameter on portfolio adjustment no.2	15
$\psi_a 2$	parameter on portfolio adjustment no.3	2
ζ	liquidity premium	0.005
b_G	government bond supply	2.8
$\bar{\beta}$	discount factor	0.98
$\bar{T}$	steady state government transfers	1e-5
ρ_T	autocorrelation government transfers	0.8
$\bar{b}$	borrowing constraint	0
σ_e	standard error of earnings	0.92
ρ_e	autocorrelation of earnings	0.966
ζ	steady state liquidity premium	0.1
n_e	points for Markov chain of e	3
n_b	points for liquid asset grid	20
n_a	points for illiquid asset grid	25

The reference implementation provides a simple syntax for expressing heterogeneous agent models, which is based on the widely used YAML format.[36] Similar as the mod-file in Dynare, the file allows to specify variables and parameters as well as meta-parameters such as, e.g., the grids used to represent the distribution of idiosyncratic states across agents. Importantly, the package – via the YAML file – permits a standardized way to specify the function $W(\cdot)$ including its inputs and outputs. This is illustrated in Fig. C.1 (key: "`decisions`") for a part of the specification of the one-asset-HANK model from Appendix A. Subfunctions of $W(\cdot)$ (e.g. the function `egm_step`) can be described as a conventional Python function that is defined in an external functions file and referenced in the YAML. In the example, `Wa` and `Wb` are the recursive decision object w_t from Section 2.1 while `a`,`b`,`c`,`uce` are the agents actions in a_t.

The figure further shows a part of the system of aggregate equations ("`equations`"), which constitute the function $f(\cdot)$.[37] Together with the specification of $W(\cdot)$, this allows the combination of the disaggregated parts of various heterogeneous agents models with arbitrary systems of aggregate equations. Additionally, the file further allows to stage parameter values and the inputs required for the steady state search (not shown in the figure).

Appendix C. Representative agent models and automatic differentiation

This appendix is merely given for completeness and documents how the software implementation handles nonlinear *representative* agent models. Analog to the representation in Section 2 for heterogeneous agent models, a representative agent model can simply be written as

$$z_t = f(x_{t-1}, x_t, x_{t+1}), \tag{C.1}$$

where the solution requires $z_t = \mathbf{0}$. For simplicity, denote the three Jacobians of this function as

$$f_A(x) = \partial f / \partial x_{t-1}, \tag{C.2}$$

$$f_B(x) = \partial f / \partial x_t, \tag{C.3}$$

$$f_C(x) = \partial f / \partial x_{t+1}. \tag{C.4}$$

For each $f(x_{t-1}, x_t, x_{t+1}) = z_t$, these three Jacobians can be obtained at low costs via backwards propagation with only n evaluations of f. Since $dim(z_t) = n \ll dim(\mathbf{z}) = n(T-1)$, each evaluation is computationally cheap. The sequence space Jacobian is then given by the block tridiagonal matrix

$$\boldsymbol{J}(\mathbf{x}) = \begin{bmatrix} f_B(x_1) & f_C(x_2) & & & & & \\ f_A(x_1) & f_B(x_2) & f_C(x_3) & & & & \\ & f_A(x_2) & f_B(x_3) & f_C(x_4) & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & f_A(x_{T-3}) & f_B(x_{T-2}) & f_C(x_{T-1}) & \\ & & & & f_A(x_{T-2}) & f_B(x_{T-1}) \end{bmatrix}. \tag{C.5}$$

[36] YAML ("Yet Another Markup Language") and is a standardized human-readable data-serialization language. The format is similar to XML but has a minimal syntax in order to be easily usable. It is useful to provide data input in a clear and simple way across programming languages, and is widely used in applications that require a high level human-computer interaction, such as configuration files or data storage.

[37] For example, expressing the aggregate relationships in $f(\cdot)$ is, in its core, what the "model" block in Dynare's mod-file does.

During each Newton iteration we seek to solve

$$J(\mathbf{x}_i)(\mathbf{x}_{i+1} - \mathbf{x}_i) = -F(\mathbf{x}_i) = -\mathbf{z}_i. \tag{C.6}$$

Following the ideas of Laffargue (1990) and Juillard et al. (1996), this can be solved efficiently by bringing $J(\mathbf{x})$ in a block bidiagonal form

$$\hat{J} = \begin{bmatrix} I & M_1 & & & & \\ & I & M_2 & & & \\ & & I & M_3 & & \\ & & & \ddots & \ddots & \\ & & & & I & M_{T-2} \\ & & & & & I \end{bmatrix}. \tag{C.7}$$

This can be done by initializing $M_0 = \mathbf{0}$ and $\hat{z}_0 = \mathbf{0}$ and for $j \in 1, 2, \dots, T-1$ setting

$$K_j = f_B(x_j) - f_A(x_j)M_{j-1}, \tag{C.8}$$

$$M_j = K_j^{-1} f_C(x_j), \tag{C.9}$$

$$\hat{z}_j = K_j^{-1} z_j - f_A(x_j)\hat{z}_{j-1}, \tag{C.10}$$

which is equivalent to recursively solving for and subtracting each a pair of equations in $J(\mathbf{x})$. Each entry y_j of $\mathbf{y}_i = \mathbf{x}_{i+1} - \mathbf{x}_i$ can then simply be found via the recursion

$$y_j = \hat{z}_j - M_t y_{j+1}, \tag{C.11}$$

with $y_T = \mathbf{0}$.

```
# EGM stage: marginal values & decisions
decisions:
  inputs: [WaPrime,WbPrime]
  calls: |
    z_grid = income(skills_grid, tax, w, n, transfers)
    Psi = marginal_cost_grid(a_grid, Ra−1, psi_a0, psi_a1, psi_a2)
    WaPrimeExp = expect_transition(skills_transition, WaPrime)
    WbPrimeExp = expect_transition(skills_transition, WbPrime)
    Wa, Wb, a, b, c, uce = egm_step(WaPrimeExp, WbPrimeExp, a_grid, b_grid, z_grid, skills_grid, kappa_grid, \
       beta, sigma_c, Rb−1, Ra−1, psi_a0, psi_a1, psi_a2, Psi)
  outputs: [a,b,c,uce]

# intermediate stage: aggregation
aux_equations: |
    # calculate asset share of top−10%
    top10a = 1 - percentile(a, dist, .9)
    # aggregation
    UCE = sum(dist*uce, axis=(0,1,2))
    ...

# main stage: aggregate equations
equations:
    ~ psi_w*(piwn/piwntilde - 1)*piwn/piwntilde = wage_markup/(wage_markup−1)*chi*n**(1+sigma_l) + \
          1/(1−wage_markup)*(1 - tax)*w*n*UCE + \
          psi_w*beta*(piwnPrime/piwntildePrime - 1)*piwnPrime/piwntildePrime # wage Phillips curve
    ~ piwn = wn/wnLag*pi # wage inflation
    ~ w = max(iota*wLag/pi, wn) # dowards nominal wage rigidity

    ~ div = (1 - psi_p/2*(pi/pitilde - 1)**2)*y - w * n - i # dividends
    ~ Rb = Rr - zeta # real bond returns
    ~ Ra = assetshareLag * (div + equity) / equityLag + (1 - assetshareLag) * Rr # real asset returns
    ...
```

Fig. C.1. Part of the YAML-file which specifies the two-asset HANK from Section 3. The block `decisions` represents the function $W(\cdot)$ which depends on w_{t+1} (here: `WaPrime` and `WbPrime`) and aggregate variables such as wages `w`, labor hours `n`, and parameters such as σ_c (here `sigma_c`). The outputs are disaggregated savings a_{it} and b_{it}, consumption c_{it} and marginal utilities. The equations block shows the first aggregate equations starting with Eq. (73).

References

Achdou, Y., Han, J., Lasry, J.-M., Lions, P.-L., Moll, B., 2022. Income and wealth distribution in macroeconomics: a continuous-time approach. Rev. Econ. Stud. 89 (1), 45–86.

Ahn, S., Kaplan, G., Moll, B., Winberry, T., Wolf, C., 2018. When inequality matters for macro and macro matters for inequality. NBER Macroecon. Annu. 32 (1), 1–75.

Algan, Y., Allais, O., Den Haan, W.J., 2008. Solving heterogeneous-agent models with parameterized cross-sectional distributions. J. Econ. Dyn. Control 32 (3), 875–908.

Auclert, A., 2019. Monetary policy and the redistribution channel. Am. Econ. Rev. 109 (6), 2333–2367.

Auclert, A., Bardóczy, B., Rognlie, M., Straub, L., 2021. Using the sequence-space jacobian to solve and estimate heterogeneous-agent models. Econometrica 89 (5), 2375–2408.

Auclert, A., Rognlie, M., 2017. Aggregate demand and the top 1 percent. Am. Econ. Rev. 107 (5), 588–592.

Auclert, A., Rognlie, M., 2018. Inequality and Aggregate Demand. Technical Report. National Bureau of Economic Research.

Azinovic, M., Gaegauf, L., Scheidegger, S., 2022. Deep equilibrium nets. Int. Econ. Rev. (Philadelphia) 63 (4), 1471–1525.

Bayer, C., Born, B., Luetticke, R., 2020. Shocks, Frictions, and Inequality in US Business Cycles. CEPR Discussion Papers 14364.

Bayer, C., Born, B., Luetticke, R., 2023. The liquidity channel of fiscal policy. J. Monet. Econ. 134, 86–117.

Ben-Israel, A., 1965. A modified newton-raphson method for the solution of systems of equations. Israel J. Math. 3, 94–98.

Bianchi, F., Melosi, L., Rottner, M., 2021. Hitting the elusive inflation target. J. Monet. Econ. 124, 107–122.

Boehl, G., 2022. An Ensemble MCMC Sampler for Robust Bayesian Inference, Available at SSRN 4250395. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4250395.

Boppart, T., Krusell, P., Mitman, K., 2018. Exploiting MIT shocks in heterogeneous-agent economies: the impulse response as a numerical derivative. J. Econ. Dyna. Control 89, 68–92.

Calvo, G.A., 1983. Staggered prices in a utility-maximizing framework. J. Monet. Econ. 12 (3), 383–398.

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

De Ferra, S., Mitman, K., Romei, F., 2020. Household heterogeneity and the transmission of foreign shocks. J. Int. Econ. 124, 103303.

Fair, R.C., Taylor, J.B., 1980. Solution and Maximum Likelihood Estimation of Dynamic Nonlinear RationalExpectations Models. Technical Report. National Bureau of Economic Research.

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

Fernández-Villaverde, J., Marbet, J., Nuño, G., Rachedi, O., 2025. Inequality and the zero lower bound. J. Econom. 249, 105819.

Golosov, M., Lucas, R.E., 2007. Menu costs and phillips curves. Econometrica 75 (1), 1–35.

Gornemann, N., Kuester, K., Nakajima, M., 2016. Doves for the rich, hawks for the poor? Distributional consequences of monetary policy no. 089. ECONtribute Discussion Paper, 2021.

Greenwood, J., Hercowitz, Z., Huffman, G.W., 1988. Investment, capacity utilization, and the real business cycle. Am. Econ. Rev. 78 (3), 402–417.

Gust, C., Herbst, E., López-Salido, D., Smith, M.E., 2017. The empirical implications of the interest-rate lower bound. Am. Econ. Rev. 107 (7), 1971–2006.

Gust, C.J., Herbst, E.P., López-Salido, J.D., Smith, M.E., 2012. The Empirical Implications of the Interest-Rate Lower Bound. Technical Report. Board of Governors of the Federal Reserve System (US).

Hagedorn, M., Luo, J., Manovskii, I., Mitman, K., 2019. Forward guidance. J. Monet. Econ. 102, 1–23.

Hintermaier, T., Koeniger, W., 2010. The method of endogenous gridpoints with occasionally binding constraints among endogenous variables. J. Econ. Dyn. Control 34 (10), 2074–2088.

Juillard, M., Laxton, D., McAdam, P., Pioro, H., 1998. An algorithm competition: first-order iterations versus newton-based techniques. J. Econ. Dyn. Control 22 (8-9), 1291–1318.

Juillard, M., et al., 1996. Dynare: a Program for The Resolution and Simulation of Dynamic Models With Forward Variables Through the Use of a Relaxation Algorithm. Vol. 9602. Citeseer.

Kahou, M.E., Fernández-Villaverde, J., Perla, J., Sood, A., 2021. Exploiting Symmetry in High-Dimensional Dynamic Programming. Technical Report. National Bureau of Economic Research.

Kaplan, G., Moll, B., Violante, G.L., 2018. Monetary Policy According to HANK. NBER Working Papers 3. National Bureau of Economic Research, Inc. https://ideas.repec.org/p/nbr/nberwo/21897.html.

Khan, A., Thomas, J.K., 2013. Credit shocks and aggregate fluctuations in an economy with production heterogeneity. Am. Econ. Rev. 103 (1), 66–99.

Klenow, P.J., Kryvtsov, O., 2008. State-dependent or time-dependent pricing: does it matter for recent US inflation? Q. J. Econ. 123 (3), 863–904.

Krueger, D., Mitman, K., Perri, F., 2015. Macroeconomics and heterogeneity, including inequality. Handbook of Macroeconomics (forthcoming).

Laffargue, J.-P., 1990. Résolution d'un modèle macroéconomique avec anticipations rationnelles. Ann. Econ. Stat. (1986), 97–119.

Lin, A., Peruffo, M., 2024. Aggregate Uncertainty, HANK, and the ZLB. Working Paper Series 2911. European Central Bank.

Lindé, J., Trabandt, M., 2018. Should we use linearized models to calculate fiscal multipliers? J. Appl. Econometr. 33 (7), 937–965.

Maliar, L., Maliar, S., Winant, P., 2021. Deep learning for solving dynamic economic models. J. Monet. Econ. 122, 76–101.

McKay, A., Nakamura, E., Steinsson, J., 2016. The power of forward guidance revisited. Am. Econ. Rev. 106 (10), 3133–3158.

Mises, R.V., Pollaczek-Geiringer, H., 1929. Praktische verfahren der gleichungsauflösung. ZAMM-J. Appl. Math. Mech./Zeitschrift für Angewandte Mathematik und Mechanik 9 (1), 58–77.

Nakamura, E., Steinsson, J., 2010. Monetary non-neutrality in a multisector menu cost model. Q. J. Econ. 125 (3), 961–1013.

Petrosky-Nadeau, N., Zhang, L., Kuehn, L.-A., 2018. Endogenous disasters. Am. Econ. Rev. 108 (8), 2212–2245.

Reiter, M., 2009. Solving heterogeneous-agent models by projection and perturbation. J. Econ. Dyn. Control 33 (3), 649–665.

Reiter, M., 2023. State Reduction and Second-Order Perturbations of Heterogeneous Agent Models. Technical Report. IHS Working Paper.

Rotemberg, J.J., 1982. Monopolistic price adjustment and aggregate output. Rev. Econ. Stud. 49 (4), 517–531.

Saad, Y., Schultz, M.H., 1986. Gmres: a generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. 7 (3), 856–869.

Scheidegger, S., Bilionis, I., 2019. Machine learning for high-dimensional dynamic stochastic economies. J. Comput. Sci. 33, 68–82.

Smets, F., Wouters, R., 2007. Shocks and frictions in US business cycles: a Bayesian DSGE approach. Am. Econ. Rev. 97 (3), 586–606.

Van der Vorst, H.A., 1992. Bi-CGSTAB: a fast and smoothly converging variant of bi-CG for the solution of nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. 13 (2), 631–644.

Winberry, T., 2018. A method for solving and estimating heterogeneous agent macro models. Quant. Econom. 9 (3), 1123–1151.

Young, E.R., 2010. Solving the incomplete markets model with aggregate uncertainty using the krusell–smith algorithm and non-stochastic simulations. J. Econ. Dyn. Control 34 (1), 36–41.