An Ensemble MCMC Sampler for Robust Bayesian Inference

Gregor Boehl

University of Bonn

Abstract

This paper proposes a Differential-Independence Mixture Ensemble (DIME) sampler for the Bayesian estimation of structural models. DIME allows the estimation of models that are computationally expensive to evaluate with challenging, multimodal, high-dimensional posterior distributions and ex-ante unknown properties. It combines the advantages of gradient-free global multi-start optimizers with the properties of Monte Carlo Markov chains to quickly explore the typical set. DIME is used to estimate a two-asset heterogeneous agent New Keynesian ("HANK") model, for the first time including the households' preference parameters. The results point towards a less accentuated role of household heterogeneity for the empirical macroeconomic dynamics.

Keywords: Bayesian Estimation, Monte Carlo Methods, Heterogeneous Agents, Global Optimization
JEL: C11, C13, C15, E10

1 Introduction

Bayesian methods are used ubiquitously in all fields of economics since the pioneering work of Geweke (1999) and Schorfheide (2000). They are an essential tool to bring complex structural models to the data, such as modern New Keynesian-type DSGE models or Bayesian vector autoregression models. They allow for the evaluation and comparison of the empirical performance of these models and the quantification of the effects of potential policy actions. They provide a framework for incorporating prior beliefs and new data in a coherent way, and allow for the explicit modeling of uncertainty, which is important for many economic applications.

Yet, despite their powerfulness, the application of Bayesian methods can in practice be quite challenging for two reasons: first, as the degree of complexity in contemporary economic models increases, so do the computational costs of solving these models. Consequently, the number of model evaluations required by Bayesian estimations may be prohibitively large. Second, Bayesian inference requires the *identification* of and, subsequently, *sampling from the typical set* of the parameter posterior distribution associated with these

⋆Address: Institute for Macroeconomics and Econometrics, University of Bonn, Adenauerallee 24-42, 53113 Bonn, Germany. I am grateful to Christian Bayer, Flora Budianto, Cees Diks, Keith Kuester, Alexander Meyer-Gohde, Frank Schorfheide, Felix Strobel and participants of several conferences and seminars for discussions and helpful comments on the contents of this paper. Part of the research leading to the results in this paper has received financial support from the Alfred P. Sloan Foundation under the grant agreement G-2016-7176 for the Macroeconomic Model Comparison Initiative (MMCI) at the Institute for Monetary and Financial Stability. I also gratefully acknowledge financial support by the Deutsche Forschungsgemeinschaft (DFG) under CRC-TR 224 (projects C01 and C05) and under project number 441540692.

Email address: gboehl@uni-bonn.de

models.[1] This posterior distribution usually is a highly complex topology with large dimensionality and ex-ante unknown properties. Conventional Markov chain Monte Carlo (MCMC) samplers and optimization tools often do not perform well on such distributions – while theoretical results show convergence as the number of iterations goes to infinity, convergence is in practice not achived in finite time. Yet, the quality and reliability of an estimation – and thereby their usefulness for economic analysis – crucially depend on our ability to pin-down the posterior distribution precisely.

This paper substantiates this ability and expands the set of models feasible for estimation by introducing a novel sampling approach: the *differential-independence mixture ensemble* (DIME) Markov chain Monte Carlo method. In its core, the algorithm combines two novel concepts from the field of computational astrophysics: the *differential evolution* method and *independence sampling*. While both methods come with substantial weaknesses in practice, the DIME sampler exploits their complementaries while chancelling out the individual shortcomings of the two methods. This creates a powerful tool to satisfy four central practical requirements:

i) Good performance for high-demensional, multimodal and complex distributions.[2]
ii) Fast burn-in to the typical set of the posterior absent any prior posterior mode density optimization or informative initial guesses.
iii) The speed of convergence scales well with the number of chains, allowing for the efficient use of parallelization.
iv) The proposal distribution is generated endogenously from the current state of all chains, allowing to sample from posterior distributions with ex-ante unknown properties.

Point ii) is in particular desirable to overcome the common practice to treat the *identification* of the typical set of the posterior separately from the problem of actual *sampling* from it. The former is often done using numerical optimization routines for mode finding, which may also consume a significant amount of processing time. The problem is that the posterior of DSGE models is often not only high-dimensional, but may also be discontinuous and feature many local maxima.[3] Numerical optimizers tend to behave unstable for such target functions, can show strong dependence on initial guesses, and may, if at all, only converge to a local maximum that is rather distant from the typical set. Such bad starting points in turn tend to cause malperformance of the actual sampling algorithm.

Point iii) is important because the progressive evolution of structural models makes them more and more expensive to evaluate, e.g. because they include severe nonlinearities or because agents are heterogeneous across multiple dimensions.[4] As multi-core architectures are widespread nowadays and machines with larger number of processors are successively becoming more affordable, researchers want to take advantage of this development by being able to effectively run estimations in parallel, thereby significantly reducing total runtimes.

To meet these requirements, instead of using a single or small number of recursive chains (such as e.g. the random-walk Metropolis algorithm), DIME relies on an *ensemble* of a large number of chains which

[1] The *typical set* is an important concept in information theory. It can roughly be defined as the central log density band into which almost all random draws from that distribution will fall (Betancourt, 2017; Carpenter et al., 2017).

[2] The term *multimodal* here in a broader sense, also refers to distributions for which the typical set is disjunct or exhibits discontinuities.

[3] This may, e.g., be due to the model's various cross-equation restrictions, a misalignment of prior and likelihood distributions. or issues with indeterminacy. A forward looking model is *indeterminate* if there exists no unique rational expectations solution. An additional problem with nonlinear models is that nonlinear Bayesian filtering is usually also based on sampling, leading to possibly noisy likelihood estimates.

[4] See, e.g. Boehl and Strobel (2020), for the estimation of medium scale DSGE models with the zero-lower bound on nominal interest rates as an example for nonlinear estimation, or Bayer et al. (2020) for the estimation of heterogeneous agent models.

jointly evolve over time. For each iteration, proposals are generated based on the current state of the full ensemble and, as the ensemble successively converges, proposal steps naturally adapt direction and scale of the estimated posterior distribution. After convergence, the invariant distribution of *all* chains corresponds to the target distribution. DIME is mixing between a local and a global transition kernel: the local kernel explores the direct proximity of one particular chain. The global kernel, in contrast, reshuffles chains over the complete domain of the current approximation of the posterior distribution. This means that DIME MCMC is equally efficient in converging quickly to the high-density region of the posterior (called *burn-in*) and for posterior sampling, and makes no difference between these two stages. The sampler can hence be seen as a "Swiss Army knife" for structural econometric analysis.

The local transition kernel builds on the *differential evolution* (DE) concept developed in the literature on global optimization. DE optimizes a function by maintaining a population of candidate solutions and creating new candidate solutions by combining existing ones, and then keeping whichever candidate solution has the best fitness on the optimization problem at hand. This can be turned into an MCMC method by exchanging the last step by the Metropolis-Hastings algorithm.[5] A major problem with this MCMC version of DE is that, although proposals are state-dependent and adaptive, all chains evolve ex-ante independently. This frequently causes the dispersion among chains to increase over time, which deteriorates the quality of proposals, thereby inducing slow convergence of the full ensemble overall. DE-MCMC also does not perform well with multimodal distributions because chains are unlikely to switch modes.

In contrast to the local kernel, at the core of the *global transition kernel* lies an ensemble version of a modified adaptive Independence Metropolis-Hastings method, where candidates are created based on a proposal distribution that is independent of the state of a single chain.[6] This attribute makes the global transition kernel fully robust against odd-shaped and multimodal distributions. However, independence Metropolis-Hastings performs only well if the proposal distribution is stationary and close to the target distribution. Since it is almost impossible meet this requirement ex-ante – the target distribution usually is a black box –, the algorithm had limited practical appeal. To circumvent this problem, this paper develops a time-varying proposal distribution that adjusts to new ensembles based on their average posterior density. This puts decaying weights on early samples but guarantees convergence to a stationary proposal distribution once the average density of candidates converges. While this improves performance considerably, convergence of the global transition kernel *alone* is still slow.

DIME MCMC exploits the complementarity of the local and global transition kernel: the combination of the two kernel dispels the individual weaknesses. In a mixture, the global kernel occasionally reshuffles some of the chains, which counteracts dispersion of the ensemble and ensures that individual chains do not "get stuck" in local maxima. This, in turn, also increases the quality of the proposals from the local transition kernel. The independent proposals from the global transition kernel also make sure that chains switch between modes for multimodal distributions. The local kernel generates good proposal candidates during burn in, which provides updates for the proposal distribution of the global transition kernel. Its differential evolution heritage further allows DIME to be applied on target distributions (or objective functions) that are discontinuous or noisy. Thus, DIME has some similarities to multi-start optimizers and searches the complete relevant function domain. It hence combines the advantages of a broad class of global optimizers with the properties of a Markov chain Monte Carlo (MCMC) sampler, satisfying requirements i) and ii).

The ensemble structure makes DIME MCMC "embarrassingly parallelizable", which addresses point iii).

[5] See Storn and Price (1997) for the global optimizer and Ter Braak (2006); Nelson et al. (2013) for Differential Evolution MCMC.

[6] For independence Metropolis-Hastings see, e.g., Tierney (1994). For specifications of adaptive independence Metropolis-Hastings see for example Haario et al. (2001) and Roberts and Rosenthal (2007). A similar algorithm from the global optimization literature is the covariance matrix adaptation evolution strategy (CMA-ES, Igel et al., 2007).

As I show in Section 4, the sampler scales well in terms of the quality of proposals (which increases with the number of chains) and the number of iterations (lesser chains increase convergence rate per function evaluation). The method is essentially self-tuning and only requires setting the number of chains as the only necessary metaparameter, thus satisfying requirement iv).[7]

Additionally and independently from the core methodological contribution outlined above, I introduce a bijective mapping between parameter and proposal space which ensures that the proposal distribution respects the support of the prior distribution. This allows the sampling algorithm to run in unbounded space, which helps to avoid large rejection rates due to draws falling outside the prior support and, thereby, improves sampling efficiency. This is similar to the so-called *bijectors* which are applied in the literature on neural networks (e.g., Dillon et al., 2017).[8]

Although DIME MCMC is straightforward to implement, this paper comes with reference implementations in Python and Julia programming languages, and for matlab. The implementations for Python and Julia can directly be installed through the official software repositories and are actively developed at Github. The Python package integrates into the established emcee-package, which is a collection of (ensemble) MCMC samplers (Foreman-Mackey et al., 2013).[9]

I assay the performance of DIME MCMC on three distinct but important use cases. To start with, I evaluate the algorithm's capability to deal with high-dimensional and bimodal distributions with ex-ante known properties. I document that the sampler performs well on such distributions, even when the two modes (and thus the typical set) are fully disconnected. I then test the performance of the sampler on the estimation exercise from Smets and Wouters (2007). DIME MCMC returns the original parameter estimates independently of the number of chains used. For the given example, convergence times scale well with the number of chains, which suggests that the losses through parallelization are limited to the computational overhead of serialization.

Finally, I estimate a heterogeneous agents New Keynesian model, including the households preference parameters. These parameters may be of particular relevance on their own as they govern the economy's steady state distribution of assets. This exercise was so far deemed impossible due to the large computational costs associated with solving for the steady state distribution for each single likelihood evaluation of the model, and is only enabled by the fact that DIME MCMC is trivial to parallelize. The estimation results point towards a rather attenuated role of portfolio choice for macroeconomic dynamics, with the parameter that determines the magnitude of the liquidity friction being identified significantly below its prior mean. The degree of idiosyncratic income risk is also estimated to be below its prior mean, but still in the range of values used in the literature.

Literature

The workhorse of Bayesian estimations in many economic applications is the random walk Metropolis Hastings (RWMH) algorithm, which dates back to the seminal work of Metropolis et al. (1953) and Hastings (1970). The shortcomings of RWMH are well documented (e.g. Chib and Ramamurthy, 2010; Herbst and Schorfheide, 2015; Betancourt, 2017). The main issue is that convergence of RWMH to the posterior distribution can be extremely slow, and sampling from ill-shaped or multimodal distributions is hardly possible in

[7] Sections 3 and 4 discuss the role of the number of chains and of the kernel mixing probability, and provide sane defaults for these parameters.

[8] In the practical application of neural networks, bijectors are used to create proxy-posteriors which feature a more favorable geometry.

[9] Documentation and downloads for the Python package can be found at https://github.com/gboehl/dime_sampler. The standalone version in Julia programming language is located at https://github.com/gboehl/DIMESampler.jl. The matlab implementation can be found at https://github.com/gboehl/dime-mcmc-matlab.

practice. To circumvent the first problem, numerical optimization routines are frequently used to find good initialization values for RWMH. These routines are, however, often slow as well, and not very robust when applied to more complicated posterior distributions. In particular, they tend to "get stuck" at local maxima. Another problem with RWMH as well as with most numerical optimizers is that they are not parallelizable due to their recursive nature. They therefore can not benefit from multi-core architectures, which is a major drawback if the posterior density is computationally expensive to evaluate.

Well-known alternatives to RWMH include Gibbs and slice sampling (Geman and Geman, 1984; Damlen et al., 1999), which perform better on high-dimensional distributions. They are, however, not robust to multimodal distributions and do not perform well for burn-in and convergence to the high probability density region. Also, these methods can not trivially be parallelized. A recent innovation from the econometrics community is the sequential Monte Carlo (SMC) method introduced in Herbst and Schorfheide (2014). The core idea is to run many RWMH chains in parallel interrupted by occasional resampling stages to ensure that all chains converge to the high probability density region. In order to prevent convergence to local optima, the authors develop a tempering scheme for SMC. By construction, this circumvents many of the shortcomings of standard RWMH and, given the right choice of a tempering scheme, can also perform well on multimodal distributions. SMC is also reported to work well on vector autoregression (VAR) models (Bognanni and Herbst, 2018). The combination of tempering with RWMH chains may have the disadvantage of relatively slow convergence. Additionally, the method has relatively many degrees of freedom in the choice of metaparameters, which may determine overall performance. In contrast, the proposal density of DIME is endogenous and, through the adaptation extensions, chains converge more quickly. As SMC, DIME can straightforwardly be applied to VAR models.

Research in the field of astrophysics has recently made considerable progress on the frontier of Monte Carlo sampling. Ensemble MCMC is conceptionally introduced by Goodman and Weare (2010). The authors develop the idea of an ensemble of Markov chains which, based on the current state of all chains, generates proposals inspired by the numerical optimization method of Nelder and Mead (1965). They show that such sampler is affine invariant and "uniformly effective over all the convex bodies of a given dimension regardless of their shape", thereby significantly outperforming RWMH. The success of Ensemble MCMC methods is accelerated by its excellent implementation in the open source packet *emcee* (Foreman-Mackey et al., 2013).[10] As shown in Section 4, Goodman and Weare (2010) indeed performs well in terms of sampling efficiency but, at least for the models considered here, is rather slow to converge to the posterior distribution. As acknowledged by the authors, the method by construction does not perform well for multimodal distributions.

Vrugt et al. (2009) identify similar core problems with DE-MCMC (Ter Braak, 2006) as those that are documented here, and propose a series of workarounds. As such, a common problem with multi-chain methods (like DE-MCMC) is that when considering interacting vectors, the entire ensemble has to be considered as a whole, which increases n-fold the dimension of the target and may thus significantly impact convergence. To address this problem, the authors add a scheme to resample DE-MCMC chains that are stuck, which disburdens the problem of overly dispersed chains.[11] However, the proposed heuristic for outlier detection may not work well for all distributions in practice. To allow better support for multimodal distributions the authors add DE-MCMC proposals for which the jump distance is unity. While this is a practical workaround, it is likely to also slow down convergence, in particular for more challenging distributions. A nice addition is a crossover step to decrease autocorrelation similar to the *snooker move* introduced in

[10] Emcee is implemented in the Python language and can be found at https://github.com/dfm/emcee. The package also provides routines for efficient parallelization.

[11] The necessity to replace malperforming chains is an important issue that is also pointed out in ter Braak and Vrugt (2008).

ter Braak and Vrugt (2008). An extension of DIME along these lines indeed increases convergence speed and decreases autocorrelation times, but comes at the expense of not performing reliably on multimodal distributions.

The recent rise of frameworks allowing for *automatic differentiation* (AD)[12] has renewed interest in the Hamiltonian Monte Carlo (HMC) method (Duane et al., 1987; Childers et al., 2022). HMC proposals are based on the Jacobian of the posterior distribution. While these are normally expensive to evaluate (e.g. via finite difference methods), AD provides computationally more efficient means for their calculation. HMC clearly outperforms RWMH (and many other competitors) in terms of sampling efficiency and in its capability to sample from more complex distributions. Drawbacks of HMC are that it does not necessary provide fast burn-in and does not perform well for multimodal distributions if the modes are sufficiently disconnected. HMC is also recursive by nature, which prohibits efficient parallelization. Importantly, HMC requires the implementation of the likelihood function – and hence the complete structural model – in a framework that allows for AD, which may require a major programming effort. Note that even with AD the evaluation of the Jacobian is significant more costly than a standard likelihood evaluation. Section 6 briefly touches upon a mixture sampler of DIME with the HMC method.[13]

The rest of the paper is structured as follows. Section 2 explains the basic DIME algorithm. Section 3 studies the performance of the algorithm on a high dimensional bimodal distribution. In Section 4 the sampler is used on the Smets-Wouters model and in Section 5 it is applied to the estimation of a large-scale HANK model. Section 6 concludes.

2 Mixture Ensemble MCMC Sampling

Let $\pi(x)$ be the probability density of a target distribution with $x \in \mathbb{R}^n$. In practice, $\pi(x)$ is often the posterior density $\pi(x) = p(x|Y)$, which for given data Y and model x equals

$$p(x|Y) = \frac{p(Y|x)p(x)}{p(Y)}. \tag{1}$$

$p(Y|x)$ is the likelihood which, provided (x, Y), can be calculated using various Bayesian filtering techniques. For many use cases the evaluation of $p(Y|x)$ is computationally expensive. Let me assume that the prior $p(x)$ is specified such that it is straightforward to evaluate and to sample from, and

$$p(Y) = \int p(Y|x)p(x)dx \tag{2}$$

is an unknown constant for given data Y. We then wish to draw a sufficiently large number of samples from π in order to approximate some quantity

$$E_\pi\left[h(x)\right] = \int h(x)\pi(x)dx \approx \frac{1}{N}\sum_j h(x_j) \tag{3}$$

with as few likelihood evaluations as possible.

[12] AD is e.g. available through the Python packages *JAX* or *TensorFlow*, or in the new *Julia* programming language. Boehl (2023) contains a primer on AD and also shows how it can be used to solve heterogeneous agent models. A descendant of HMC is implemented in the well-known *STAN* framework (Carpenter et al., 2017).

[13] Other than DIME, HMC requires that the parameter space is continuous (c.f. Neal et al., 2011), which is generally not the case for DSGE models due to parameter combinations for which the model is indetermined or explosive.

2.1 DIME vs. Random Walk Metropolis-Hastings

As a reference point, let me briefly sketch the classic random walk Metropolis-Hastings algorithm (Hastings, 1970, RWMH). Start with a *single* parameter vector X_i at iteration i. A new replacement candidate is generated by $\hat{X}_i = X_i + \varepsilon_i$ where $\varepsilon_i \sim \mathcal{N}(0, \Sigma)$ is called the RMWH *proposal distribution*, which is often assumed to follow a multivariate normal distribution. The replacement candidate $\hat{X}_i$ is accepted with the Metropolis acceptance probability

$$P(X_{i+1} = \hat{X}_h) = \min\left\{1, \frac{\pi(\hat{X}_h)}{\pi(X_h)}\right\}. \tag{4}$$

If it is accepted, set $X_{i+1} = \hat{X}_i$. Otherwise, set $X_{i+1} = X_i$. A large literature discusses the properties of RWMH, see e.g. Sokal (1997) or Roberts and Rosenthal (2001).

The practical performance of the algorithm crucially depends on the choice of the proposal distribution, i.e. here on the covariance matrix Σ. This may be problematic since Σ has $\frac{d(d+1)}{2}$ degrees of freedom and it is challenging to determine ex-ante which choice of Σ will maintain a high acceptance ratio while still exploring the target distribution to a satisfactory degree. To maintain a sufficiently large acceptance ratio, Σ is often scaled down to relatively small values. Consequently, RWMH is very slow to converge to the high probability density region of the posterior (so-called *burn-in* or *thermalization*). To speed up computation, RWMH is thus often used subsequent to a numerical optimization routine, which is supposed to provide better starting values. As discussed above, such numerical optimization routines may also have severe limitations.

DIME MCMC uses a different approach. It combines the characteristics of a broad class of global optimizers with the properties of a MCMC sampler. The first feature is that the sampler draws proposals from a local transition kernel – a replacement candidate that, locally for each individual chain, is created relative to its previous state – as well as global proposals that are independent of the state of a single chain. Both proposal kernels adapt to the state of the complete ensemble, explicitly for the global transition kernel and implicitly for the local transition kernel. The coexistence of local and global transition kernel prevents single chains from "getting stuck" at local maxima, speeds up convergence, and eases sampling from distributions with two or more modes, even if these are fully separated. The second feature is the separation of proposal space from parameter space, which ensures that any proposed replacement candidate has a positive prior probability. This increases acceptance rates notably.

2.2 The Ensemble

In the spirit of Goodman and Weare (2010) consider an *ensemble*

$$\mathbf{X}_i = (X_{i,1}, \ldots, X_{i,n_c}), \tag{5}$$

of n_c individual *chains* $X_{i,j}$ (or *particles*, in SMC terminology) indexed by $j = 1, 2, \ldots, n_c$ running in parallel at each iteration i. As in Herbst and Schorfheide (2014), initialize the ensemble with n_c draws from the prior distribution

$$\mathbf{X}_0 \overset{n_c}{\sim} p(x). \tag{6}$$

Initializing the ensemble with the prior distribution ensures that the full set of prior information on the relevant parameter space is considered, independently of potential multimodality or possible discontinuities.[14]

[14] Goodman and Weare (2010) suggest to initialize the ensemble as a small ball around some initial value. However, if the posterior is oddly shaped – e.g., if it is bimodal –, this bears the risk that the ensemble can not fully unfold. It may also be unclear which initial

2.3 Strategy mixture

In each iteration and for each chain the sampler draws proposals that are mixtures of a local and a global transition kernel.[15] Let the global proposal kernel be selected with probability χ and, respectively, the local transition kernel be chosen with probability $1 - \chi$.

Each iteration i of a DIME MCMC run then comprises:

1. Update the proposal distribution for the global transition kernel based on $\mathbf{X}_i$.
2. To each chain i randomly assign a transition kernel $K_{i,j} \in \{G, L\}$ with probabilities $(\chi, 1 - \chi)$.
3. For each chain i, propose a replacement candidate vector $\hat{X}_{i,j}$ based on the assigned transition kernel.
4. For each chain i calculate the factor weight $\omega_{i,j}$.
5. For each chain i, evaluate the posterior density $\pi(\hat{X}_{i,j})$ of the candidate.
6. For each chain i, generate $X_{i+1,j}$ by accepting $\hat{X}_{i,j}$ with a Metropolis acceptance probability of

$$P(X_{i+1,j} = \hat{X}_{i,j}) = \min\left\{1, \frac{\pi(\hat{X}_{i,j})}{\pi(X_{i,j})}\omega_{i,j}\right\}, \qquad i = 1, 2, \ldots, n_c, \tag{7}$$

or reject $\hat{X}_{i,j}$ and set $X_{i+1,j} = X_{i,j}$ with probability

$$P(X_{i+1,j} = X_{i,j}) = 1 - P(X_{i+1,j} = \hat{X}_{i,j}). \tag{8}$$

The theoretical properties of the mixture of two or more kernels are fairly well understood. Proposition 1 summarizes the central result on the ergodicity and convergence of kernel mixtures.

Proposition 1. *Suppose two transition kernels P_1 and P_2 have invariant distribution π and P_1 is uniformly ergodic. Then for $0 < \chi < 1$ the kernel $\chi P_1 + (1 - \chi)P_2$ is uniformly ergodic with invariant distribution π.*

Proof. The proof is provided by Tierney (1994) following Propositions 3 and 4.

The remainder of this section goes through the central components – local and global kernel, and factor weights – in detail. Section 4 investigates the question of the optimal number of chains n_c contra the number of iterations in detail. As documented there, a value of $n_c \in (4n, 6n)$ is often a good choice, where larger ensembles help to tackle more irregular posterior distributions, as e.g. bimodal distributions, but fewer chains may speed up burn-in.

2.4 Local transition kernel

The local kernel is *local* in the sense that for each chain j to which the local transition kernel is assigned, the candidate is proposed relative to the current state $X_{i,j}$ of j. Yet, the relative increment is based on the state of the full ensemble. At its core, the random-walk proposal distribution of RWMH is replaced with a proposal that follows the *differential evolution* concept of Ter Braak (2006).

More formally, for each iteration i and each chain $j : K_{i,j} = L$ draw two reference chains $\{k, l\} \in \{1, 2, \ldots, n_c\}$ with $k \neq j$ and $l \neq k \wedge l \neq j$. Take the difference of the state of these two chains as a

value to choose, in particular if we seek to avoid nonlinear optimization routines.

[15] See Tierney (1994) for a theoretical discussion of mixture kernels. An alternative approach would be to draw the transition kernel not *per chain and iteration* but *for all chains per iteration*.

displacement vector which is added to the state $X_{i,j}$ of chain j. The replacement candidate for chain $X_{i,j}$ is then $\hat{X}_{i,j} = f_b\left(\hat{X}_{i,j}\right)$ with

$$\hat{X}_{i,j} = X_{i,j} + \gamma(X_{i,k} - X_{i,l}) + \epsilon_{i,j}, \qquad \forall j \in \left\{j : K_{i,j} = L\right\} \tag{9}$$

where γ is a scaling factor and $\epsilon_{i,j}$ is some (very) small noise.

As the ensemble evolves over time, proposal steps naturally adapt direction and scale of the current estimate of the posterior distribution. When the ensemble converges to the posterior distribution, so does the proposal distribution. Note that probability to draw the displacement vector $X_{i,k} - X_{i,l}$ is exactly equal to drawing the displacement vector $X_{i,l} - X_{i,k}$. Denoting the respective Metropolis-Hastings proposal distribution by g it thus holds that

$$g\left(\hat{X}_{i,j}|X_{i,j}\right) = g\left(X_{i,j}|\hat{X}_{i,j}\right) \tag{10}$$

and Equation (8) implies *detailed balance* for $\omega_{i,j} = 1$.[16]

Proposition 2. *The local transition kernel L yields an ensemble of Markov chains* $\mathbb{X}_i = \{X_{i,1}, \dots X_{i,n_c}\}$ *with* π *as the unique stationary distribution of each* $X_{i,j} \in \mathbb{X}_i$.

Proof. The local transition kernel concurs with the differential evolution kernel introduced in Ter Braak (2006). The proof of ergodicity to π is provided in ter Braak and Vrugt (2008).[17] Intuitively, given the stationary distribution, the proposal distribution is the γ-scaled difference of two draws from the posterior distribution, which by itself is a stationary and symmetric distribution. This result on $\mathbb{X}$ applies one-to-one to the stationary distribution of $\mathbb{X}$.

From the intuition of the proof it also follows that, if $\pi(x)$ follows a Gaussian distribution, after convergence each individual proposal $\hat{X}_{i,j}$ is of the same form as an RWMH proposal. This can be verified by acknowledging that, if $\pi(x)$ is Gaussian, each draw $X_{i,j}$ is also Gaussian, and the difference between two chains hence also follows a Gaussian distribution. Under the assumption that the target distribution is near-Gaussian, the optimal choice for the scale γ is $\gamma = \frac{2.38}{\sqrt{2n}}$ from the RWMH literature (e.g. Roberts and Rosenthal, 2001), which is expected to give an acceptance probability of 23% for high-dimensional posteriors, i.e. for large n. Throughout this paper I set γ to this default value. The additional $\epsilon_{i,j}$ follows a normal distribution with mean zero and a standard deviation of 1e−5.

2.5 Global proposal kernel

In contrast to the local transition kernel, the global proposal is *global* in so far as candidate proposals only depend on the global state of the ensemble (and its history), but not directly on the current state of a single chain.

For each chain j in iteration i with $K_{i,j} = G$ the displacement vector is drawn from an independent but adaptive proposal distribution. The distribution adapts such that it roughly corresponds to the current estimate of the posterior. A natural choice for such proposal distribution is the multivariate t-distribution with fixed degrees of freedom ν. This distribution is especially useful because for $\nu > 2$ it can be parameterized over its mean and covariance (μ_i, Σ_i), and exhibits fat tails.[18] Thus, let

$$\hat{X}_{i,j} \sim t_\nu\left(\mu_i, \Sigma_i\right), \tag{11}$$

[16] Detailed balance is a central feature of Markov chains which guarantees that each chain satisfies reversibility.

[17] As pointed out in ter Braak and Vrugt (2008), the original proof in Ter Braak (2006) contained an error.

[18] The benefits of a fat tailed proposal distribution for adaptive independence MCMC is also pointed out by Holden et al. (2009).

and, to again satisfy *detailed balance* in (8), set

$$\omega_{i,j} = \frac{f^t(X_{i,j})}{f^t(\hat{X}_{i,j})}, \tag{12}$$

where f^t is the density function of the multivariate t-distribution as defined in (14).

In each iteration i the ensemble $\mathbf{X}_i$ is used to update the parameters (μ_i, Σ_i) with weights proportional to the average posterior density of $\mathbf{X}_i$. This choice of weights ensures flexibility of the proposal distribution during burn-in but also stationarity after convergence. More formally, define the *absolute weight* of the ensemble $\mathbf{X}_i$ in iteration i on the proposal distribution as

$$w_i = a_i \sum_j^{n_c} \pi(X_{i,j}), \tag{13}$$

where $a_i = \frac{1}{n_c} \sum_j^{n_c} \mathbb{1}_{\{X_{i,j} \neq \hat{X}_{i-1,j}\}} \left(X_{i,j}\right)$ is the mean acceptance ratio in i. Let W_i be the *cumulative weight* in i initialized with $W_0 = 0$. Denote by $(\mu_i^{\mathbf{X}}, \Sigma_i^{\mathbf{X}})$ the sample mean and sample covariance matrix of the current ensemble $\mathbf{X}_i$. Then for each chain $j : K_{i,j} = G$ the proposal is given by

$$\hat{X}_{i,j} \sim \ t_\nu \left(\mu_i, \frac{\nu - 2}{\nu} \Sigma_i\right), \tag{14}$$

with

$$\mu_i = \left(\frac{W_{i-1}}{W_i}\right) \mu_{i-1} + \left(\frac{w_i}{W_i}\right) \mu_i^{\mathbf{X}}, \tag{15}$$

$$\Sigma_i = \left(\frac{W_{i-1}}{W_i}\right) \Sigma_{i-1} + \left(\frac{w_i}{W_i}\right) \Sigma_i^{\mathbf{X}}, \tag{16}$$

$$W_i = W_{i-1} + w_i. \tag{17}$$

Note that the proposal density is (almost) independent of chain j.

The weighted updating of draws from each new iteration has the strong advantage that during burn-in, newer updates have more weight than old ones and the proposal distribution adapts quickly to the current shape of the estimated target distribution. However, once the chains converge we have, for sufficiently large n_c, that $\sum_j^{n_c} \pi(X_{i,j}) \approx \sum_j^{n_c} \pi(X_{i+s,j})$ for $s = 1, 2, \ldots$ and more recent draws have decaying weights.[19]

Proposition 3. *The Global transition kernel G yields an ensemble of Markov chains* $\mathbb{X}_i = \{X_{i,1}, \ldots X_{i,n_c}\}$ *with* π *as the unique stationary distribution of each* $X_{i,j} \in \mathbb{X}_i$.

Proof. See Roberts and Rosenthal (2007) for a detailed proof of adaptive independence Metropolis Hastings. Since weights on most recent iterations are decaying, the procedure converges to the specification in Haario et al. (2001) and enjoys the same convergence properties therein.

A natural choice for the degrees of freedom ν of the multivariate t distribution is to pick rather low values, which imply fatter tails of the proposal distribution. All results of this paper are rather insensitive

[19] Note that it is not an actual requirement that the cumulative density is approximately equal across ensembles. Σ_i converges even if $\sum_j^{n_c} \pi(X_{i,j})$ varies a lot as long as it is stationary.

to the choice of ν, and throughout the following sections I use $\nu = 10$. While the above specification of the mean/covariance updating could be tweaked with a number of additional parameters (e.g. a tempering scheme for the probability weights), this is unnecessary in practice. If a researcher wishes to replace *less* chains per iteration, it is sufficient to decrease the probability χ for the global transition kernel and vice versa. This leaves the specification of the global transition kernel essentially parameter free, and in total requires the researcher to only specify n_c and χ as the necessary parameters for DIME MCMC. Throughout this paper I set $\chi = 0.1$, which provides a good compromise between fast burn-in and robustness for multimodal distributions. As discussed in section 4 a value of n_c between $4n$ and $6n$ often delivers good performance.

2.6 Optional: Proposal space vs. parameter space

The prior distribution $p(x)$ often has bounded support. Naturally, replacement candidates beyond these bounds are always rejected. This is in particular problematic for medium- and large-scale DSGE models as these frequently feature exogenous AR(1) processes with roots close to a unity. Since the prior of these roots is bounded by $(0, 1)$, estimates close to unit roots will often cause poor sampling performance because any proposal with values of the AR-coefficient larger one will be rejected. A model with several AR(1) processes close to unit roots will hence feature a rather low acceptance fraction during MCMC sampling.

To circumvent this problem, the above method can be extended by defining the *parameter space* $\mathbb{Z} : x \in \mathbb{Z} \Leftrightarrow p(x) > 0$ to be the space of all parameter combinations for which the prior density is positive. Redefine $\mathbb{X}$ from above as the *proposal space* $\mathbb{X} = \mathbb{R}^n$ which is unbounded, and let f_b be a *bijective map*

$$f_b : \mathbb{R}^n \to \mathbb{Z} \tag{18}$$

such that for any $x \in \mathbb{Z}$ there exists a unique $z \in \mathbb{R}^n$ for which $f_b(z) = x$. f_b then is always uniquely invertible, and by definition, f_b maps within the bounds of the prior distribution whereas its domain is unbounded. f_b ensures that every sample has a positive prior density.

While $\mathbf{Z}_i$ holds the ensemble in parameter space, let

$$\mathbf{X}_i = (X_{i,1}, \dots, X_{i,n_c}) = (f_b^{-1}(X_{i,1}), \dots, f_b^{-1}(X_{i,n_c})) \tag{19}$$

be its complementary representation in proposal space.

A straightforward choice for the functional form of the bijective transform f_b is to chose $x_q = \exp(z_q) + \underline{b}$ for priors that are bounded below by $\underline{b}$ (e.g. following a gamma and inverse gamma distribution), and the logistic function $x_q = \frac{\bar{b}-\underline{b}}{1+\exp(-z_q)} + \underline{b}$ for priors with two-sided bounds $(\underline{b}, \bar{b})$ (e.g. the beta distribution).[20] While the bijective mapping is used throughout this paper, all presented results also hold without bijection. This comes at the cost of moderately larger rejection rates and, thus, slightly slower rates of convergence.

3 A high-dimensional bimodal toy distribution

This section studies the performance of DIME MCMC on a distribution with known properties. I focus on a class of high dimensional bimodal distributions where the two modes may be disconnected and can have different density masses. Such distributions are known to be challenging for MCMC samplers: while it can shown theoretically that single Monte Carlo methods such as RWMH will converge to the posterior

[20] Another natural choice would be $x_q = \Phi^{-1}\left(F^q\left(z_q\right)\right)$, where Φ^{-1} is the quantile function of the standard normal distribution and F^q is the CDF of prior q. This effectively transforms the prior to a multivariate Gaussian, which may be beneficial for the approximation of the proposal distribution of the global transition kernel.

distribution almost surely if the number of iterations goes to infinity, it is well known that they often fail to do so in *finite* time. Rather, they tend to "get stuck" in one of the modes, thereby misrepresenting the true posterior. This makes this exercise a veritable challenge for DIME.

The probability density of the random variable M is given by the multivariate Gaussian mixture

$$\pi_M(x) = \lambda P(X = x) + (1 - \lambda)P(Y = x) \tag{20}$$

where $X \sim \mathcal{N}_n(\mu_X, \sigma I_n)$ and $Y \sim \mathcal{N}_n(\mu_Y, \sigma I_n)$ are both n-dimensional Gaussian distributions with the same covariance, which is the identity matrix scaled by the scalar $\sigma > 0$. $\lambda \in (0, 1)$ is a weighting parameter and $\mu_X = (m/2, 0, \cdots, 0)'$ and $\mu_Y = (-m/2, 0, \cdots, 0)'$ are both vectors of zeroes apart from the first entries, which are $m/2$ and $-m/2$ respectively. The distribution of M is then bimodal whenever $m \neq 0$, and the distance between the two modes is given by $|m|$. When keeping σ fix, increasing m complicates Monte Carlo sampling because the modes are less connected. Corresponding with the typical size of a target distribution when estimating medium-scale DSGE models, let M be in $n = 35$ dimensions.[21]

Figure 1 illustrates this exercise graphically by marginalizing over the first dimension. The shaded areas mark the 2.5%-percentile and the median of the first dimension. Each ensemble is initialized with a sample from $\mathcal{N}_n(\mathbf{0}_n, \sqrt{2}I_n)$. The initial ensemble is hence distributed across the domain of M, with relatively more chains closer to the origin (dashed blue line in Figure 1). Calculations are done for $\sigma = 0.05$ and distances of $m \in \{1, 2, 3\}$ (the columns of figure 1).[22] The first row shows the target distribution for $\lambda = 0.5$ where both modes peak at the same maximum density. For $m = 1$ both modes are connected, meaning that for any point between the modes the density is still reasonably large (that is, larger than 0.1 for the cases considered here). For $m = 2$ the trough between the models is relatively short in distance, but the minimum density is already close to zero. The gap for which the density is zero again increases considerably when setting $m = 3$, for which the modes are fully disconnected. Here, the typical set is clearly disjunct and thus difficult to traverse. The challenge for MCMC sampling lies in the fact that the chains must be able to bridge this gap, which for conventional samplers is unlikely once the density in the intermediate region is close to zero.

For each of the nine exercises, I conduct 100 batches of 210 chains each (correspondingly, $n_c = 6n$), let each batch run for 2000 iterations with $\chi = 0.1$, and then calculate the 2.5%-percentile and the median over the first dimension. Table 1 presents the root mean squared errors (RMSE) of these target measures over all batches. As the table suggests, DIME MCMC performs very well over all nine exercises. Even when M is fully disconnected ($m = 3$), the sampling error only increases marginally. The only exception is the estimate of the median for the first row where $\lambda = 0.5$. This finding can, however, be attributed to the peculiar effect that for $m = 2$ and $m = 3$ the posterior density in the gap between the two modes is almost zero because both modes have the exact same density masses. Correspondingly, this region contains no chains and a precise quantification of the median is impossible unless we let the number of iterations go to infinity.

For the simulations in the second and third row of Figure 1 and Table 1 λ is set to 0.33 and 0.25, respectively. This example is more challenging because some chains must jump between the modes in order to correctly reflect the different density masses of the modes. In practice any single-particle sampler is likely to "get stuck" in either of the mode, thereby ultimately misrepresents the posterior distribution. Yet, also for this example RMSEs are very small and acceptance ratios are in the desired range between 20-25%. This neat performance is granted by the global transition kernel, which allows to reshuffle chains between the two modes. Consequently, DIME performs less good if the probability of drawing the global transition kernel χ

[21] The posterior of the model of Smets and Wouters (2007) has 36 dimensions while the posterior of the HANK model estimated in Section 5 has 31 dimensions.

[22] These values are chosen to demonstrate the frontier of what is possible with DIME, without additional adjustments of the algorithm.

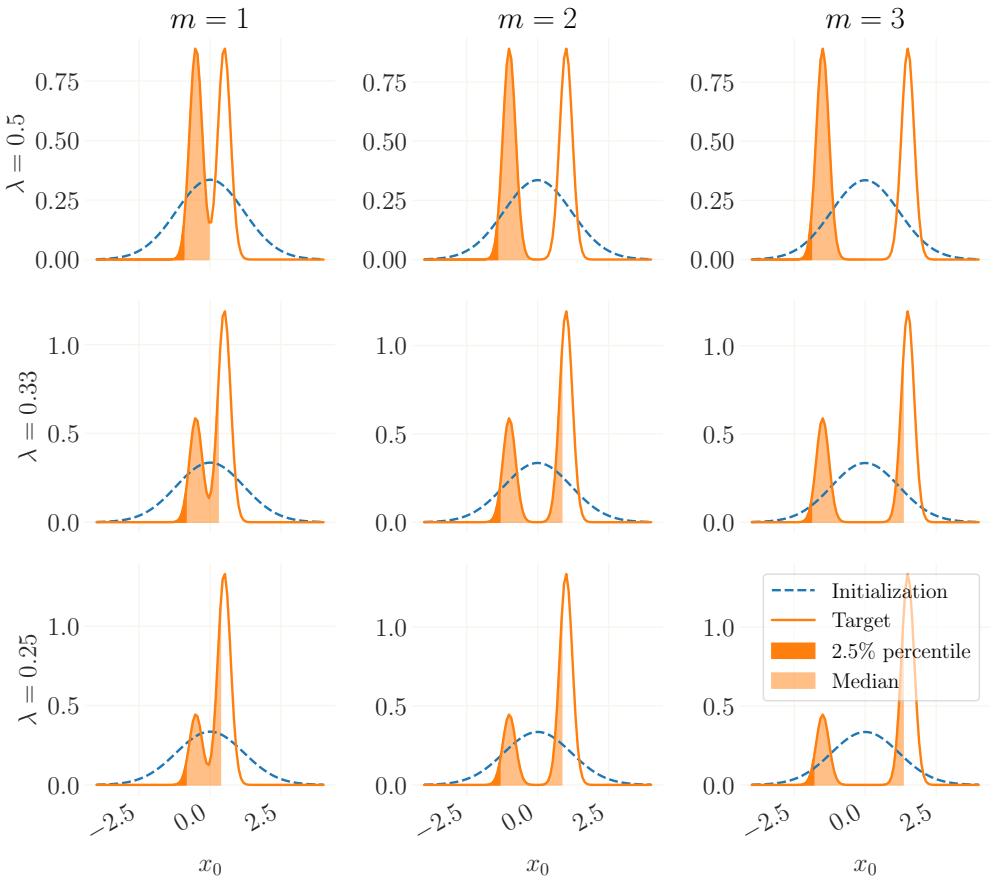

Figure 1: A 35 dimensional multivariate Gaussian mixture, marginalized over the first dimension (orange line). The dashed line depicts the initialization distribution of the ensemble. The frontier between the dark orange and light orange shaded area marks the 2.5%-percentile and the frontier between the light shaded area and no shade marks the median of the distribution.

is set larger than 20%. In that case, too many chains are reshuffled too early, thereby causing estimates of the global proposal distribution to ignore the second mode. This corroborates the previous recommendation of setting $\chi = 0.1$ for black-box distributions. For this setup, DIME MCMC seems to be able to reliably sample from the typical set of high-dimensional and bimodal distributions, even if the modes are fully disconnected and the typical set is thus discontinuous.

	$m = 1$		$m = 2$		$m = 3$	
	2.5% HDI	median	2.5% HDI	median	2.5% HDI	median
$\lambda = 0.5$	0.00827	0.08253	0.00960	0.58256	0.01239	1.08592
$\lambda = 0.33$	0.00946	0.01337	0.01004	0.01709	0.01453	0.02222
$\lambda = 0.25$	0.01253	0.00944	0.01308	0.01148	0.01897	0.01592

Table 1: RMSEs of the estimated 2.5%-percentile and the median of the first dimension of the target distributions. Results obtained from 100 batches.

4 The Smets-Wouters model

A common benchmark case for the Bayesian estimation of DSGE models is the work of Smets and Wouters (2007, henceforth SW), who pioneered the use of Bayesian methods for bringing medium-scale DSGE models to the data. I use this prominent reference in three exercises. First, I asses whether DIME MCMC is able to recover the posterior distribution from the original paper. Clearly, this can be seen as a minimum requirement for any sampler in oder to be of interest for macroeconometricians. Secondly, I use the model of SW to compare DIME with the perfomance of the local and global transition kernel *alone*, as well as with another popular ensemble MCMC sampler from the literature. Lastly, I use their model to numerically evaluate the trade-off of more chains versus longer chains.

For each of the exercises exactly the same model specification, priors, data and data treatment as in the original paper are used. All estimations are done on a workstation with 40 Intel Xeon CPUs with 3.1GHz each and a total of 32GB RAM. I use the package *pydsge* for parsing and solving the linear model, and to calculate the likelihood using the standard Kalman filter.[23]

4.1 Comparison with the original estimates

To reproduce the estimation from SW I let an ensemble of 200 chains run for 3000 iterations, of which 500 are kept as the posterior. The original estimation relies on 250.000 samples (of which 50.000 are discarded) obtained using RWMH after running an optimization procedure from pre-optimized starting values. Table A.4 in Appendix A shows summary statistics over the posterior distribution of the estimation together with posterior statistics from SW. Overall, the DIME MCMC estimates and the posterior values from the original estimation of SW are very closely aligned. Notable differences are the estimate of the standard deviation of the risk premium shock, σ_u, which is substantially larger than the SW estimate, as well as in the estimate of the steady state labor supply $\bar{l}$. Judging from the standard deviation of the latter estimate, the parameter seems not well identified. In summary, the estimates indicates that the DIME MCMC can fully recover the original values of SW. The table also shows the marginals over the proposal distribution of the global transition kernel which, in terms of mean and standard deviation and as expected, is very closely aligned to the posterior distribution.

4.2 Comparison of different Ensemble MCMC samplers

Turn now to the comparison of the performance of DIME MCMC with its individual components – the local and global transition kernel – taken alone, as well as the "Stretch" move of Goodman and Weare (2010). I let each sampler run ten times over different random seeds. For each sampler and seed I chose

[23] Pydsge is a toolbox to solve, filter, and estimate DSGE models in Python language, which is presented in Boehl and Strobel (2022a). The package is available in the official Python repositories and developed and maintained at GitHub: https://github.com/gboehl/pydsge.

the same initialization and, again, let 200 chains run for 3000 periods. To allow for fair comparison the bijective mapping between proposal and parameter space is used for *all* samplers. Figure 2 plots the log-density of each *single* chain over time. In each panel, the dashed line marks the mode i.e., the maximum posterior density value. The different colors correspond each to a different ensemble run. The top-left panel plots the batches using the DIME sampler. Chains converge quickly towards the high density region of the posterior, reaching the 68% set of the posterior roughly in period 300 and the 97.5% set in about iteration 500. Although convergence is difficult to asses, it seems as if all DIME chains across all ensembles have converged to the posterior roughly at iteration 700. Throughout the convergence period the single chains remain relatively close to each other, both within and across ensembles.

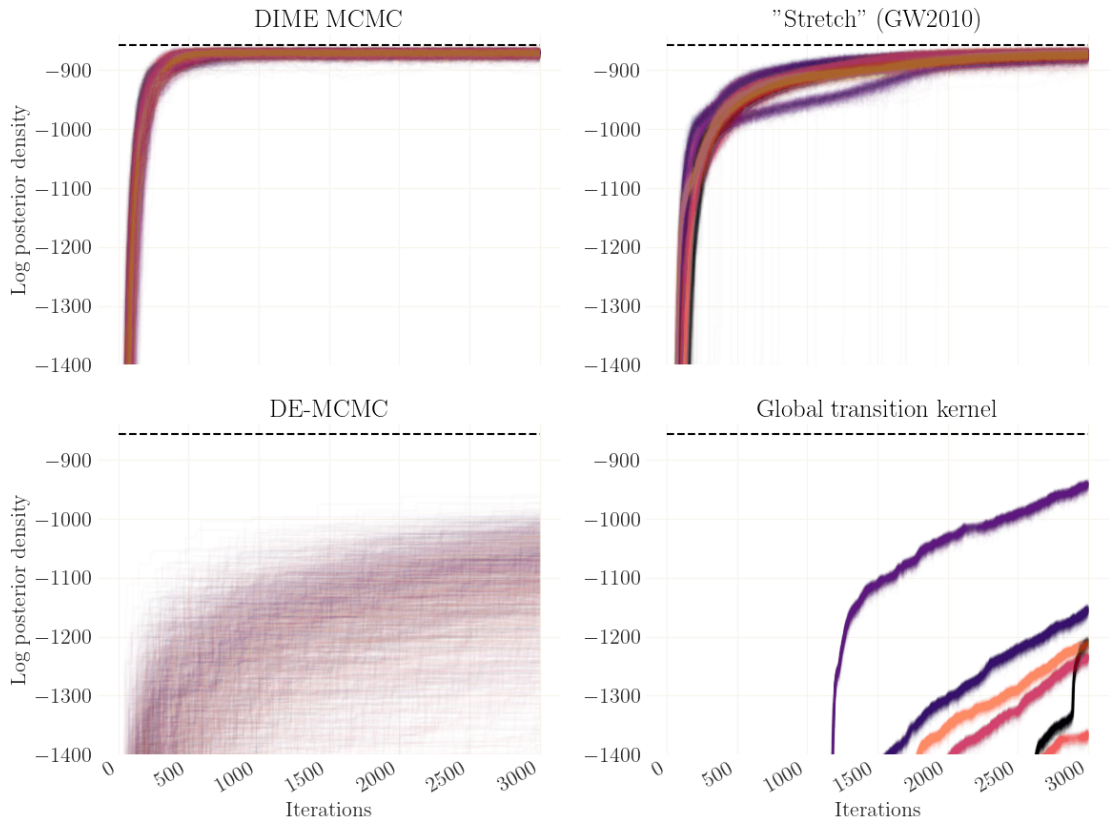

Figure 2: Using different ensemble MCMC methods to estimate the model of Smets and Wouters (2007). The "DE-MCMC" method concurrs with the local transition kernel. The panels show each using a different Ensemble MCMC method, the traces of the log-likelihood of several ensembles over time, Each colors represent a different ensemble with different random seed. For each ensemble all individual chains are plotted. For all panels the same scaling is used.

The panel at the top-right plots the performance of the "Stretch" move of Goodman and Weare (2010). This proposal kernel is quite popular in the field of astrophysics. Across batches, initial convergence of the ensembles is relatively rapid, but convergence then slows down. The 68% and 97.5% sets are reached between iterations 1500 and 2000 and after 2250, respectively. Correspondingly, the chains do not converge

to the posterior before iteration 2500. Additionally, convergence behavior differs slightly across batches.

The bottom-left panel in Figure 2 shows ensembles following the local transition kernel, which concurs with the differential evolution MCMC (DE-MCMC) method of Ter Braak (2006) in the implementation of Foreman-Mackey et al. (2013). The graphic suggests that burn-in for DE-MCMC is slow and the ensemble does not converge to the prior distribution within the given 3000 iterations. An apparent problem seems to be that dispersion in log-density across chains in each ensemble is very large. The likely reason is that each chain moves ex-ante independently, i.e. state of the complete ensemble is only used for relative repositioning of each chain. When ensemble dispersion is high, the quality of replacement proposals deteriorates and convergence slows down even further, thereby causing single chains to converge very slowly.

The bottom-right panel plots a batch of ensembles following only the global transition kernel, i.e. without mixing with the local transition kernel. As apparent from the figure, the adaptive independence Metropolis-Hastings approach *alone* performs quite badly and exhibits slow convergence. To understand this, note that most proposals lie close to the current maximum likelihood value. In the absence of mixing with other kernels, the proposal distribution quickly shrinks towards a narrow neighborhood around this maximum. Consequently, the global kernel only generates proposals in the direct vicinity of the maximum and does not sufficiently explore the global domain. Correspondingly, performance during burn-in varies much over random seeds.

DIME MCMC is a mixture kernel of DE-MCMC and the global transition kernel. Figure 2 clearly illustrates their individual weaknesses. The DE-MCMC ensemble is overdispersed, which causes unfavorable individual proposals and, in turn, slow converge. In contrast, the ensemble of the global transition kernel converges to a narrow ball because chains with a higher probability density have a larger weight in the proposal distribution. Thus, candidate proposals will lie in the immediate proximity of the current ensemble, again causing slow convergence. The key to the performance of DIME is that both kernels are strongly complementary and these individual weaknesses chancel out: when mixing the two kernels, the "fairly good" proposals from the global transition kernel are sufficient to reshuffle chains that would otherwise (i.e. with DE-MCMC alone) be stuck in regions with lower probability density.[24] This reshuffling is efficient to decrease the dispersion of the ensemble and, consequently, the proposals of the local transition kernel improve, which helps to explore the broader neighborhood of the current state of the ensemble.

4.3 The number of chains n_c

Next, let me benchmark the sensitivity of the estimation results with respect to the number of chains n_c. Figure 3 illustrates burn-in speed and convergence dynamics in terms of the number of total function evaluations. For the chosen range of $n_c \in (4n, 6n)$ it seems that no setup emerges which is to be strongly preferred. I start with $n_c = 2n$, which is the minimum number of chains suggested by Foreman-Mackey et al. (2013). As depicted in top-left panel, convergence is slower than for a larger number of chains and the course of the different ensembles shows larger variation. For more chains ($n_c = 4n$, top-right panel) convergence is faster, with no significant difference to $n_c = 6n$ in the bottom-left panel. For larger ensembles ($n_c = 8n$, bottom-right) convergence per function iteration is again marginally slower whereas individual ensembles are almost indistinguishable.

This exercise reveals a mild trade-off between the number of iterations and the quality of the proposal candidates. For just a few chains per ensemble, each iteration requires only few function evaluations. However, the relatively small number of chains produces less favorable replacement proposals, which implies

[24] Following a similar intuition, Vrugt et al. (2009) use the inter-quartile range to discover potential outlier chains, which are then replaced with the current best chain.

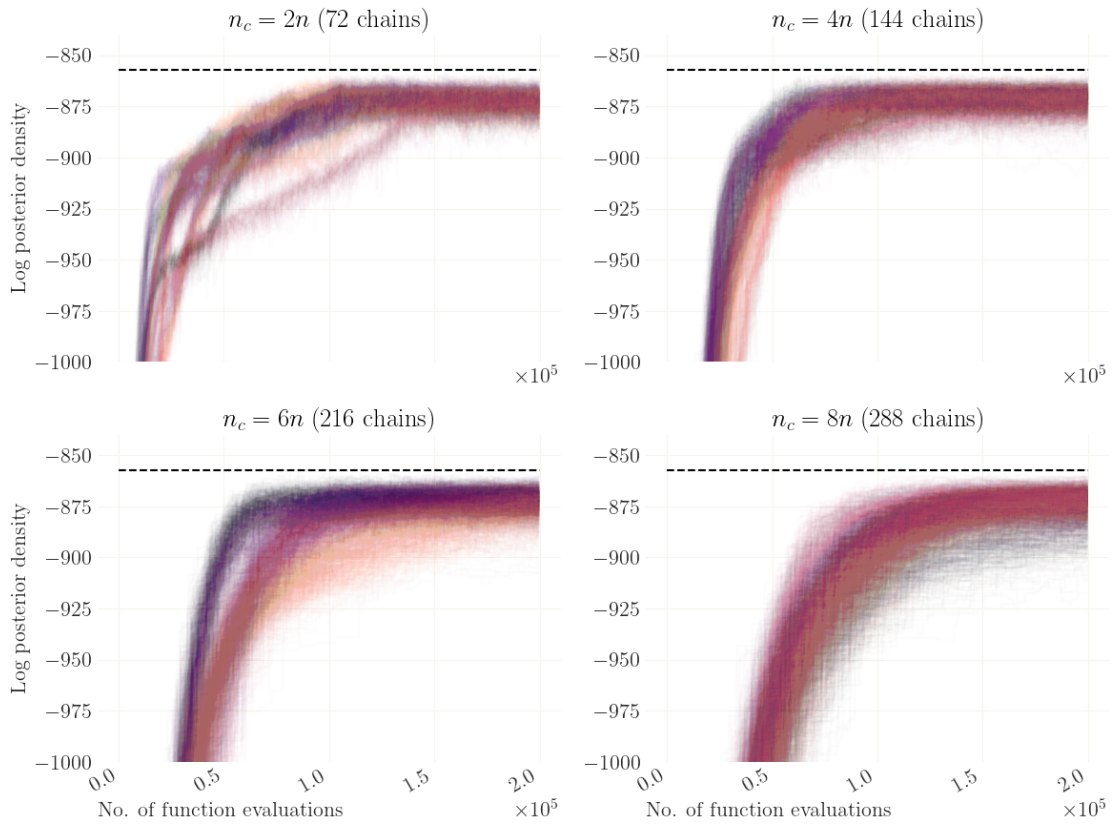

Figure 3: Using DIME MCMC to estimate the model of Smets and Wouters (2007). Each panel shows the course of the log-likelihood of several ensembles over time, using different numbers n_c of chains. Each colors represent a different ensemble with different random seed. For each ensemble all individual chains are plotted. The scaling of all panels is the same.

that more iterations are needed until convergence. When, in contrast, ensembles are large, each iteration is relatively costly and fewer iterations can be done for a given number of function evaluations. However, for a large range of n_c between $4n$ and $6n$ a larger number of chains approximately compensates one-to-one for fewer iterations. Importantly, this suggests that estimations can be scaled very well when parallelizing chains on computers with a larger number of processing units. An increase in the number of chains does always require less chain iterations. Even if this relationship would not be one-to-one, this implies that it is advisable to use at least as many chains as numbers of processors.

Appendix B provides Gelman and Rubin (1992) statistics over the number of chains. The Gelman-Rubin statistic is a measure of convergence. The results substantiate the hypothesis that the optimal number of chains lie between $4n$ and $6n$, with a lower number of iterations (i.e., a larger number of chains) causing higher Gelman-Rubin coefficients. This indicates that it is more important to run many iterations than to run a large number of chains. However, average integrated autocorrelation times (Sokal, 1997) across chains, parameters and different ensemble sizes are relatively constant around 40. This in turn suggests again that the ensemble size does not have a major influence on sampling quality.

Overall I suggest to chose n_c to be a multiple of the number of available processors which lies in in the range $5n$ and $6n$, and to set $\chi = 10\%$. It is advisable to monitor the trace plot of the likelihood function (e.g. as in Figure 3) and the histogram of the posterior. For very rugged or mulimodal distributions the number of chains should be increased. In such cases it is additionally expedient to decrease χ to prevent chains from getting reshuffled too early. In contrast, if for some reason the posterior is expected to be rather near-Gaussian, a larger value of χ can be chosen, which will decrease autocorrelation times and hence requires fewer ensemble iterations during the sampling stage, i.e. after burn-in.

5 Full estimation of HANK

To explore the full potential of DIME MCMC I use the sampler to estimate a Heterogeneous-Agent New Keynesian (HANK) model with portfolio choice and including all the features of a conventional medium-scale DSGE model. The central novelty relative to the literature is that I include the households' preference parameters in the set of estimated parameters, which increases the complexity of the calculations significantly.

HANK models are a relatively new class of models (see, e.g., Gornemann et al. (2012) and Kaplan et al. (2018a)) that combine the New Keynesian paradigm with household heterogeneity and incomplete financial markets. This allows, for example, to study the impact of economic inequality on macroeconomic aggregates and vice versa. While the estimation of HANK models is pioneered by Winberry (2018), Bayer et al. (2020, henceforth BBL) and Auclert et al. (2021), these papers do not estimate the parameters of households' preferences that govern the households' optimization problem. The reason for excluding these parameters is that they alter the model's steady state, which would then have to be re-evaluated for every posterior draw. As the re-evaluation of the steady state involves finding a stationary distribution such that all equilibrium conditions are satisfied, this comes at large computational costs. Consequently, BBL and Auclert et al. (2021) both opt to calibrate all parameters which affect the model's steady state and focus on estimating the remaining parameters.

However, the households' preference parameters have the potential to form central attributes of the ergodic distribution of assets and income, and may thus qualitatively and quantitatively determine the magnitude of the novel channels exposed by this class of models. Hence, these parameters could potentially affect the macroeconomic dynamics of this class of models fundamentally. Since at the same time, their inclusion in the estimation is computationally expensive – finding the steady state and the stationary distribution takes about 10 seconds for the implementation considered here – it is a perfect use case for the DIME sampler.

5.1 Model and Data

The model is the fusion of a two-asset HANK model with a medium-scale DSGE model. The HANK core shares many features with the models of Auclert et al. (2021) and Kaplan et al. (2018b). This core is extended by several frictions in the spirit of Smets and Wouters (2007), which, among other features, allow for additional endogenous transmission of aggregate shocks over time.[25] To ease comparison with the DSGE literature I use the priors of Smets and Wouters (2007). Accordingly, some of the functional forms (e.g. capital adjustment costs and Calvo pricing) are adapted from there. In the following I discuss only those equations that deviate from Auclert et al. (2021) and refer the interested reader to Appendix C for further details on the model.

[25] It is well known that such endogenous persistence is a crucial feature to replicate the hump-shaped empirical responses that are reported in the VAR literature.

Households supply labor and have access to a liquid and an illiquid asset. Importantly, they face borrowing constraints on both assets, and adjustment costs on the illiquid asset. Firms accumulate capital, and staggered price setting results in a conventional Phillips curve. Adding ad-hoc price indexation with parameter ι_p, inflation π_t is determined by

$$\pi_t - \bar{\pi} = \frac{\beta}{1+\beta\iota_p}\left(E_t\pi_{t+1} - \bar{\pi}\right) + \frac{\iota_p}{1+\beta\iota_p}(\pi_{t-1} - \bar{\pi}) + \kappa_p\left(\widehat{MC}_t - \frac{1}{\mu}\right) + \epsilon_{p,t}, \tag{21}$$

where $\bar{\pi}$ is the steady state inflation. $\epsilon_{p,t}$ is assumed to follow an AR(1) process around its zero mean and the slope of the Phillips curve is given by $\kappa_p = \frac{1-\zeta_p\beta}{1+\iota_p\beta}\frac{1-\zeta_p}{\zeta_p}$. Labor unions set nominal wages which are also subject to staggered pricing, giving rise to a Phillips curve for wages. Adding wage indexation with parameter ι_w, this yields

$$\begin{aligned}\pi_t^w - \bar{\pi} = \frac{\beta}{1+\beta\iota_w}\left(E_t\pi_{t+1}^w - \bar{\pi}\right) + \frac{\iota_w}{1+\beta\iota_w}(\pi_{t-1}^w - \bar{\pi}) \\ + \kappa_w\left(\varphi N_t^{1+\nu} - \frac{(1-\tau_t)w_tN_t}{\mu_t^w}\int e_{it}c_{it}^{-\sigma}di\right) + \epsilon_{w,t},\end{aligned} \tag{22}$$

where $\epsilon_{w,t}$ as well follows an AR(1) process and $\kappa_w = \frac{1-\zeta_w\beta}{1+\iota_w\beta}\frac{1-\zeta_w}{\zeta_w}$. Monetary policy sets the nominal interest rate r_t following a conventional monetary policy rule,

$$r_t^n - r^n = \rho\left(r_{t-1}^n - r^n\right) + (1-\rho)\left[\phi_\pi\left(\pi_t - \bar{\pi}\right) + \phi_y\Delta\ln Y_t\right] + \epsilon_{r,t}, \tag{23}$$

with $\epsilon_{r,t}$ as an exogenous AR(1) process representing monetary policy surprises. Note that in order to remain agnostic about the central bank's welfare objective, a traditional measure of output gap is absent in this equation. The setup of capital adjustment costs is as in Smets and Wouters (2007) and yields the following expressions for Tobin's Q and the firm's investment decisions:

$$R_{t+1}q_t = (1-\delta)E_tq_{t+1} + \alpha E_t\left\{Z_{t+1}\frac{N_{t+1}}{K_t}^{1-\alpha}\widehat{MC}_{t+1}\right\}, \tag{24}$$

$$1 = \exp(\epsilon_{i,t})q_t\left[1 - S\left(\frac{I_t}{I_{t-1}}\right) - S'\left(\frac{I_t}{I_{t-1}}\right)\frac{I_t}{I_{t-1}}\right] + E_t\left\{\exp(\epsilon_{i,t+1})\frac{q_{t+1}}{R_{t+1}}S'\left(\frac{I_{t+1}}{I_t}\right)\left(\frac{I_{t+1}}{I_t}\right)^2\right\}, \tag{25}$$

where R_t is the gross real interest rate on liquid assets, $S(x) = \frac{1}{2S''}(x-1)^2$ is a quadratic adjustment cost function, and $\epsilon_{i,t}$ is an exogenous AR(1) process on the marginal productivity of investment. Finally, labor income taxation is progressive with parameter Ξ such that after-tax labor income y_{jt} is given by

$$y_{jt} = {y_{jt}^p}^{1-\Xi} + \int p(e_{jt})\left({y_{jt}^p} - {y_{jt}^p}^{1-\Xi}\right), \tag{26}$$

with pretax income $y_{jt}^p = (1-\tau_t)w_tN_te_{it}$.

For the estimation I use a subset of the data used in Boehl et al. (forthcoming) which amounts to a relatively conventional setup for medium scale models: growth rates of consumption, investment, output and wages, together with inflation, labor hours and the federal funds rate. The data is at quarterly frequency and ranges from 1983:I to 2008:IV. As in Justiniano et al. (2010), investment and consumption time series are adjusted such that investment also includes durables consumption. In the model, those seven observables are matched by seven economic shocks, which are all defined in percentage deviations from the steady state:

the two markup shocks, the monetary policy shock, a government spending shock on G_t, a discount factor shock on β_t and the shock on the marginal efficiency of investment, $\epsilon_{i,t}$. Further details can be found in Appendix D.

5.2 *Estimation methodology*

Model solution and likelihood inference is done following the methodology introduced in Auclert et al. (2021).[26] In brief, let y_t be the time t vector of model variables (including disaggregated variables) and let the sequence of first-order conditions and market clearing conditions, up to some distant point T periods in the future, be

$$F = \{f(y_{t-1}, y_t, E_t y_{t+1}; x)\}_{t=0}^{T} = \vec{0}, \tag{27}$$

which depends on the parameter vector x. Denote by $Y_t \subset y_t$ only the aggregated variables and by $Z_t \subset y_t$ those variables that are purely exogenous. The authors propose a novel and computationally efficient procedure of finding the steady state Jacobian matrix of F with respect to $\{Y_t\}_{t=0}^{T}$ and $\{Z_t\}_{t=0}^{T}$. These sequence-space Jacobians can then be used to calculate impulse responses to aggregate shocks up to a first order approximation. Notably, this works for the broad class of models for which it is not required to explicitly keep track of any of the disaggregated distribution variables on a global domain. Simulations are based on the sequence space rather than, as in BBL, the *state space*. The authors show that the first-order sequence space representation can be used directly for likelihood inference, without the need for using the Kalman filter (which would require a state space representation). In their application, the authors are able to re-use (parts of) the Jacobians depending on the types of parameters to be estimated. In contrast, in my application each Jacobian has to be calculated from scratch due to the re-evaluation of the steady state for each parameter draw.

In a deterministic setup, the steady state $\bar{y}$ must satisfy

$$f(\bar{y}, \bar{y}, \bar{y}; x) = \vec{0}. \tag{28}$$

Given a guess for the steady state values of aggregated variables $\bar{Y}$, the stationary distribution of idiosyncratic variables can be found by solving for the stationary decision rules via backward iteration, and solving for the stationary distribution via forward iteration. Hence, there exists a known mapping $\bar{Y} \to \bar{y}$, and finding $\bar{Y}$ can be done using conventional root finding methods. Often, the size of this root finding problem can further be reduced to only searching a subset $\bar{K} \subset \bar{Y}$ since $\bar{Y}$ can be expressed in terms of this subset. Still, finding $\bar{y}$ is relatively time consuming and must be repeated for any parameter draw x if the households' micro parameters change.[27]

5.3 *Estimation results*

As usual, some parameters are fixed prior to the estimation. These parameters, most of which configure the technical setup of the estimation (e.g. the number of grid points), can be found in table C.6 in Appendix C. All other parameters are estimated using the priors presented in the first three columns in tables 2 and 3,

[26] The authors provide their set of methods as a Python toolbox maintained at GitHub: https://github.com/shade-econ/sequence-jacobian.

[27] For any numerical root finding method a good initial guess is crucial. This also holds for finding the steady state. For bad initial guesses, the root search may either diverge, crash due to numerical errors when solving for the stationary distribution, or simply take up a very long time. This is problematic because it also prohibits the calculation of the likelihood for cases in which a likelihood actually exists. In practice, for every draw I use the steady state values for the prior mean as the initial guess, which causes about 2/3 of all parameter vectors sampled from the prior distribution to be accepted.

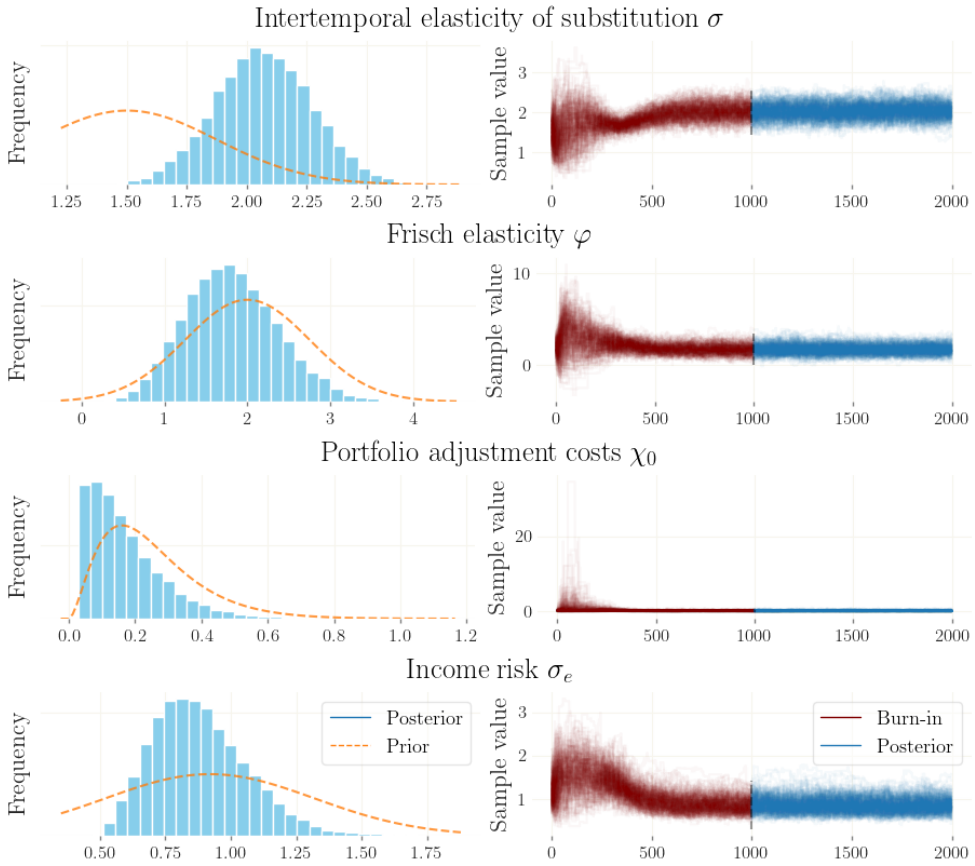

Figure 4: Posterior histogram and prior distribution (left side, blue and orange) of selected households' parameters with trace plots of the ensemble over iterations (right side). The part colored in red is discarded as burn-in.

which follow the specification of Smets and Wouters (2007). Exceptions are the portfolio adjustment cost parameter χ_0, tax progressively parameter Ξ, and the standard deviation of the AR(1) process for idiosyncratic labor productivity σ^e, which are specific to the HANK model. For these parameters I opt for generally flat priors. I let the prior mean of σ^e be 0.92 as in Auclert et al. (2021) and, for the same reason, set the prior mean of χ_0 to 0.25.

For the estimation I run a DIME MCMC ensemble with $n_c = 192$ chains for 2000 iterations. The last 1000 iterations are kept as a sample from the posterior. The number of chains is the number of available CPUs (48) times 4 and, with $n_c \approx 5.33n$, lies in the range recommended in the previous section. The ensemble converges to the high-density region of the posterior after about 800 iterations and the full estimation takes 84 hours on the machine with 48 cores. Respectively, on a machine with 192 cores each chain would have a dedicated processor and the estimation would take less then a day (about 21h). On a machine with even more cores the number of chains should increase accordingly, which would still reduce the overall processing time almost one-to-one (see section 4). The figures F.6 and F.7 to F.11 in Appendix F graphically illustrate and document the convergence of the ensemble. Tables 2 and 3 show summary statistics of the posterior distribution of model parameters.

		Prior			Posterior		
		distribution	mean	std.	mean	std.	mode
σ	intertemporal elasticity of substitution	normal	1.500	0.375	2.043	0.202	1.850
φ	Frisch elasticity	normal	2.000	0.750	1.738	0.562	1.805
ζ_p	Calvo parameter for price setting	beta	0.500	0.100	0.590	0.050	0.592
ζ_w	Calvo parameter of wage setting	beta	0.500	0.100	0.416	0.069	0.413
ι_p	price inertia	beta	0.500	0.150	0.331	0.129	0.335
ι_w	wage inertia	beta	0.500	0.150	0.322	0.147	0.303
S''	derivative capital adjustment costs	gamma	4.000	2.000	2.279	0.702	1.725
ϕ_π	monetary policy coefficient inflation	gamma	1.500	0.250	2.322	0.219	2.198
ϕ_y	monetary policy coefficient output	gamma	0.125	0.050	0.222	0.063	0.205
ρ	monetary policy persistence	beta	0.750	0.100	0.652	0.052	0.680
$\bar{y}$	trend output	normal	0.400	0.100	0.438	0.026	0.434
$\bar{n}$	steady state labor hours	normal	0.000	2.000	-0.047	1.961	1.469
π^*	inflation target	gamma	0.625	0.100	0.596	0.051	0.624
i^*	steady state nominal interest rate	gamma	1.250	0.100	1.239	0.089	1.259
χ_0	portfolio adjustment costs (scale)	gamma	0.250	0.150	0.153	0.118	0.094
Ξ	tax progressivity	beta	0.200	0.100	0.089	0.059	0.071
σ^e	standard deviation of labor productivity	normal	0.920	0.400	0.860	0.185	1.064

Table 2: Estimation results for HANK: model parameters

Figure 5 shows impulse response functions to a monetary policy and a TFP shock sampled from the posterior of the estimated model. These impulse responses look rather conventional: output increases persistently as the consequence of a TFP shock, while the fall in inflation is rather transitory. The effect of monetary policy shocks is rather short lived and stimulates output and inflation alike.

This paper focusses on the performance of the DIME sampler instead of the economic dynamics of the estimated HANK model. For this reason I deem an in-debt analysis of the economic implications of the estimated model out of the scope of this paper and leave it as a promising endeavour for future research. Nevertheless, a cursory comparison of the parameter estimates from the HANK model with those of Smets and Wouters (2007) – for a somewhat smaller sample – reveals some surprising differences.[28] In HANK, the inverse elasticity of substitution, σ, is relatively large. This differs to the estimate of SW and the findings documented in Boehl and Strobel (2022b,a) for US data until 2019, who report values close-to unity. This estimate is likely to be related to the fact that HANK models feature an additional precautionary savings channel which originates from the assumption of incomplete financial markets.

An interesting finding is that in the HANK model both the price and the wage Phillips curve are identified to be relatively steep, which is reflected by Calvo adjustment probabilities ζ_p and ζ_w to be estimated relatively low. This stands in contrast to many more recent estimates which find rather large values for these parameters, which suggests a very flat Phillips curve. While this effect may come from different data samples and slightly different specifications of the Phillips curves, it calls for further investigation. The relatively lower estimate of S'' is also documented in BBL and may indicate that capital adjustment costs play a smaller role in the HANK model, which may be due to the fact that in the HANK model, portfolio adjustment represent a additional friction that actively influences the capital investment decision. The other parameters in table 2, which govern the monetary policy rule and the steady state values of the observables, are well-aligned with the original estimates in SW. These parameters are likely identified independently of

[28] The estimation of Smets and Wouters (2007) is replicated in Appendix A.

		Prior			Posterior		
		distribution	mean	std.	mean	std.	mode
ρ_z	AR coefficient technology shock	beta	0.500	0.200	0.957	0.018	0.960
ρ_r	AR coefficient MP shock	beta	0.500	0.200	0.619	0.070	0.640
ρ_g	AR coefficient gov. spending shock	beta	0.500	0.200	0.993	0.006	0.993
ρ_w	AR coefficient wage MU shock	beta	0.500	0.200	0.985	0.006	0.980
ρ_p	AR coefficient price MU shock	beta	0.500	0.200	0.911	0.028	0.917
ρ_i	AR coefficient investment shock	beta	0.500	0.200	0.837	0.042	0.836
ρ_β	AR coefficient interest wedge shock	beta	0.500	0.200	0.962	0.030	0.991
σ_z	standard dev. technology shock	inv.gamma	0.100	0.250	0.415	0.036	0.430
σ_r	standard dev. MP shock	inv.gamma	0.100	0.250	0.130	0.020	0.113
σ_g	standard dev. gov. spending shock	inv.gamma	0.100	0.250	1.148	0.088	1.105
σ_w	standard dev. wage MU shock	inv.gamma	0.100	0.250	2.662	0.732	2.448
σ_p	standard dev. price MU shock	inv.gamma	0.100	0.250	0.201	0.047	0.188
σ_i	standard dev. investment shock	inv.gamma	0.100	0.250	1.289	0.215	1.350
σ_β	standard dev. interest wedge shock	inv.gamma	0.100	0.250	0.047	0.017	0.029

Table 3: Estimation results for HANK: parameters of exogenous processes

the model's setup of the household sector.

The estimate of the portfolio adjustment cost parameter χ_0 is well below its respective prior mean, pointing towards a less accentuated role of the households' portfolio choice problem. Complementary, the standard deviation of the idiosyncratic labor productivity, σ_z, is also slightly below to its prior value. Both of these values correspond to the parameters chosen by Kaplan et al. (2018a) rather than those of Auclert et al. (2021). While by no means this evidence can be used to evaluate the role of idiosyncratic income risk or portfolio choice, it also calls for further investigation. Lastly, the estimates of the parameters that govern the exogenous autoregressive processes are much in line with conventional estimates, where technology, government spending and investment specific shocks are usually highly autocorrelated.

In Appendix E I repeat the estimation but while letting the households' state space being represented on a smaller grid (480 nodes instead to 2625 nodes). This reduction cuts the estimation time about one-third to 50 hours in total. As the reported estimates suggest, the reduction in the number of approximation nodes does not have a significant impact on the estimation results. Consequently, it may be possible to obtain reliable results from using a smaller representation of the households' state space.

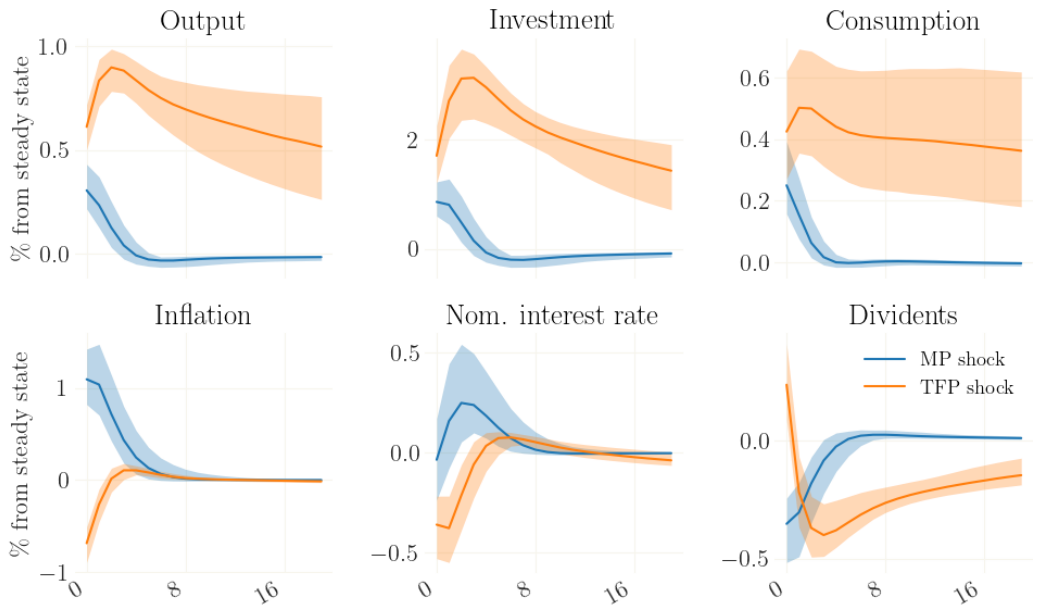

Figure 5: Impulse response functions to a monetary policy shock (blue) and a shock to TFP (orange). Responses and credible sets correspond to 1000 simulations drawn from the posterior distribution. The measures are annualized where applicable.

6 Conclusion

This paper develops the differential-independence mixture ensemble (DIME) MCMC sampler. The sampler can be seen as a "Swiss Army knife" that is applicable for posterior sampling and global optimization problems alike. I show that the method performs well for high-dimensional and multimodal distributions. The proposal density of DIME is generated endogenously from the state of an ensemble of many chains, thereby automatically adapting to the shape of the current estimate of the posterior distribution. A separation of parameter space and proposal space guarantees that proposals respect the bounds of prior distribution, which results in significantly higher acceptance rates and, consequently, in higher sampling efficiency.

Mixing between local and global proposal leads to very fast burn-in and convergence to the high density region of the posterior. I show that DIME MCMC is easy to parallelize, where the number of iterations required for convergence decreases almost one-to-one with the number of chains. This makes the method feasible for large-scale problems with models that are computationally expensive to simulate.

The DIME sampler allows, for the first time, to include the households' micro parameters when estimating a HANK model with portfolio choice. These parameters, through the households' decisions, determine the endogenous distribution of assets. The detailed analysis of the estimated model, e.g. by putting the resulting parameter estimates in relation to estimates from micro data, is a promising endeavour for future research.

A natural extension to DIME, also for future research, is to replace the differential-evolution proposal in the local transition kernel by a HMC proposal for applications in which automatic differentiation is feasible. In such setting, HMC supersedes differential evolution MCMC: if the Jacobian can be evaluated at low computational costs, proposals can readily be well-adopted to the actual shape of the posterior. Yet, the

mixture with the global transition kernel would remain powerful as it can speed up burn-in and enables sampling from very challenging multimodal distributions.

References

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

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

Betancourt, M., 2017. A conceptual introduction to hamiltonian monte carlo. arXiv preprint arXiv:1701.02434 .

Boehl, G., 2023. Robust nonlinear transition dynamics in hank. Available at SSRN 4433585 .

Boehl, G., Goy, G., Strobel, F., forthcoming. A structural investigation of quantitative easing. Review of Economics and Statistics .

Boehl, G., Strobel, F., 2020. US business cycle dynamics at the zero lower bound. Bundesbank Discussion Papers 65/2020. Deutsche Bundesbank.

Boehl, G., Strobel, F., 2022a. Estimation of DSGE Models with the Effective Lower Bound. CRC 224 Discussion Papers. University of Bonn and University of Mannheim, Germany.

Boehl, G., Strobel, F., 2022b. The Empirical Performance of Financial Frictions Since 2008. CRC 224 Discussion Papers. University of Bonn and University of Mannheim, Germany.

Bognanni, M., Herbst, E., 2018. A sequential monte carlo approach to inference in multiple-equation markov-switching models. Journal of Applied Econometrics 33, 126–140.

ter Braak, C.J., Vrugt, J.A., 2008. Differential evolution markov chain with snooker updater and fewer chains. Statistics and Computing 18, 435–446.

Carpenter, B., Gelman, A., Hoffman, M.D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., Riddell, A., 2017. Stan: A probabilistic programming language. Journal of statistical software 76.

Carroll, C.D., 2006. The method of endogenous gridpoints for solving dynamic stochastic optimization problems. Economics letters 91, 312–320.

Chib, S., Ramamurthy, S., 2010. Tailored randomized block mcmc methods with application to dsge models. Journal of Econometrics 155, 19–38.

Childers, D., Fernández-Villaverde, J., Perla, J., Rackauckas, C., Wu, P., 2022. Differentiable State-Space Models and Hamiltonian Monte Carlo Estimation. Technical Report. National Bureau of Economic Research.

Damlen, P., Wakefield, J., Walker, S., 1999. Gibbs sampling for bayesian non-conjugate and hierarchical models by using auxiliary variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 61, 331–344.

Dillon, J.V., Langmore, I., Tran, D., Brevdo, E., Vasudevan, S., Moore, D., Patton, B., Alemi, A., Hoffman, M., Saurous, R.A., 2017. Tensorflow distributions. arXiv preprint arXiv:1711.10604 .

Duane, S., Kennedy, A.D., Pendleton, B.J., Roweth, D., 1987. Hybrid monte carlo. Physics letters B 195, 216–222.

Edge, R.M., Gürkaynak, R.S., Kisacikoglu, B., 2013. Judging the DSGE model by its forecast. Technical Report. mimeo.

Flegal, J.M., Haran, M., Jones, G.L., 2008. Markov chain monte carlo: Can we trust the third significant figure? Statistical Science , 250–260.

Foreman-Mackey, D., Hogg, D.W., Lang, D., Goodman, J., 2013. emcee: the mcmc hammer. Publications of the Astronomical Society of the Pacific 125, 306.

Gelman, A., Rubin, D.B., 1992. Inference from iterative simulation using multiple sequences. Statistical science , 457–472.

Geman, S., Geman, D., 1984. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE Transactions on pattern analysis and machine intelligence , 721–741.

Geweke, J., 1999. Using simulation methods for bayesian econometric models: inference, development, and communication. Econometric reviews 18, 1–73.

Goodman, J., Weare, J., 2010. Ensemble samplers with affine invariance. Communications in applied mathematics and computational science 5, 65–80.

Gornemann, N., Kuester, K., Nakajima, M., 2012. Monetary policy with heterogeneous agents. Working Papers 12-21. Federal Reserve Bank of Philadelphia. URL: http://ideas.repec.org/p/fip/fedpwp/12-21.html.

Haario, H., Saksman, E., Tamminen, J., 2001. An adaptive metropolis algorithm. Bernoulli , 223–242.

Hastings, W.K., 1970. Monte carlo sampling methods using markov chains and their applications .

Herbst, E., Schorfheide, F., 2014. Sequential monte carlo sampling for dsge models. Journal of Applied Econometrics 29, 1073–1098.

Herbst, E.P., Schorfheide, F., 2015. Bayesian estimation of dsge models, in: Bayesian Estimation of DSGE Models. Princeton University Press.

Holden, L., Hauge, R., Holden, M., 2009. Adaptive independent metropolis–hastings. The Annals of Applied Probability 19, 395–413.

Igel, C., Hansen, N., Roth, S., 2007. Covariance matrix adaptation for multi-objective optimization. Evolutionary computation 15, 1–28.

Justiniano, A., Primiceri, G.E., Tambalotti, A., 2010. Investment shocks and business cycles. Journal of Monetary Economics 57, 132–145. URL: https://ideas.repec.org/a/eee/moneco/v57y2010i2p132-145.html.

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

Kaplan, G., Moll, B., Violante, G.L., 2018b. Monetary policy according to hank. American Economic Review 108, 697–743.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E., 1953. Equation of state calculations by fast computing machines. The journal of chemical physics 21, 1087–1092.

Neal, R.M., et al., 2011. Mcmc using hamiltonian dynamics. Handbook of markov chain monte carlo 2, 2.

Nelder, J.A., Mead, R., 1965. A simplex method for function minimization. The computer journal 7, 308–313.

Nelson, B., Ford, E.B., Payne, M.J., 2013. Run dmc: an efficient, parallel code for analyzing radial velocity observations using n-body integrations and differential evolution markov chain monte carlo. The Astrophysical Journal Supplement Series 210, 11.

Roberts, G.O., Rosenthal, J.S., 2001. Optimal scaling for various metropolis-hastings algorithms. Statistical science 16, 351–367.

Roberts, G.O., Rosenthal, J.S., 2007. Coupling and ergodicity of adaptive markov chain monte carlo algorithms. Journal of applied probability 44, 458–475.

Schorfheide, F., 2000. Loss function-based evaluation of dsge models. Journal of Applied Econometrics 15, 645–670.

Smets, F., Wouters, R., 2007. Shocks and frictions in us business cycles: A bayesian dsge approach. American Economic Review 97, 586–606.

Sokal, A., 1997. Monte carlo methods in statistical mechanics: foundations and new algorithms, in: Functional integration. Springer, pp. 131–192.

Storn, R., Price, K., 1997. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Journal of global optimization 11, 341–359.

Ter Braak, C.J., 2006. A markov chain monte carlo version of the genetic algorithm differential evolution: easy bayesian computing for real parameter spaces. Statistics and Computing 16, 239–249.

Tierney, L., 1994. Markov chains for exploring posterior distributions. the Annals of Statistics , 1701–1728.

Vrugt, J.A., Ter Braak, C., Diks, C., Robinson, B.A., Hyman, J.M., Higdon, D., 2009. Accelerating markov chain monte carlo simulation by differential evolution with self-adaptive randomized subspace sampling. International journal of nonlinear sciences and numerical simulation 10, 273–290.

Winberry, T., 2018. A method for solving and estimating heterogeneous agent macro models. Quantitative Economics 9, 1123–1151.

Appendix A Posterior distribution of the estimation of the Smets-Wouters model

	Prior			Proposal		Posterior			
						SW	DIME MCMC		
	distribution	mean	std./df	mean	sd.	mean	mean	sd.	mode
σ_c	normal	1.500	0.375	1.357	0.132	1.38	1.354	0.134	1.434
σ_l	normal	2.000	0.750	1.965	0.579	1.81	1.947	0.570	1.871
β_{tpr}	gamma	0.250	0.100	0.140	0.063	0.16	0.140	0.054	0.132
h	beta	0.700	0.100	0.699	0.051	0.71	0.700	0.053	0.702
S''	normal	4.000	1.500	5.443	1.098	5.74	5.462	1.112	6.835
ι_p	beta	0.500	0.150	0.252	0.103	0.25	0.246	0.103	0.187
ι_w	beta	0.500	0.150	0.571	0.135	0.58	0.574	0.136	0.528
α	normal	0.300	0.050	0.183	0.018	0.19	0.182	0.018	0.195
ζ_p	beta	0.500	0.100	0.664	0.063	0.66	0.658	0.066	0.609
ζ_w	beta	0.500	0.100	0.728	0.071	0.70	0.726	0.068	0.705
Φ_p	normal	1.250	0.125	1.579	0.078	1.60	1.578	0.079	1.544
ψ	beta	0.500	0.150	0.542	0.121	0.54	0.547	0.122	0.487
ϕ_π	normal	1.500	0.250	2.053	0.178	2.04	2.052	0.175	2.068
ϕ_y	normal	0.125	0.050	0.095	0.023	0.08	0.094	0.023	0.106
ϕ_{dy}	normal	0.125	0.050	0.231	0.028	0.22	0.230	0.028	0.201
ρ	beta	0.750	0.100	0.817	0.026	0.81	0.817	0.026	0.810
ρ_r	beta	0.500	0.200	0.113	0.079	0.15	0.112	0.061	0.104
ρ_g	beta	0.500	0.200	0.982	0.011	0.97	0.983	0.008	0.980
ρ_z	beta	0.500	0.200	0.963	0.013	0.95	0.964	0.011	0.968
ρ_u	beta	0.500	0.200	0.264	0.140	0.95	0.259	0.146	0.231
ρ_p	beta	0.500	0.200	0.900	0.070	0.89	0.903	0.072	0.946
ρ_w	beta	0.500	0.200	0.976	0.017	0.96	0.975	0.033	0.989
ρ_i	beta	0.500	0.200	0.728	0.063	0.71	0.727	0.059	0.670
μ_p	beta	0.500	0.200	0.767	0.171	0.69	0.742	0.134	0.664
μ_w	beta	0.500	0.200	0.881	0.061	0.84	0.880	0.066	0.923
ρ_{gz}	normal	0.500	0.250	0.503	0.092	0.52	0.502	0.090	0.515
σ_g	inv.gamma	0.100	2.000	0.532	0.030	0.53	0.532	0.030	0.531
σ_u	inv.gamma	0.100	2.000	1.833	0.615	0.23	1.828	0.486	1.871
σ_z	inv.gamma	0.100	2.000	0.460	0.028	0.45	0.460	0.029	0.459
σ_r	inv.gamma	0.100	2.000	0.243	0.015	0.24	0.243	0.015	0.233
σ_p	inv.gamma	0.100	2.000	0.151	0.027	0.14	0.149	0.032	0.120
σ_w	inv.gamma	0.100	2.000	0.249	0.023	0.24	0.249	0.023	0.276
σ_i	inv.gamma	0.100	2.000	0.448	0.048	0.45	0.448	0.048	0.493
$\overline{\gamma}$	normal	0.400	0.100	0.419	0.020	0.43	0.419	0.020	0.428
$\bar{l}$	normal	0.000	2.000	0.938	1.168	0.53	0.971	1.196	0.906
$\overline{\pi}$	gamma	0.625	0.100	0.673	0.104	0.78	0.670	0.102	0.730

Table A.4: Replication and comparison of the estimation of (Smets and Wouters, 2007, SW) using DIME MCMC. The inverse gamma distribution is parameterized in terms of degrees of freedom as in dynare. The marginals from the proposal distribution is obtained by sampling from the respective multivariate t-distribution in proposal space and then applying the bijective transformation. The mean values of the original estimation (column SW) are obtained from the original paper.

Appendix B Benchmarking against the number of chains

Table B.5 shows Gelman-Rubin coefficients for different ensemble sizes and numbers of cumulative function evaluations. Each measure is the average over the mean across parameters and over ten batches. For each given number of function evaluations n_f (the columns), the sample length is split in half and only the second half is used to calculate the coefficient. E.g., for a given number of function evaluations n_f the sample from iteration $\frac{n_f/n_c}{2}$ to iteration n_f/n_c is used for calculation. The reason is that the Gelman-Rubin coefficient is sensitive to sample length, i.e. the calculation of th Gelman-Rubin coefficient requires much longer chains than a typical sample from the posterior. Note that some more recent work has cast doubt on the reliability of the coefficient to study convergence of MC Markov chains (Flegal et al., 2008).

	1e+05	2e+05	3e+05	4e+05	5e+05	6e+05	7e+05	8e+05	9e+05	1e+06
$n_c = 2n$	1.111	1.094	1.069	1.052	1.044	1.041	1.041	1.036	1.030	1.026
	(0.045)	(0.016)	(0.006)	(0.011)	(0.010)	(0.015)	(0.020)	(0.016)	(0.012)	(0.008)
$n_c = 4n$	1.283	1.247	1.223	1.170	1.146	1.127	1.098	1.088	1.084	1.078
	(0.105)	(0.083)	(0.185)	(0.155)	(0.168)	(0.156)	(0.103)	(0.098)	(0.103)	(0.104)
$n_c = 6n$	1.515	1.488	1.355	1.373	1.319	1.288	1.302	1.265	1.227	1.206
	(0.191)	(0.146)	(0.141)	(0.289)	(0.251)	(0.231)	(0.291)	(0.242)	(0.216)	(0.216)
$n_c = 8n$	1.676	1.633	1.570	1.486	1.436	1.361	1.294	1.264	1.244	1.227
	(0.144)	(0.214)	(0.307)	(0.322)	(0.360)	(0.299)	(0.225)	(0.202)	(0.194)	(0.189)

Table B.5: Gelman-Rubin coefficients over different numbers of function evaluations (per column) and numbers of chains n_c per ensemble. Values are means over the means across parameters over 10 batches. Standard deviations across batches are given in brackets.

Online Appendix (for online-publication only)

Appendix C Details on the HANK model

This part of the model is by large adopted from Auclert et al. (2021).

Appendix C.1 Households

The Bellman equation of households is given by

$$V_t(e_{it}, n_{it-1}, a_{it-1}) = \max_{c_{it}, b_{it}, a_{it}} \left\{ \frac{c_{it}^{1-\sigma}}{1-\sigma} - \varphi \frac{n_t^{1+\nu}}{1+\nu} + \beta E_t V_{t+1}(e_{it+1}, b_{it+1}, a_{it}) \right\} \tag{C.1}$$

such that

$$c_{it} + a_{it} + b_{it} = \frac{(1-\tau_t) w_t n_t}{\int P(e_{jt}) e_{jt}^{1-\Xi} dj} e_{it}^{1-\Xi} + (1 + r_t^a) a_{it-1} + (1 + r_t^b) b_{it-1} - \Phi_t(a_{it}, a_{it-1}), \tag{C.2}$$

$$a_{it} \geq 0, \tag{C.3}$$

$$b_{it} \geq \bar{b}, \tag{C.4}$$

where $\Phi_t(\cdot)$ is the portfolio adjustment cost function

$$\Phi_t(a_{it}, a_{it-1}) = \frac{\chi_1}{\chi_2} \left| \frac{a_{it} - (1 + r_t^a) a_{it-1}}{(1 + r_t^a) a_{it-1} + \chi_0} \right|^{\chi_2} [(1 + r_t^a) a_{it-1} + \chi_0], \tag{C.5}$$

with $\chi_0, \chi_1 > 0$ and $\chi_2 > 1$. Individual labor productivity e_{it} is assumed to follow a random walk process with coefficient ρ_e and a standard deviation of the innovations of σ_t^e, which is by itself assumed to follow an exogenous AR(1) process on an aggregate level. 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.

Appendix C.2 Financial market

No arbitrage at the financial market requires that

$$1 + E_t r_{t+1} = \frac{1 + i_t}{1 + E_t \pi_{t+1}} = \frac{E_t[d_{t+1} + p_{t+1}]}{p_t} = 1 + E_t r_{t+1}^a = 1 + E_t r_{t+1}^b + \omega, \tag{C.6}$$

with ω the parameter governing the cost for liquidity transformation charged by the financial intermediary. Ex-post returns are subject to surprise inflation and capital gains

$$1 + r_t = \frac{1 + i_{t-1}}{1 + \pi_t} = 1 + r_t^b + \omega \tag{C.7}$$

and

$$1 + r_t^a = \Theta_p \left(\frac{d_t + p_t}{p_{t-1}} \right) + (1 - \Theta_p)(1 + r_t), \tag{C.8}$$

where Θ_p denotes the share of equity in the illiquid portfolio.

Appendix C.3 Firms

Firms have a production function

$$y_{jt} = F(k_{jt-1}, n_{jt}) = Z_t k_{jt-1}^{\alpha} n_{jt}^{1-\alpha}, \tag{C.9}$$

and aggregate marginal costs are given by

$$\widehat{MC}_t = w_t / F_N(\cdot), \tag{C.10}$$

which enter the Phillips curve (21). Z_t is the aggregate level of TFP which follows an AR(1) process around its steady state value. Aggregate investment is given by

$$I_t = K_t - (1-\delta)K_{t-1} + S\left(\frac{I_t}{I_{t-1}}\right), \tag{C.11}$$

with the quadratic capital adjustment cost function $S(x) = \frac{1}{2S''}(x-1)^2$ as in the main body, and $\delta > 0$ the parameter for capital depreciation. Dividends are defined as

$$d_t = Y_t - w_t - I_t - \psi_t. \tag{C.12}$$

Tobin's Q and the capital investment decisions follow equations (24) and (25) from the main body.

Appendix C.4 Market clearing

The optimality condition for labor unions is (22) and the monetary policy rule is given by (23). Balanced budget requires

$$\tau_t w_t N_t = r_t B^g + G_t, \tag{C.13}$$

and market clearing requires

$$Y_t = \int c_{it} di + G_t + I_t + \psi_t + \omega b_{it} di, \tag{C.14}$$

$$p_t + B^g = \int a_{it} + b_{it} di. \tag{C.15}$$

Appendix C.5 Fixed parameters

The parameters that are not estimated are set as in table C.6.

Parameter		Value	Target
β	time preference parameter	–	r^*
χ_1	portfolio adj. cost scale	–	$B = 1.04Y$
$\bar{b}$	borrowing constraint	0	
ρ_e	autocorrelation of earnings	0.966	
ν	disutility of labor	–	N = 1
μ_p	steady state markup	–	$p + B^g = 14Y$
μ_w	steady state wage markup	1.1	
Z	TFP	0.468	Y = 1
α	capital share	0.33	K = 10Y
ω	steady state liquidity premium	0.1	
G	steady state government spending	0.2	
B^g	bond supply	2.8	
n_e	points for Markov chain of e	3	
n_b	points for liquid asset grid	25	
n_a	points for illiquid asset grid	35	

Table C.6: Parameters fixed for the estimation of HANK.

Appendix D Data

The following measurement equations are used for the HANK estimation:

$$\begin{aligned}
\text{Real GDP growth} &= \overline{\gamma} + (y_t - y_{t-1}),\\
\text{Real consumption growth} &= \overline{\gamma} + (c_t - c_{t-1}),\\
\text{Real investment growth} &= \overline{\gamma} + (i_t - i_{t-1}),\\
\text{Real wage growth} &= \overline{\gamma} + (w_t - w_{t-1}),\\
\text{Labor hours} &= \overline{n} + n_t,\\
\text{Inflation} &= \overline{\pi} + \pi_t,\\
\text{Federal funds rate} &= 100\left(\frac{\overline{\pi}}{\beta\gamma^{-\sigma_c}} - 1\right) + r_t,
\end{aligned}$$

The observables are constructed as follows:

- GDP: ln(GDP/GDPDEF/CNP16OV_ma)*100
- CONS: ln((PCEC-PCEDG) / GDPDEF / CNP16OV_ma)*100
- INV: ln((GPDI+PCEDG) / GDPDEF / CNP16OV_ma)*100
- LAB: ln(13*AWHNONAG * CE16OV / CNP16OV_ma)*100
- INFL: ln(GDPDEF)*100
- WAGE: ln(COMPNFB / GDPDEF)*100

- FFR: FEDFUNDS/4

Due to artificial dynamics in the civilian noninstitutional population series that arise from irregular updating (Edge et al., 2013), I use a 4-quarter trailing moving average from Boehl et al. (forthcoming), denoted CNP16OV_ma, to calculate per capita variables.

- GDP: GDP - Gross Domestic Product, Billions of Dollars, Quarterly, Seasonally Adjusted Annual Rate, FRED
- GDPDEF: Gross Domestic Product: Implicit Price Deflator , Index 2012=100, Quarterly, Seasonally Adjusted , FRED
- CNP16OV: Civilian noninstitutional population, Thousands of Persons, Quarterly, Seasonally Adjusted, FRED
- CNP16OV_ma: a four-quarter trailing average of CNP16OV
- PCEC: Personal Consumption Expenditures, Billions of Dollars, Quarterly, Seasonally Adjusted Annual Rate, FRED
- PCEDG: Personal Consumption Expenditures: Durable Goods, Billions of Dollars, Quarterly, Seasonally Adjusted Annual Rate, FRED
- GPDI: Gross Private Domestic Investment, Billions of Dollars, Quarterly, Seasonally Adjusted Annual Rate, FRED
- AWHNONAG: Average Weekly Hours of Production and Nonsupervisory Employees: Total private, Hours, Quarterly, Seasonally Adjusted, FRED
- CE16OV: Employment Level, Thousands of Persons, Quarterly, Seasonally Adjusted, FRED
- COMPNFB, Nonfarm Business Sector: Compensation Per Hour, Index 2012=100, Quarterly, Seasonally Adjusted, FRED
- FEDFUNDS: Effective Federal Funds Rate, Percent, FRED

Appendix E Estimation of HANK on a smaller grid

Tables E.7 and E.8 present the estimation results of HANK using a smaller grid than in Section 5. In particular, the number of grid points for the liquid asset is set to $n_b = 10$ (relative to $n_b = 25$ before) and the number of grid points for the illiquid asset is $n_a = 16$ (compared to 35 before). Finally, the state space of capital is represented by 4 nodes instead of 25 nodes. This implies a smaller grid of 480 nodes instead of the 2625 nodes before, which reduces the estimation time about 34 hours to 70 hours.

	Prior			Large grid			Small grid		
	distribution	mean	std.	mean	std.	mode	mean	std.	mode
σ	normal	1.500	0.375	2.043	0.202	1.850	2.153	0.201	2.035
φ	normal	2.000	0.750	1.738	0.562	1.805	1.712	0.518	1.790
ζ_p	beta	0.500	0.100	0.590	0.050	0.592	0.584	0.052	0.608
ζ_w	beta	0.500	0.100	0.416	0.069	0.413	0.422	0.065	0.371
ι_p	beta	0.500	0.150	0.331	0.129	0.335	0.307	0.127	0.254
ι_w	beta	0.500	0.150	0.322	0.147	0.303	0.330	0.145	0.252
S''	gamma	4.000	2.000	2.279	0.702	1.725	2.079	0.632	2.050
ϕ_π	gamma	1.500	0.250	2.322	0.219	2.198	2.204	0.213	2.253
ϕ_y	gamma	0.125	0.050	0.222	0.063	0.205	0.228	0.064	0.272
ρ	beta	0.750	0.100	0.652	0.052	0.680	0.627	0.055	0.613
$\bar{y}$	normal	0.400	0.100	0.438	0.026	0.434	0.434	0.025	0.435
$\bar{n}$	normal	0.000	2.000	-0.047	1.961	1.469	-0.005	1.947	-1.629
π^*	gamma	0.625	0.100	0.596	0.051	0.624	0.594	0.051	0.610
i^*	gamma	1.250	0.100	1.239	0.089	1.259	1.243	0.086	1.218
χ_0	gamma	0.250	0.150	0.153	0.118	0.094	0.118	0.121	0.032
Ξ	beta	0.200	0.100	0.089	0.059	0.071	0.107	0.069	0.126
σ^e	normal	0.920	0.400	0.860	0.185	1.064	0.664	0.107	0.651

Table E.7: Estimation results for HANK with small grid: model parameters

	Prior			Large grid			Small grid		
	distribution	mean	std.	mean	std.	mode	mean	std.	mode
ρ_z	beta	0.500	0.200	0.957	0.018	0.960	0.957	0.016	0.958
ρ_r	beta	0.500	0.200	0.619	0.070	0.640	0.603	0.065	0.629
ρ_g	beta	0.500	0.200	0.993	0.006	0.993	0.991	0.006	0.983
ρ_w	beta	0.500	0.200	0.985	0.006	0.980	0.989	0.004	0.992
ρ_p	beta	0.500	0.200	0.911	0.028	0.917	0.947	0.026	0.959
ρ_i	beta	0.500	0.200	0.837	0.042	0.836	0.788	0.047	0.832
ρ_β	beta	0.500	0.200	0.962	0.030	0.991	0.941	0.044	0.951
σ_z	inv.gamma	0.100	0.250	0.415	0.036	0.430	0.412	0.036	0.396
σ_r	inv.gamma	0.100	0.250	0.130	0.020	0.113	0.138	0.020	0.141
σ_g	inv.gamma	0.100	0.250	1.148	0.088	1.105	1.139	0.089	1.195
σ_w	inv.gamma	0.100	0.250	2.662	0.732	2.448	2.740	0.751	3.270
σ_p	inv.gamma	0.100	0.250	0.201	0.047	0.188	0.205	0.052	0.173
σ_i	inv.gamma	0.100	0.250	1.289	0.215	1.350	1.183	0.204	1.209
σ_β	inv.gamma	0.100	0.250	0.047	0.017	0.029	0.060	0.026	0.051

Table E.8: Estimation results for HANK with small grid: parameters of exogenous processes

Appendix F Details on the estimation of HANK

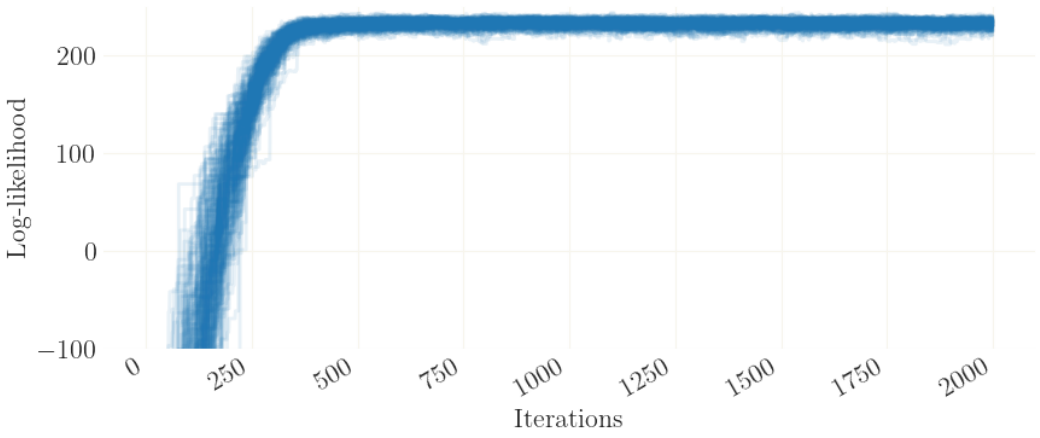

Figure F.6: Traceplot of the log-likelihood of all chains

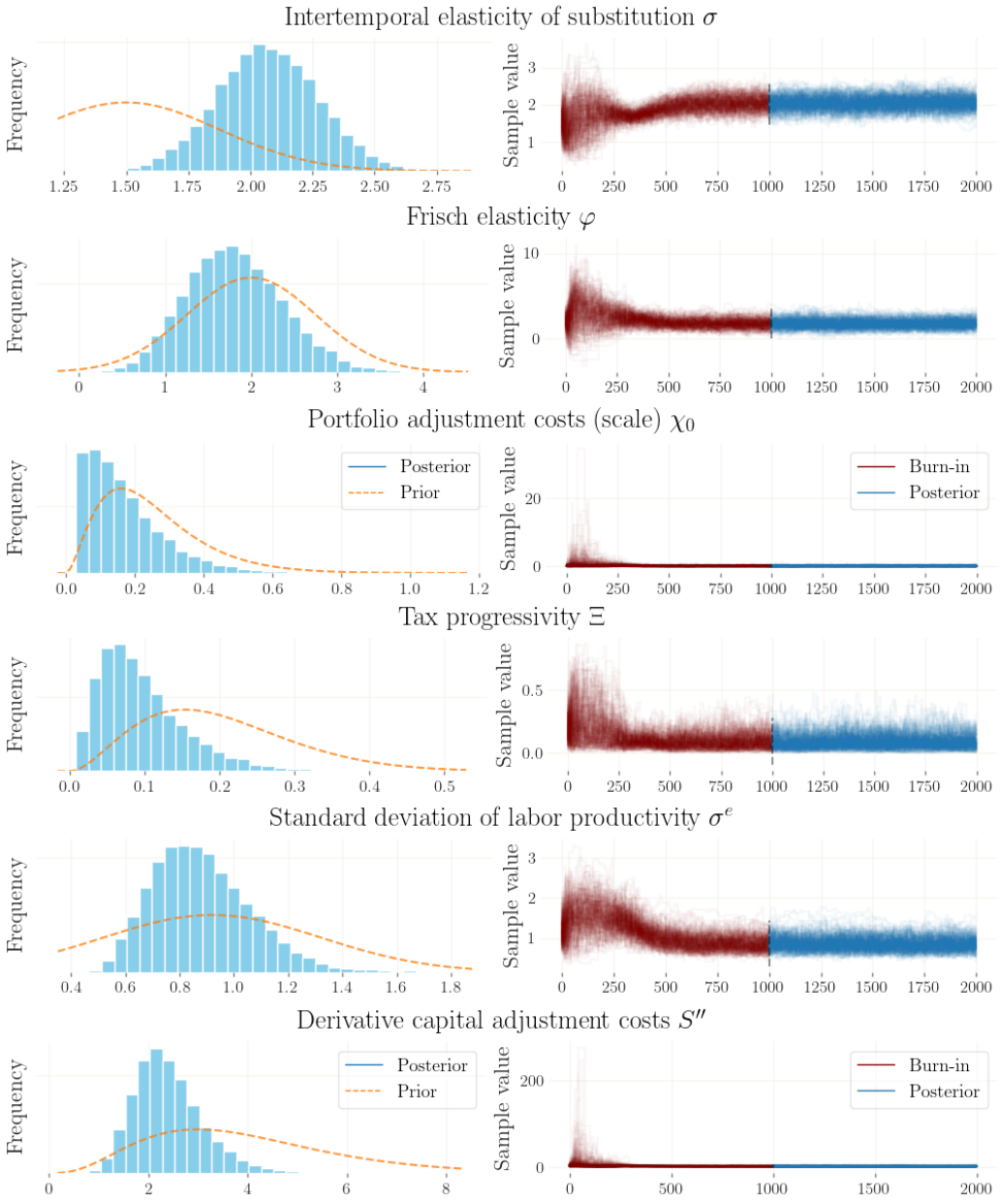

Figure F.7: Traceplots of the 192 DIME chains for the HANK estimation from Section 5. The left panels shows histograms of the marginal distribution over single parameter values. The dashed line plots the respective prior densty. The right panels displays the trace of all chains over time, as corresponding to the parameters.

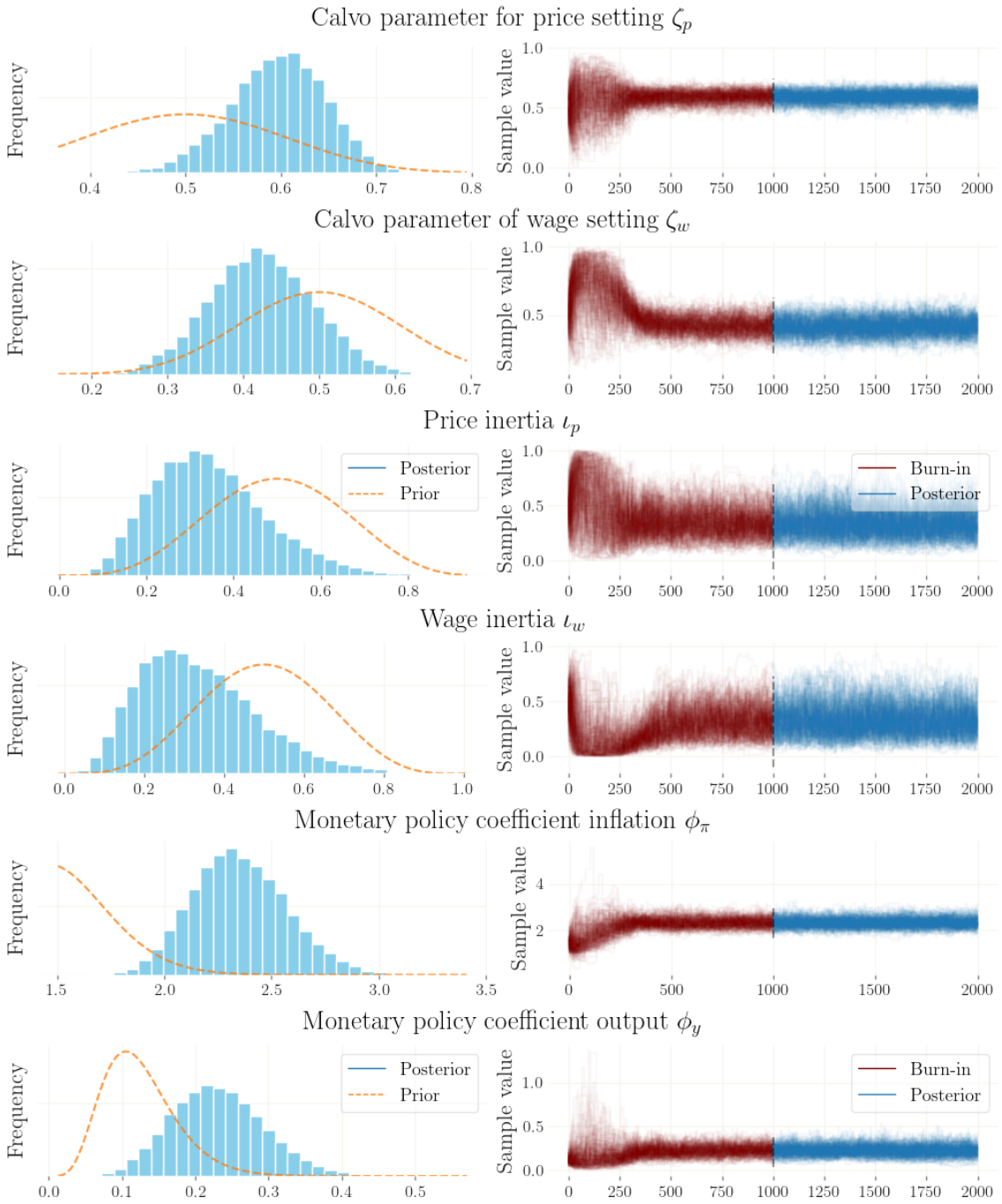

Figure F.8: Traceplots of the 192 DIME chains for the HANK estimation from Section 5. The left panels shows histograms of the marginal distribution over single parameter values. The dashed line plots the respective prior densty. The right panels displays the trace of all chains over time, as corresponding to the parameters.

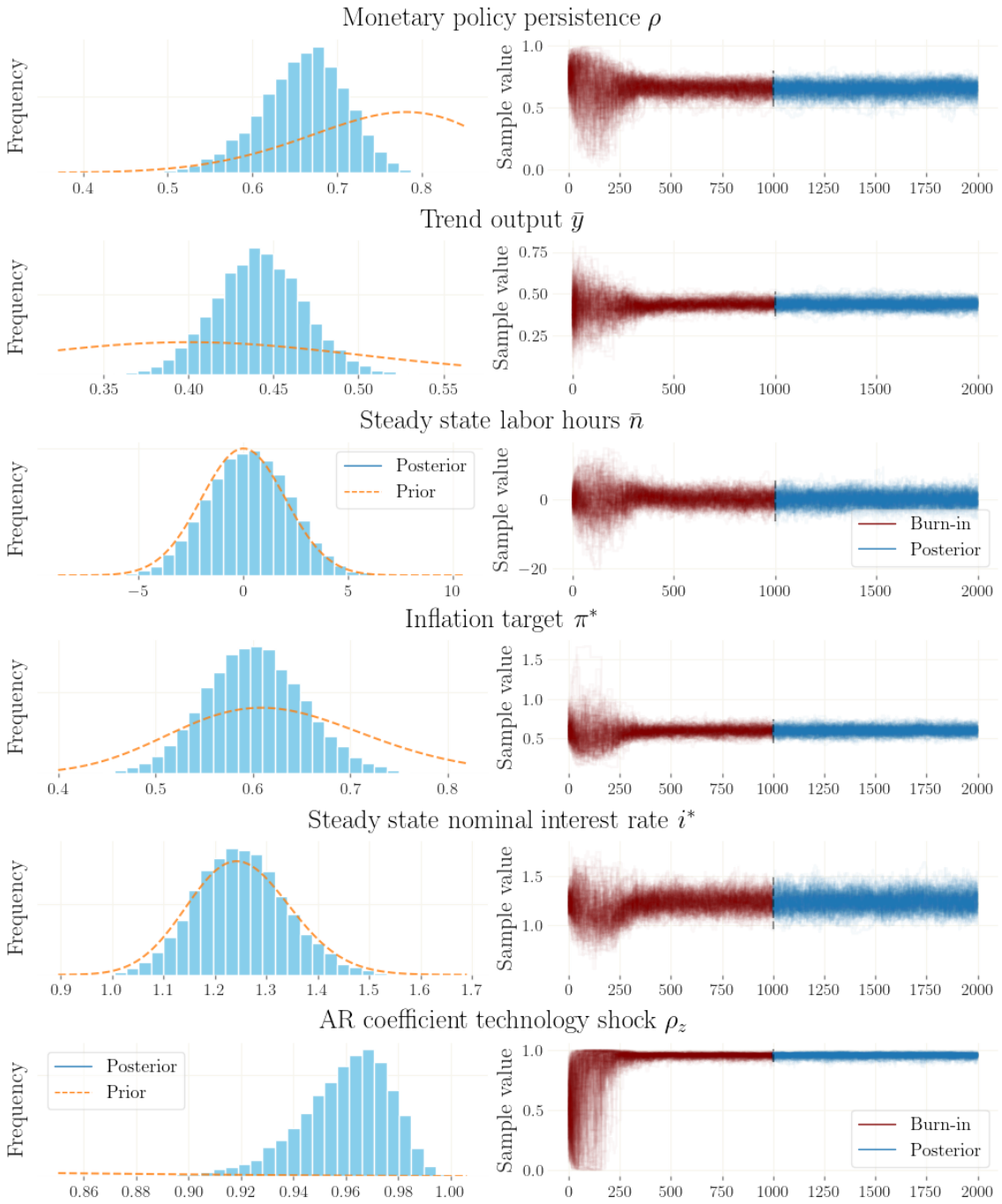

Figure F.9: Traceplots of the 192 DIME chains for the HANK estimation from Section 5. The left panels shows histograms of the marginal distribution over single parameter values. The dashed line plots the respective prior densty. The right panels displays the trace of all chains over time, as corresponding to the parameters.

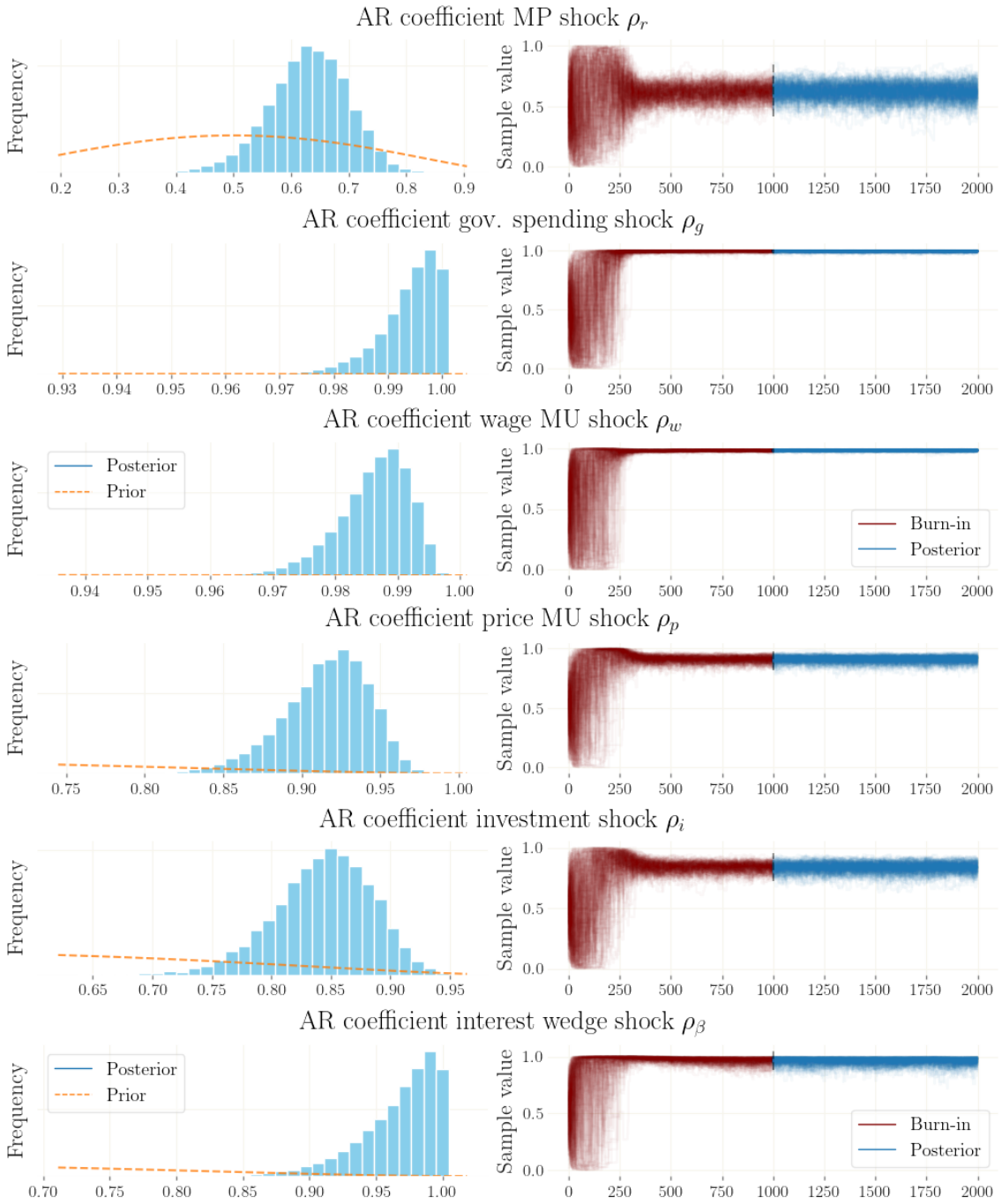

Figure F.10: Traceplots of the 192 DIME chains for the HANK estimation from Section 5. The left panels shows histograms of the marginal distribution over single parameter values. The dashed line plots the respective prior densty. The right panels displays the trace of all chains over time, as corresponding to the parameters.

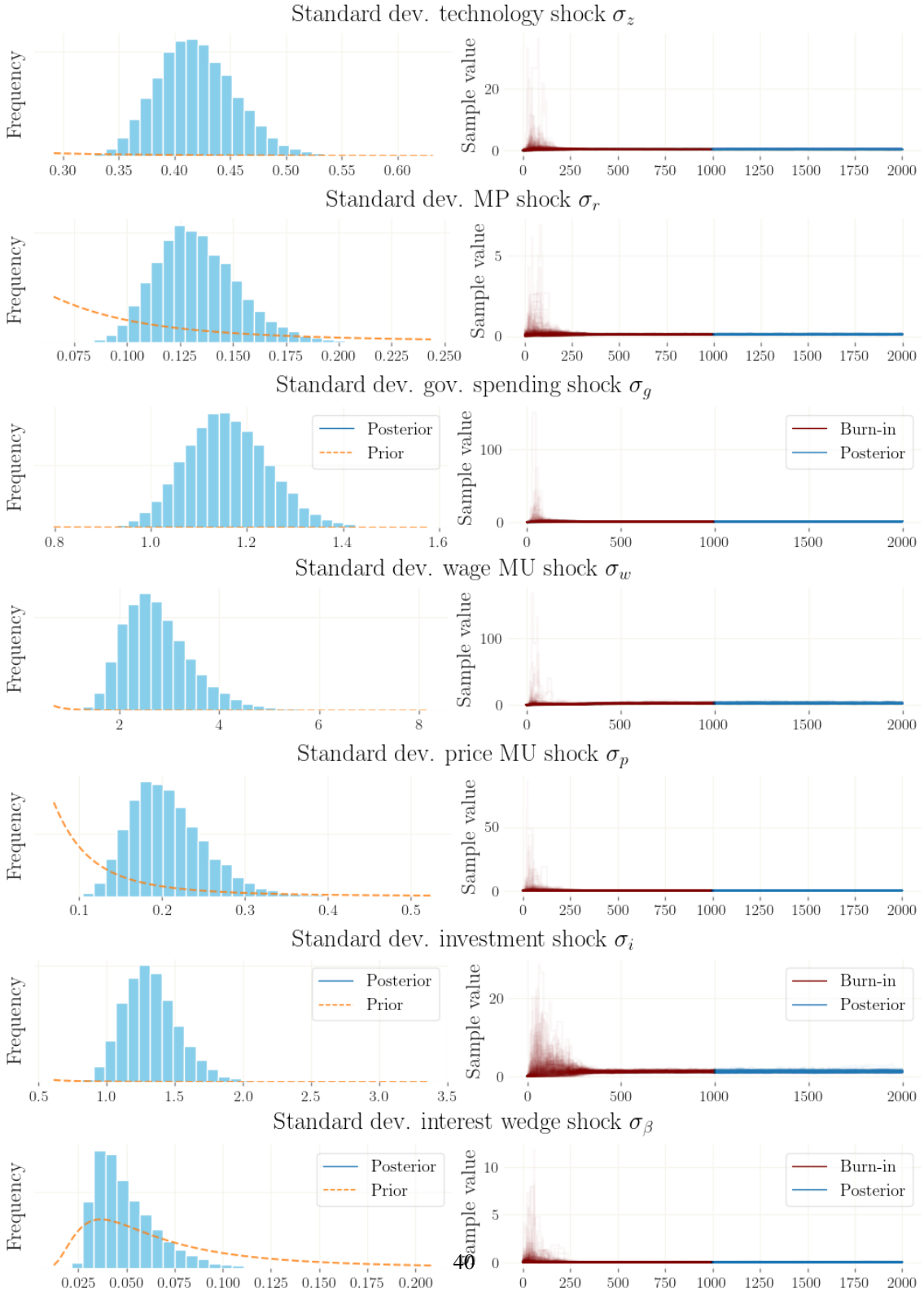

Figure F.11: Traceplots of the 192 DIME chains for the HANK estimation from Section 5. The left panels shows histograms of the marginal distribution over single parameter values. The dashed line plots the respective prior densty. The right panels displays the trace of all chains over time, as corresponding to the parameters.