The Econometrics Journal

Econometrics Journal (2021), volume **24**, pp. C33–C58.
doi: 10.1093/ectj/utaa029

Online estimation of DSGE models

MICHAEL CAI[†], MARCO DEL NEGRO[‡], EDWARD HERBST[§], ETHAN MATLIN[‖], RECA SARFATI[¶] AND FRANK SCHORFHEIDE[#]

[†]*Northwestern University, Department of Economics, 2211 Campus Dr, Evanston, IL 60208, United States.*
Email: michaelcai@u.northwestern.edu

[‡]*Federal Reserve Bank of New York, Research Department, 33 Liberty St, New York, NY 10045, United States.*
Email: marco.delnegro@ny.frb.org *(corresponding author)*

[§]*Board of Governors of the Federal Reserve System, 20th Street and Constitution Avenue N.W., Washington, DC 20551, United States.*
Email: edward.p.herbst@frb.gov

[‖]*Harvard University, Department of Economics, 1805 Cambridge St, Cambridge, MA 02138, United States.*
Email: ethanmatlin@gmail.com

[¶]*Massachusetts Institute of Technology, Department of Economics, 77 Massachusetts Ave, Cambridge, MA 02139, United States.*
Email: sarfati@mit.edu

[#]*University of Pennsylvania, Department of Economics, 33 South 36th Street Philadelphia, Pennsylvania 19104-6297, United States.*
Email: schorf@ssc.upenn.edu

First version received: 23 July 2019; final version accepted: 14 June 2020.

Summary: This paper illustrates the usefulness of sequential Monte Carlo (SMC) methods in approximating dynamic stochastic general equilibrium (DSGE) model posterior distributions. We show how the tempering schedule can be chosen adaptively, document the accuracy and runtime benefits of generalized data tempering for 'online' estimation (that is, re-estimating a model as new data become available), and provide examples of multimodal posteriors that are well captured by SMC methods. We then use the online estimation of the DSGE model to compute pseudo-out-of-sample density forecasts and study the sensitivity of the predictive performance to changes in the prior distribution. We find that making priors less informative (compared with the benchmark priors used in the literature) by increasing the prior variance does not lead to a deterioration of forecast accuracy.

Keywords: *Adaptive algorithms*, *Bayesian inference*, *density forecasts*, *online estimation*, *sequential Monte Carlo methods*.

JEL codes: *C11*, *C32*, *C53*, *E32*, *E37*, *E52*.

1. INTRODUCTION

The goal of this paper is to provide a framework for performing 'online' estimation of Bayesian dynamic stochastic general equilibrium (DSGE) models using sequential Monte Carlo (SMC) techniques. We borrow the term online estimation from the statistics and machine-learning literature to describe the task of re-estimating a model frequently as new data become available. This is standard practice in central bank settings, where models may be re-estimated every one to three months for each policy-briefing cycle or after each data release. The conventional approach—to run a completely new estimation to update the model with new information—is time-consuming and can require considerable user supervision. Moreover, as macroeconomic models become more complex, for example the current vintage of heterogeneous-agent New Keynesian models, even seemingly generous time constraints may become binding. Online estimation is also important for academic research, where economists often compare predictions from a variety of models in pseudo-out-of-sample forecast exercises that require recursive estimation.

In both policy and academic settings, then, there is a need for reliable and efficient estimation algorithms. In this paper, we propose a generic SMC algorithm for online estimation that minimizes computational time but maintains the ability to handle complex, i.e., multimodal, posterior distributions with minimal user monitoring. In an empirical section, we verify the properties of the algorithm by estimating a suite of DSGE models. In an application prototypical for both policy institutions and academic research, we use the algorithm to generate recursive forecasts using these DSGE models.

In online estimation applications of SMC methods, parameter estimates based on data available in the previous period will be adjusted to capture the additional information contained in the observations from the current period. However, a similar technique can be used to transform estimates of, say, a linearized DSGE model, into estimates of a version of the model that has been solved with nonlinear techniques and that shares the same set of parameters. Thus, our framework should be of value to anyone interested in estimating complex models in a stepwise fashion, either by sequentially increasing the sample information or by mutating preliminary estimates from a relatively simple model (linear dynamics, heterogeneous-agent models with a coarse approximation), which can be computed quickly, into estimates of a more complex model (nonlinear dynamics, heterogeneous-agent models with a finer approximation), which would take a long time to compute from scratch.

SMC methods have traditionally been used to solve nonlinear filtering problems, an example being the bootstrap particle filter of Gordon et al. (1993). Subsequently, Chopin (2002) showed how to adapt particle filtering techniques to conduct posterior inference for a static parameter vector. The first paper that applied SMC techniques to posterior inference for the parameters of a (small-scale) DSGE model was Creal (2007). Subsequent work by Herbst and Schorfheide (2014, 2015) fine-tuned the algorithm so that it could be used for the estimation of medium- and large-scale models.

In order to frame the paper's contributions, a brief summary of how SMC works is in order. SMC algorithms approximate a target posterior distribution by creating intermediate approximations to a sequence of bridge distributions, indexed in this paper by n. At each stage, the current bridge

We thank participants at various conferences and seminars for helpful comments. We also thank two anonymous referees for their feedback, and Jaap Abbring, the editor, and especially Jeff Campbell, the special issue editor, for their thoughful guidance. FS acknowledges financial support from the National Science Foundation under Grant SES 1851634. The views expressed in this paper are those of the authors and do not necessarily reflect the position of the Federal Reserve Bank of New York, the Federal Reserve Board or the Federal Reserve System.

distribution is represented by a swarm of so-called particles. Each particle is composed of a value and a weight. Weighted averages of the particle values converge to expectations under the stage-n distribution. The transition from stage $n-1$ to n involves changing the particle weights and values so that the swarm adapts to the new distribution. Typically, these bridge distributions are composed either by using the full-sample likelihood (*likelihood tempering*)—generated by raising this likelihood function to the power of ϕ_n, where ϕ_n increases from zero to one—or by sequentially adding observations to the likelihood function (*data tempering*). While the data tempering approach seems most natural for an online estimation algorithm, previous work (Herbst and Schorfheide, 2015) has shown that it may perform poorly relative to likelihood tempering.

This paper makes three main contributions. First, under likelihood tempering, we replace a predetermined (or fixed) tempering schedule $\{\phi_n\}$ for the DSGE model likelihood function by a schedule that is constructed adaptively. The adaptive tempering schedule chooses the amount of information that is added to the likelihood function in stage n to achieve a particular variance of the particle weights. While adaptive tempering schedules have been used in the statistics literature before (e.g., Jasra et al. 2011), their use for the estimation of DSGE model parameters is new. This kind of adaptation is an important prerequisite for efficient online estimation, as it avoids unnecessary computations. Our adaptive schedules are calibrated by a single tuning parameter that controls the desired variance of the particle weights. We assess how this tuning parameter affects the accuracy–runtime trade-off for the algorithm.

Second, we modify the SMC algorithm so that the initial particles are drawn from a previously computed posterior distribution instead of from the prior distribution. This initial posterior can result from estimating the model on a shorter sample or from a simpler version of the same model (e.g., linear versus nonlinear, as discussed above) on the full sample. In the former case, our approach can be viewed as a form of generalized data tempering. Our approach is more general in that it allows users to add information from fractions of observations and accommodates data revisions, which are pervasive in macro applications. When combined with adaptive tempering, this generalized approach avoids the pitfalls associated with standard data tempering. This is because in the periods in which the new observation(s) are quite informative and shift the posterior distribution substantially, the algorithm will use a larger number of intermediate stages to reach the new posterior to maintain accuracy. In other periods, where the posterior distribution remains essentially unchanged, however, the number of intermediate stages is kept small to reduce runtime.

Third, we contribute to the literature that assesses the real-time pseudo-out-of-sample forecast performance of DSGE models. Here *real-time* means that for a forecast using a sample ending at time t, the data vintage used to estimate the model is one that would have been available to the econometrician at the time. *Pseudo-out-of-sample* means that the forecasts were produced ex post.[1] We use the proposed SMC techniques to conduct online estimation of the Smets and Wouters (2007) model and a version of this model with financial frictions. Our forecast evaluation exercises extend previous results in Del Negro and Schorfheide (2013) and Cai et al. (2019), which were conducted with parameter estimates from the widely used random walk Metropolis Hastings (RWMH) algorithm. In particular, we study the effects of reducing the informativeness of the prior distribution on forecasting performance. Despite the emergence of multiple modes in the posterior distribution, our SMC-based results show that the large increase in the prior standard deviation has surprisingly small effects on forecasting accuracy, thereby debunking the notion that priors in DSGE models are chosen to improve the model's predictive ability.

[1] Cai et al. (2019) provide a genuine real-time forecast evaluation that uses the NY Fed DSGE model's forecasts.

The remainder of this paper is organized as follows. In Section 2 we outline the basic structure of an SMC algorithm designed for posterior inference on a time-invariant parameter vector θ. We review different tempering approaches and present an algorithm for the adaptive choice of the tempering schedule. Section 3 provides an overview of the DSGE models that are estimated in this paper. In Section 4 we study various dimensions of the performance of SMC algorithms: we assess the accuracy and runtime trade-offs of adaptive tempering schedules, we document the benefits of generalized data tempering for online estimation, and we demonstrate the ability of SMC algorithms to accurately approximate multimodal posteriors. Section 5 contains various pseudo-out-of-sample forecasting assessments for models that are estimated by SMC. Finally, Section 6 concludes. The Online Appendix provides further details on model specifications, prior distributions, and computational aspects. It also contains additional empirical results.

2. ADAPTIVE SMC ALGORITHMS FOR POSTERIOR INFERENCE

SMC techniques to generate draws from posterior distributions of a static parameter θ are emerging as an attractive alternative to Markov chain Monte Carlo (MCMC) methods. SMC algorithms can be easily parallelized and, properly tuned, may produce more accurate approximations of posterior distributions than MCMC algorithms. Chopin (2002) showed how to adapt particle filtering techniques to conduct posterior inference for a static parameter vector. Textbook treatments of SMC algorithms are provided by, for instance, Liu (2001) and Cappé et al. (2005). This section reviews the standard SMC algorithm (Subsection 2.1), contrasts our generalized tempering approach with existing alternatives (Subsection 2.2), and finally describes our adaptive tempering algorithm (Subsection 2.3).

The first paper that applied SMC techniques to posterior inference in small-scale DSGE models was Creal (2007). Herbst and Schorfheide (2014) develop the algorithm further, provide some convergence results for an adaptive version of the algorithm building on the theoretical analysis of Chopin (2004), and show that a properly tailored SMC algorithm delivers more reliable posterior inference for large-scale DSGE models with a multimodal posterior than the standard RWMH algorithm. Creal (2012) provides a recent survey of SMC applications in econometrics. Durham and Geweke (2014) show how to parallelize a flexible and self-tuning SMC algorithm for the estimation of time series models on graphical processing units (GPUs). The remainder of this section draws heavily from the more detailed exposition in Herbst and Schorfheide (2014, 2015).

2.1. *SMC algorithms for posterior inference*

SMC combines features of classic importance sampling and modern MCMC techniques. The starting point is the creation of a sequence of intermediate or bridge distributions $\{\pi_n(\theta)\}_{n=0}^{N_\phi}$ that converge to the target posterior distribution, i.e., $\pi_{N_\phi}(\theta) = \pi(\theta)$. At any stage, the (intermediate) posterior distribution $\pi_n(\theta)$ is represented by a swarm of particles $\{\theta_n^i, W_n^i\}_{i=1}^N$ in the sense that the Monte Carlo average

$$\bar{h}_{n,N} = \frac{1}{N}\sum_{i=1}^{N} W_n^i h(\theta^i) \xrightarrow{a.s.} \mathbb{E}_{\pi_n}[h(\theta_n)],$$

as $N \longrightarrow \infty$, for each $n = 0, \ldots, N_\phi$. The bridge distributions are posterior distributions constructed from stage-n likelihood functions:

$$\pi_n(\theta) = \frac{p_n(Y|\theta)p(\theta)}{\int p_n(Y|\theta)p(\theta)d\theta},$$

with the convention that $p_0(Y|\theta) = 1$, i.e., the intial particles are drawn from the prior, and $p_{N_\phi}(Y|\theta) = p(Y|\theta)$. The actual forms of the likelihood sequences depend on the tempering approach and will be discussed in Subsection 2.2 below. We adopt the convention that the weights W_n^i are normalized to average to one.

The SMC algorithm proceeds iteratively from $n = 0$ to $n = N_\phi$. Starting from stage $n-1$ particles $\{\theta_{n-1}^i, W_{n-1}^i\}_{i=1}^N$, each stage n of the algorithm targets the posterior π_n and consists of three steps: *correction*, that is, reweighting the stage $n-1$ particles to reflect the density in iteration n; *selection*, that is, eliminating a highly uneven distribution of particle weights (degeneracy) by resampling the particles; and *mutation*, that is, propagating the particles forward using a Markov transition kernel to adapt the particle values to the stage n bridge density.

ALGORITHM 1 (GENERIC SMC ALGORITHM).

(1) **Initialization.** ($n = 0$ and $\phi_0 = 0$.) Draw the initial particles from the prior: $\theta_1^i \overset{iid}{\sim} p(\theta)$ and $W_1^i = 1, i = 1, \ldots, N$.
(2) **Recursion.** For $n = 1, \ldots, N_\phi$,
   (a) **Correction.** Reweight the particles from stage $n-1$ by defining the incremental weights
   $$\tilde{w}_n^i = \frac{p_n(Y|\theta_{n-1}^i)}{p_{n-1}(Y|\theta_{n-1}^i)},$$
   and the normalized weights
   $$\tilde{W}_n^i = \frac{\tilde{w}_n^i W_{n-1}^i}{\frac{1}{N}\sum_{i=1}^N \tilde{w}_n^i W_{n-1}^i}, \quad i = 1, \ldots, N.$$
   (b) **Selection (Optional).** Resample the swarm of particles, $\{\theta_{n-1}^i, \tilde{W}_n^i\}_{i=1}^N$, and denote resampled particles by $\{\hat{\theta}_n^i, W_n^i\}_{i=1}^N$, where $W_n^i = 1$ for all i.
   (c) **Mutation.** Starting from $\hat{\theta}_n^i$, propagate the particles $\{\hat{\theta}_n^i, W_n^i\}$ via N_{MH} steps of a Metropolis–Hastings (MH) algorithm with transition density $K_n(\theta|\tilde{\theta}; \zeta_n)$ and stationary distribution $\pi_n(\theta)$. Note that the weights are unchanged, and denote the mutated particles by $\{\theta_n^i, W_n^i\}_{i=1}^N$.

   An approximation of $\mathbb{E}_{\pi_n}[h(\theta)]$ is given by
   $$\bar{h}_{n,N} = \frac{1}{N}\sum_{i=1}^N h(\theta_n^i)W_n^i.$$
(3) For $n = N_\phi$ ($\phi_{N_\phi} = 1$), the final importance sampling approximation of $\mathbb{E}_\pi[h(\theta)]$ is given by:
   $$\bar{h}_{N_\phi,N} = \sum_{i=1}^N h(\theta_{N_\phi}^i)W_{N_\phi}^i.$$

Because we are using a proper prior, we initialize the algorithm with iid draws from the prior density $p(\theta)$. The correction step is a classic importance sampling step, in which the particle weights are updated to reflect the stage n distribution $\pi_n(\theta)$. The selection step is optional. On the one hand, resampling adds noise to the Monte Carlo approximation, which is undesirable. On the

other, it equalizes the particle weights, which increases the accuracy of subsequent importance sampling approximations. The decision of whether or not to resample is typically based on a threshold rule for the variance of the particle weights. This variance can be transformed into an effective particle sample size:

$$\widehat{ESS}_n = N \Big/ \left(\frac{1}{N} \sum_{i=1}^{N} (\tilde{W}_n^i)^2 \right).$$

If the particles have equal weights, then $\widehat{ESS}_n = N$. If one particle has weight N and all other particles have weight 0, then $\widehat{ESS}_n = 1$. These are the upper and lower bounds for the effective sample size. To balance the trade-off between adding noise and equalizing particle weights, the resampling step is typically executed if $\widehat{ESS}_n$ falls below a threshold $\underline{N}$, for example $N/2$ or $N/3$. An overview of specific resampling schemes is provided in, for instance, the books by Liu (2001) and Cappé et al. (2005) (and references cited therein). We are using systematic resampling in the applications below.

The mutation step changes the particle values. In the absence of the mutation step, the particle values would be restricted to the set of values drawn in the initial stage from the prior distribution. This would clearly be inefficient, because the prior distribution is typically a poor proposal distribution for the posterior in an importance sampling algorithm. As the algorithm cycles through the N_ϕ stages, the particle values successively adapt to the shape of the posterior distribution. This is the key difference between SMC and classic importance sampling. The transition kernel $K_n(\theta|\tilde{\theta}; \zeta_n)$ is designed to have the following invariance property:

$$\pi_n(\theta_n) = \int K_n(\theta_n|\hat{\theta}_n; \zeta_n)\pi_n(\hat{\theta}_n)d\hat{\theta}_n.$$

Thus, if $\hat{\theta}_n^i$ is a draw from π_n, then so is θ_n^i. The mutation step can be implemented by using one or more steps of a MH algorithm. The probability of mutating the particles can be increased by blocking the elements of the parameter vector θ or by iterating the MH algorithm over multiple steps. The vector ζ_n summarizes the tuning parameters of the MH algorithm.

The SMC algorithm produces as a by-product an approximation of the marginal likelihood. Note that

$$\frac{1}{N} \sum_{i=1}^{N} \tilde{w}_n^i \tilde{W}_{n-1}^i \approx \int \frac{p_n(Y|\theta)}{p_{n-1}(Y|\theta)} \left[\frac{p_{n-1}(Y|\theta)p(\theta)}{\int p_{n-1}(Y|\theta)p(\theta)d\theta} \right] d\theta = \frac{\int p_n(Y|\theta)p(\theta)d\theta}{\int p_{n-1}(Y|\theta)p(\theta)d\theta}.$$

Thus, it can be shown that the approximation

$$\hat{p}(Y) = \prod_{n=1}^{N_\phi} \left(\frac{1}{N} \sum_{i=1}^{N} \tilde{w}_n^i W_{n-1}^i \right), \tag{2.1}$$

converges almost surely to $p(Y)$ as the number of particles $N \longrightarrow \infty$; see, for instance, Herbst and Schorfheide (2014).

2.2. *Likelihood, data, and generalized tempering*

The stage n likelihood functions are generated in different ways. Under likelihood tempering, one takes power transformations of the entire likelihood function:

$$p_n(Y|\theta) = [p(Y|\theta)]^{\phi_n}, \quad \phi_n \uparrow 1. \tag{2.2}$$

The advantage of likelihood tempering is that one can make, through the choice of ϕ_n, consecutive posteriors arbitrarily 'close' to one another. Under data tempering, sets of observations are gradually added to the likelihood function, that is,

$$p_n(Y|\theta) = p(y_{1:\lfloor \phi_n T \rfloor}|\theta), \quad \phi_n \uparrow 1,$$

where $\lfloor x \rfloor$ is the largest integer that is less than or equal to x. Data tempering is particularly attractive in time series applications. But because individual observations are not divisible, the data tempering approach is less flexible. Though the data tempering approach might seem well suited for online estimation, in practice it performs poorly because adding an observation can change the posterior substantially. Moreover, conventional data tempering is not easily adapted to revised data.

Our approach generalizes both likelihood and data tempering as follows. Imagine that one has draws from the posterior

$$\tilde{\pi}(\theta) \propto \tilde{p}(\tilde{Y}|\theta)p(\theta),$$

where the posterior $\tilde{\pi}(\theta)$ differs from the posterior $\pi(\theta)$ because either the sample (Y versus $\tilde{Y}$), or the model ($p(Y|\theta)$ versus $\tilde{p}(\tilde{Y}|\theta)$), or both are different.[2] We define the stage n likelihood function as

$$p_n(Y|\theta) = [p(Y|\theta)]^{\phi_n}[\tilde{p}(\tilde{Y}|\theta)]^{1-\phi_n}, \quad \phi_n \uparrow 1. \tag{2.3}$$

First, if one sets $\tilde{p}(\cdot) = 1$, then (2) is identical to likelihood tempering. Second, suppose one sets $\tilde{p}(\cdot) = p(\cdot)$, $Y = y_{1:T}$, and $\tilde{Y} = y_{1:T_1}$, where $T > T_1$. Then, tempering this likelihood allows for a gradual transition from $p(y_{1:T_1}|\theta)$ to $p(y_{1:T}|\theta)$ as ϕ_n increases from 0 to 1. This leads to a generalized version of data tempering in which we can add informational content to the likelihood that corresponds to a fraction of an observation y_t. This may be important if the additional sample $y_{T_1+1:T}$ substantially affects the likelihood (e.g., $y_{T_1+1:T}$ includes the Great Recession).

Third, by allowing Y to differ from $\tilde{Y}$, we can accommodate data revisions between time T_1 and T. For online estimation, one can use the most recent estimation to jump-start a new estimation on revised data, without starting from scratch. Finally, by allowing $p(\cdot)$ and $\tilde{p}(\cdot)$ to differ, one can transition between the posterior distributions of two models that share the same parameters, e.g., DSGE models solved by a first- and second-order perturbation method. We will evaluate the accuracy of the generalized tempering approach in Section 4 and use it in the real-time forecast evaluation of Section 5.

2.3. Adaptive algorithms

The implementation of the SMC algorithm requires the choice of several tuning constants. First, the user has to choose the number of particles N. As shown in Chopin (2004), Monte Carlo averages computed from the output of the SMC algorithm satisfy a central limit theoremas the number of particles increases to infinity. This means that the variance of the Monte Carlo approximation decreases at the rate $1/N$. Second, the user has to determine the tempering schedule ϕ_n and the number of bridge distributions N_ϕ. Third, the threshold level $\underline{N}$ for $\widehat{ESS}_n$ needs to be set to determine whether the resampling step should be executed in iteration n. Finally, the implementation of the mutation step requires the choice of the number of MH steps (N_{MH}), the

[2] It is straightforward to generalize our approach to encompass differences in the prior.

number of blocks into which the parameter vector θ is partitioned (N_{blocks}), and the parameters ζ_n that control the Markov transition kernel $K_n(\theta_n|\hat{\theta}_n^i;\zeta_n)$.

Our implementation of the algorithm starts from a choice of N, $\underline{N}$, N_{MH}, and N_{blocks}. The remaining features of the algorithm are determined adaptively. As in Herbst and Schorfheide (2015), we use a RWMH algorithm to implement the mutation step. The proposal density takes the form $N(\hat{\theta}^i, c_n^2, \tilde{\Sigma}_n)$. The scaling constant c and the covariance matrix $\tilde{\Sigma}_n$ can be easily chosen adaptively; see Herbst and Schorfheide (2015, Algorithm 10). Based on the MH rejection frequency, c can be adjusted to achieve a target rejection rate of approximately 25–40%. For $\tilde{\Sigma}_n$ one can use an approximation of the posterior covariance matrix computed at the end of the stage n correction step.

In the current paper, we will focus on the adaptive choice of the tempering schedule, building on work by Jasra et al. (2011), Del Moral et al. (2012), Schäfer and Chopin (2013), Geweke and Frischknecht (2014), and Zhou et al. (2016). The key idea is to choose ϕ_n to target a desired level $\widehat{ESS}_n^*$. Roughly, the closer the desired $\widehat{ESS}_n^*$ is to the previous $\widehat{ESS}_{n-1}$, the smaller the increment $\phi_n - \phi_{n-1}$ and therefore the smaller the information increase in the likelihood function. In order to formally describe the choice of ϕ_n, we define the functions

$$w^i(\phi) = [p(Y|\theta_{n-1}^i)]^{\phi-\phi_{n-1}}, \quad \tilde{W}_n^i(\phi) = \frac{w^i(\phi)W_{n-1}^i}{\frac{1}{N}\sum_{i=1}^N w^i(\phi)W_{n-1}^i}, \quad \widehat{ESS}(\phi) = N \Big/ \left(\frac{1}{N}\sum_{i=1}^N (\tilde{W}_n^i(\phi))^2\right).$$

We will choose ϕ to target a desired level of ESS: (3)

$$f(\phi) = \widehat{ESS}(\phi) - \alpha\widehat{ESS}_{n-1} = 0, \tag{2.4}$$

where α is a tuning constant that captures the targeted reduction in ESS. For instance, if $\alpha = 0.95$, then the algorithm allows for a 5% reduction in the effective sample size at each stage. The algorithm can be summarized as follows.

ALGORITHM 2 (ADAPTIVE TEMPERING SCHEDULE).

(1) If $f(1) \geq 0$, then set $\phi_n = 1$.
(2) If $f(1) < 0$, let $\phi_n \in (\phi_{n-1}, 1)$ be the smallest value of ϕ_n such that $f(\phi_n) = 0$.

It is important to note that Algorithm 2 is guaranteed to generate a 'well-formed'—monotonically increasing—tempering schedule. To see this, suppose that $\phi_{n-1} < 1$. First, note that $f(\phi_{n-1}) = (1-\alpha)\widehat{ESS}_{n-1} > 0$. Second, if $f(1) \geq 0$, we set $\phi_n = 1 > \phi_{n-1}$ and the algorithm terminates. Alternatively, if $f(1) < 0$, then by continuity of $f(\phi)$ and the compactness of the interval $[\phi_{n-1}, 1]$, there exists at least one root of $f(\phi) = 0$. We define ϕ_n to be the smallest one. The formulation in Algorithm 2 is also attractive because the values of the likelihood function used in (3) have already been stored in memory. This means that even exhaustive root-finding methods will typically find ϕ_n quickly.[3]

[3] A refined version of Algorithm 2 that addresses potential numerical challenges in finding the root of $f(\phi)$ is provided in Subsection A.2 of the Online Appendix.

3. DSGE MODELS

In the subsequent applications we consider three DSGE models. The precise specifications of these models, including their linearized equilibrium conditions, measurement equations, and prior distributions, are provided in Section B of the Online Appendix.

The first model is a small-scale New Keynesian DSGE model that has been widely studied in the literature (see Woodford, 2003, or Galí, 2008, for textbook treatments). The particular specification used in this paper is based on the one in An and Schorfheide (2007), henceforth AS. The model economy consists of final goods producing firms, intermediate goods producing firms, households, a central bank, and a fiscal authority. Labour is the only factor of production. Intermediate goods producers act as monopolistic competitors and face downward-sloping demand curves for their products. They face quadratic costs for adjusting their nominal prices, which generates price rigidity and real effects of changes in monetary policy. The model solution can be reduced to three key equations: a consumption Euler equation, a New Keynesian Phillips curve, and a monetary policy rule. Fluctuations are driven by three exogenous shocks, and the model is estimated based on output growth, inflation, and federal funds rate data.

The second model is the Smets and Wouters (2007) model, henceforth SW, which is based on earlier work by Christiano et al. (2005) and Smets and Wouters (2003), and is the prototypical medium-scale New Keynesian model. In the SW model, capital is a factor of intermediate goods production, and, in addition to price stickiness, the model also features nominal wage stickiness. In order to generate a richer autocorrelation structure, the model includes investment adjustment costs, habit formation in consumption, and partial dynamic indexation of prices and wages to lagged values. The SW model is estimated using the following seven macroeconomic time series: output growth, consumption growth, investment growth, real wage growth, hours worked, inflation, and the federal funds rate.[4]

The third DSGE model, SWFF, is obtained by extending the SW model in two dimensions. First, building on work by Bernanke et al. (1999), Christiano et al. (2003), De Graeve (2008), and Christiano et al. (2014), we add financial frictions to the SW model. Banks collect deposits from households and lend to entrepreneurs who use these funds as well as their own wealth to acquire physical capital, which is rented to intermediate goods producers. Entrepreneurs are subject to idiosyncratic disturbances that affect their ability to manage capital. Their revenue may thus be too low to pay back the bank loans. Banks protect themselves against default risk by pooling all loans and charging a spread over the deposit rate. This spread varies exogenously, owing to changes in the riskiness of entrepreneurs' projects, and endogenously as a function of the entrepreneurs' leverage. In estimating the model, we use the Baa-10-year Treasury spread as the observable corresponding to this spread. Second, we include a time-varying target inflation rate to capture low-frequency movements of inflation. To anchor the estimates of the target inflation rate, we include long-run inflation expectations into the set of observables.[5]

[4] To generate forecasts between 2009 and 2015 that do not violate the zero-lower-bound constraint on nominal interest rates, we add anticipated monetary policy shocks to the interest rate feedback rules and include (survey) expectations of future interest rates at the forecast origin as observables.

[5] Del Negro and Schorfheide (2013) showed that introducing a time-varying inflation target into the SW model improves inflation forecasts.

Table 1. Configuration of the SMC algorithm for the two models.

	AS	SW
Number of particles	$N = 3,000$	$N = 12,000$
Fixed tempering schedule	$N_\phi = 200, \lambda = 2$	$N_\phi = 500, \lambda = 2.1$
Mutation	$N_{blocks} = 3$	$N_{blocks} = 3$
Selection/Resampling	$\underline{N} = N/2$	$\underline{N} = N/2$

4. SMC ESTIMATION AT WORK

We now illustrate various dimensions of the performance of the SMC algorithm. Subsection 4.1 documents the shape of the adaptive tempering schedule as well as speed-versus-accuracy trade-offs when tuning the adaptive tempering. Subsection 4.2 uses generalized tempering for the online estimation of DSGE models and document its runtime advantages. Finally, in Subsection 4.3 we show that the SMC algorithm is able to reveal multimodal features of DSGE model posteriors.

4.1. Adaptive likelihood tempering

We described in Subsection 2.3 how the tempering schedule for the SMC algorithm can be generated adaptively. The tuning parameter α controls the desired level of reduction in ESS in (2.15). The closer α is to one, the smaller the desired ESS reduction, and therefore the smaller the change in the tempering parameter and the larger the number of tempering steps. We will explore the shape of the adaptive tempering schedule generated by Algorithm 2, the runtime of SMC Algorithm 1, and the accuracy of the resulting Monte Carlo approximation as a function of α. Rather than reporting results for individual DSGE model parameters, we consider the standard deviation of the log marginal data density (MDD) defined in (1), computed across multiple runs of the SMC algorithm, as a measure of accuracy of the Monte Carlo approximation.

We consider the AS and SW models, estimated based on data from 1966:Q4 to 2016:Q3.[6] In addition to the adaptive tempering schedule, we also consider a fixed tempering schedule of the form

$$\phi_n = \left(\frac{n}{N_\phi}\right)^\lambda. \tag{4.1}$$

This schedule is used in the SMC applications in Herbst and Schorfheide (2014) and Herbst and Schorfheide (2015). The user-specified tuning parameters for the SMC algorithm are summarized in Table 1.

Results for the AS model based on adaptive likelihood tempering, see (2.2), are summarized in Table 2. We also report results for the fixed tempering schedule (4) in the second column of the table. For now we set $N_{MH} = 1$. The remaining columns show results for the adaptive schedule with different choices of the tolerated ESS reduction α. The adaptive schedule length is increasing from approximately 112 stages for $\alpha = 0.90$ to 506 stages for $\alpha = 0.98$. As mentioned above, the closer α is to one, the smaller the increase in ϕ_n. This leads to a large number of stages, which, in turn, increases the precision of the Monte Carlo approximation. The standard deviation of log $\hat{p}(Y)$ is 1.54 for $\alpha = 0.9$ and 0.21 for $\alpha = 0.98$ The mean of the log MDD is increasing as

[6] For the estimation of both models, we use a pre-sample from 1965:Q4 to 1966:Q3.

Table 2. AS model: fixed and adaptive tempering schedules, $N_{MH} = 1$.

	Fixed	$\alpha = 0.90$	$\alpha = 0.95$	$\alpha = 0.97$	$\alpha = 0.98$
Mean log (MDD)	− 1032.60	− 1034.21	− 1032.48	− 1032.07	− 1031.92
Std dev log (MDD)	0.76	1.48	0.61	0.32	0.22
Schedule length	200.00	112.17	218.80	350.06	505.46
Resamples	14.63	15.37	15.03	14.99	14.00
Runtime [min]	1.29	0.88	1.53	2.21	3.13

Notes: Results are based on $N_{run} = 400$ runs of the SMC algorithm. We report averages across runs for the runtime, schedule length, and number of resampling steps.

Table 3. SW model: fixed and adaptive tempering schedules, $N_{MH} = 1$.

	Fixed	$\alpha = 0.90$	$\alpha = 0.95$	$\alpha = 0.97$	$\alpha = 0.98$
Mean log (MDD)	− 1178.43	− 1186.23	− 1180.04	− 1178.31	− 1177.72
Std dev log (MDD)	1.34	3.07	1.53	1.03	1.01
Schedule length	500.00	200.53	389.75	618.53	887.42
Resamples	26.75	28.09	27.25	26.33	25.09
Runtime [min]	80.23	31.36	61.29	97.79	139.99

Notes: Results are based on $N_{run} = 200$ runs of the SMC algorithm. We report averages across runs for the runtime, schedule length, and number of resampling steps.

the precision increases. This is the result of Jensen's inequality. MDD approximations obtained from SMC algorithms tend to be unbiased, which means that log MDD approximations exhibit a downward bias.

The runtime of the algorithm increases approximately linearly in the number of stages, since each stage takes approximately the same amount of time. The number of times that the selection step is executed ('resamples' in the table) is approximately constant as a function of α. However, because N_ϕ is increasing, the fraction of stages at which the particles are resampled decreases from 14% to 3%.[7]

In addition to the AS model, we also evaluate the posterior of the SW model using the SMC algorithm with adaptive likelihood tempering. Results are provided in Table 3. Because the dimension of the parameter space of the SW model is much larger than that of the AS model, we are using more particles and multiple blocks in the mutation step when approximating the posterior (see Table 1). For a given α, the log MDD approximation for SW is less accurate than that for AS, which is consistent with the SW model having more parameters that need to be integrated out and a less regular likelihood surface. In general, the results for the SW model are qualitatively similar to those reported for the AS model in Table 2.

In Figure 1, we plot the fixed and adaptive tempering schedules for both models. All of the adaptive schedules are convex. Very little information, less than under the fixed schedule, is added to the likelihood function in the early stages, whereas a large amount of information is added during the later stages. This is consistent with the findings of Herbst and Schorfheide (2014, 2015), who examined the performance of the SMC approximations under the fixed schedule (4.1) for various values of λ. A primary advantage of using an adaptive tempering schedule rather than a fixed one is the savings in 'human-time', corresponding to fewer tuning parameters. While

[7] Dividing the number of resamples by schedule length.

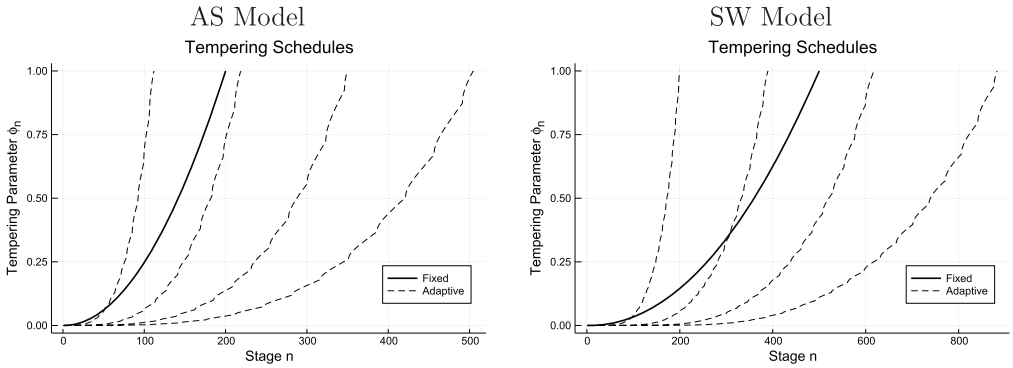

Figure 1. Tempering schedules, $N_{MH} = 1$.

Notes: The figure depicts (pointwise) median ϕ_n values across $N_{run} = 400$ for AS and $N_{run} = 200$ for SW. The solid lines represent the fixed schedule, parameterized according to Table 1. The dashed lines represent a range of adaptive schedules: $\alpha = 0.9, 0.95, 0.97, 0.98$ (the order of the lines from left to right corresponds to increasing values of α).

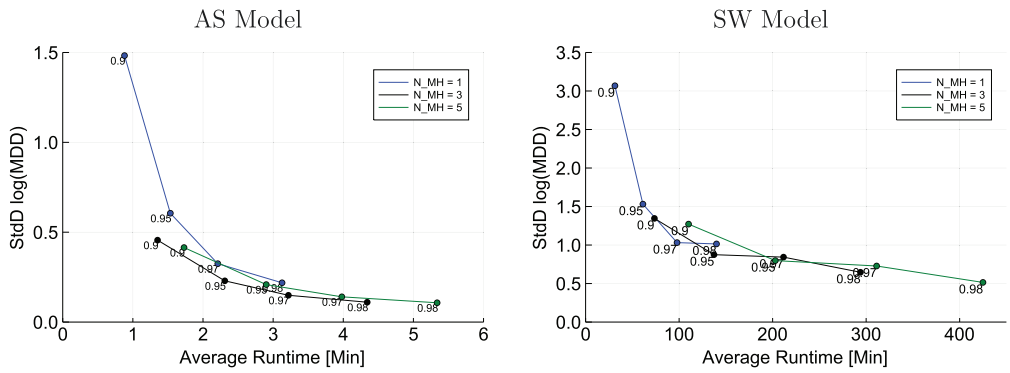

Figure 2. Trade-off between runtime and accuracy, for multiple Metropolis–Hastings steps.

Notes: AS results are based on $N_{run} = 400$, and SW results are based on $N_{run} = 200$ runs of the SMC algorithm. The blue, black, and green lines correspond to N_{MH} values equal to 1, 3, and 5, respectively.

there may be an 'optimal' fixed schedule for any given model and data, finding the best choice of the number of stages N_ϕ and the tempering schedule shape λ would require a great deal of experimentation.

Figure 2 graphically depicts time-accuracy curves for the two models. The curves for $N_{MH} = 1$ are constructed from the standard deviation and runtime numbers reported in Tables 2 and 3. The curves are convex for both models. For $N_{MH} = 1$, reducing α from 0.95 to 0.9 leads to a drastic reduction in the accuracy of the MDD approximation, while the time savings are only modest. The accuracy–runtime pairs for the fixed tempering schedules (not shown in the figure) essentially lie on the curves. For the AS model the accuracy–runtime pair for the fixed tempering

schedules is on the segment between $\alpha = 0.9$ and $\alpha = 0.95$, whereas for the SW model it is on the segement between $\alpha = 0.95$ and $\alpha = 0.97$.

In addition to results for $N_{MH} = 1$, Figure 2 also shows accuracy–runtime curves for $N_{MH} = 3$ and $N_{MH} = 5$. Recall that the mutation step in the SMC algorithm is necessary for particle values to adapt to each intermediate bridge distribution. In principle, one step of a Metropolis–Hastings algorithm is sufficient, because prior to the mutation, the particle swarm already represents the stage n distribution. Nonetheless, there is a potential benefit to raising the number of MH steps: it increases the probability that the particle values do change during the mutation. Unfortunately, it also increases the runtime.

According to Figure 2, the gain from increasing N_{MH} is limited. While the trade-off curves for the two models do shift, for every desirable point on the $N_{MH} = 3$ and $N_{MH} = 5$ curves (runtimes between 1 and 3 minutes), there is a point on the $N_{MH} = 1$ curve that delivers a similar performance in terms of speed and accuracy. Consider the AS model. The combination of $N_{MH} = 1$ and $\alpha = 0.98$ delivers roughly the same performance as $N_{MH} = 3$ and $\alpha = 0.97$. For the former setting we have on average 505 bridge distributions, whereas for the latter setting we have only 301 bridge distributions. Because the mutation step is executed for each bridge distribution, adding bridge distributions facilitates the change in particle values and therefore is to some extent a substitute for increasing the number of MH steps. For both models, increasing N_{MH} from 3 to 5 leads in fact to a (slight) deterioration of performance.[8] In the remainder of the paper, we set $N_{MH} = 1$ and $\alpha = 0.98$ unless otherwise noted.

4.2. Generalized tempering for online estimation

To provide a timely assessment of economic conditions and to produce accurate forecasts and policy projections, econometric modellers at central banks re-estimate their DSGE models regularly. A key impediment to online estimation of DSGE models with the RWMH algorithm is that any amendment to the previously used dataset requires a full re-estimation of the DSGE model, which can be quite time-consuming and often requires supervision.[9] SMC algorithms that are based on data tempering, on the other hand, allow for efficient online estimation of DSGE models. This online estimation entails combining the adaptive tempering schedule in Algorithm 2 with the generalized tempering in (2.14). As mentioned above, our algorithm is also amenable to data revisions.

Scenario 1. In the following illustration, we partition the sample into two subsamples: $t = 1, \ldots, T_1$ and $t = T_1 + 1, \ldots, T$, and allow for data revisions by the statistical agencies between periods $T_1 + 1$ and T. We assume that the second part of the sample becomes available after the model has been estimated on the first part of the sample using the data vintage available at the time, $\tilde{y}_{1:T_1}$. Thus, in period T we already have a swarm of particles $\{\theta^i_{T_1}, W^i_{T_1}\}_{i=1}^N$ that approximates the posterior

$$p(\theta|\tilde{y}_{1:T_1}) \propto p(\tilde{y}_{1:T_1}|\theta)p(\theta).$$

[8] Holding α fixed, one would expected that raising N_{MH} from 1 to 3, say, would approximately triple the runtime because the number of likelihood evaluations increases by a factor of three. Based on Figure 2, that is not the case. Owing to the specifics of the parallelization of the algorithm, each $N_{MH} = 3$ stage takes only 1.8 times as long as each $N_{MH} = 1$ stage.

[9] This is particularly true when the proposal covariance matrix is constructed from the Hessian of the log likelihood function evaluated at the global mode, which can be very difficult to find.

Following (2.14) with $Y = y_{1:T}$ and $\tilde{Y} = \tilde{y}_{1:T_1}$, we define the stage (n) posterior as

$$\pi_n(\theta) = \frac{p(y_{1:T}|\theta)^{\phi_n} p(\tilde{y}_{1:T_1}|\theta)^{1-\phi_n} p(\theta)}{\int p(y_{1:T}|\theta)^{\phi_n} p(\tilde{y}_{1:T_1}|\theta)^{1-\phi_n} p(\theta) d\theta}.$$

We distinguish notationally between y and $\tilde{y}$ because some observations in the $t = 1, \ldots, T_1$ sample may have been revised. The incremental weights are given by

$$\tilde{w}_n^i(\theta) = p(y_{1:T}|\theta)^{\phi_n - \phi_{n-1}} p(\tilde{y}_{1:T_1}|\theta)^{\phi_{n-1} - \phi_n},$$

and it can be verified that

$$\frac{1}{N} \sum_{i=1}^{N} \tilde{w}_n^i W_{n-1}^i \approx \frac{\int p(y_{1:T}|\theta)^{\phi_n} p(\tilde{y}_{1:T_1}|\theta)^{1-\phi_n} p(\theta) d\theta}{\int p(y_{1:T}|\theta)^{\phi_{n-1}} p(\tilde{y}_{1:T_1}|\theta)^{1-\phi_{n-1}} p(\theta) d\theta}. \tag{4.2}$$

Now define the conditional marginal data density (CMDD) as

$$\text{CMDD}_{2|1} = \prod_{n=1}^{N_\phi} \left(\frac{1}{N} \sum_{i=1}^{N} \tilde{w}_n^i W_{n-1}^i \right), \tag{4.3}$$

with the understanding that $W_{(0)}^i = W_{T_1}$. Because the product of the terms in (5) simplifies, and because $\phi_{N_\phi} = 1$ and $\phi_1 = 0$, we obtain

$$\text{CMDD}_{2|1} \approx \frac{\int p(y_{1:T}|\theta) p(\theta) d\theta}{\int p(\tilde{y}_{1:T_1}|\theta) p(\theta) d\theta} = \frac{p(y_{1:T})}{p(\tilde{y}_{1:T_1})}.$$

Note that in the special case of no data revisions ($\tilde{y}_{1:T_1} = y_{1:T_1}$), the expression simplifies to $\text{CMDD}_{2|1} \approx p(y_{T_1+1:T}|y_{1:T_1})$. We consider this case in our simulations below.

We assume that the DSGE model has been estimated using likelihood tempering based on the sample $y_{1:T_1}$, where $t = 1$ corresponds to 1966:Q4 and $t = T_1$ corresponds to 2007:Q1. The second sample, $y_{T_1+1:T}$, starts in 2007:Q2 and ends in 2016:Q3.[10] We now consider two ways of estimating the log MDD, $\log p(y_{1:T})$. Under full-sample estimation, we ignore the existing estimate based on $y_{1:T_1}$ and use likelihood tempering based on the full-sample likelihood $p(y_{1:T}|\theta)$. Under generalized tempering, we start from the existing posterior based on $y_{1:T_1}$ and use generalized data tempering to compute $\text{CMDD}_{2|1}$ in (6). We then calculate $\log p(y_{1:T_1}) + \log \text{CMDD}_{2|1}$.

Note that our choice of the sample split arguably stacks the cards against the generalized tempering approach relative to starting from scratch. This is because the second period is quite different from the first (and therefore the posterior changes quite a bit), as it includes the Great Recession, the effective lower-bound constraint on the nominal interest rate, and unconventional monetary policy interventions such as large-scale asset purchases and forward guidance.

We begin with the following numerical illustration for the AS model. For each of the $N_{run} = 400$ runs of SMC, we first generate the estimate of $\log p(y_{1:T_1}|\theta)$ by likelihood tempering and then continue with generalized tempering to obtain $\log \text{CMDD}_{2|1}$. The two numbers are added to obtain an approximation of $\log p(y_{1:T})$ that can be compared with the results from the full-sample estimation reported in Table 2. The results from the generalized tempering approach are reported in Table 4. Comparing the entries in the two tables, note that the mean and standard deviations (across runs) of the log MDDs are essentially the same. The main difference is that generalized

[10] We use a recent data vintage and abstract from data revisions in this exercise.

Table 4. AS model: generalized tempering, $N_{MH} = 1$.

	$\alpha = 0.90$	$\alpha = 0.95$	$\alpha = 0.97$	$\alpha = 0.98$
Mean log (MDD)	− 1033.95	− 1032.54	− 1032.06	− 1031.93
Std dev log (MDD)	1.37	0.61	0.32	0.24
Schedule length	24.33	47.12	75.74	106.50
Runtime [min]	0.25	0.48	0.69	0.98

Notes: Results are based on $N_{run} = 400$ runs of the SMC algorithm, starting from particles that represent $p(\theta|Y_{1:T_1})$. We report averages across runs for the runtime, schedule length, and number of resampling steps.

Table 5. SW model: generalized tempering, $N_{MH} = 1$.

	$\alpha = 0.90$	$\alpha = 0.95$	$\alpha = 0.97$	$\alpha = 0.98$
Mean log (MDD)	− 1188.93	− 1182.08	− 1180.05	− 1178.90
Std dev log (MDD)	3.10	1.83	1.11	1.06
Schedule length	56.60	115.73	194.01	290.74
Runtime [min]	16.32	33.64	56.79	85.01

Notes: Results are based on $N_{run} = 200$ runs of the SMC algorithm, starting from particles that represent $p(\theta|Y_{1:T_1})$. We report averages across runs for the runtime, schedule length, and number of resampling steps.

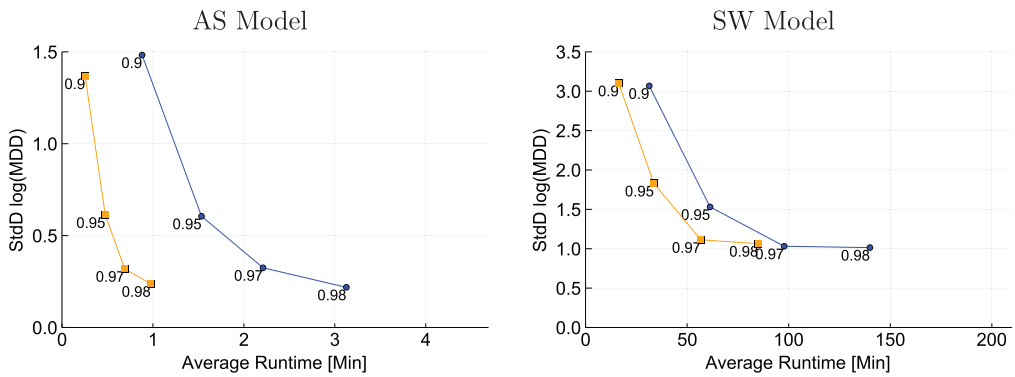

Figure 3. Trade-off between runtime and accuracy, $N_{MH} = 1$.
Notes: AS results are based on $N_{run} = 400$, and SW results are based on $N_{run} = 200$ runs of the SMC algorithm. Yellow squares correspond to generalized tempering, and blue circles correspond to full sample estimation.

tempering reduces the schedule length and the runtime by a factor of roughly 2/3 because it starts from the posterior distribution $p(\theta|y_{1:T_1})$. A comparison of the entries in Tables 3 and 5 shows that the same conclusions hold for the SW model.

In Figure 3, we provide scatter plots of average runtime versus the standard deviation of the log MDD for the AS model and the SW model. The blue circles correspond to full sample estimation and are identical to the blue circles in Figure 2. The yellow squares correspond to generalized tempering. As in Table 4, the runtime does not reflect the time it took to compute $p(\theta|y_{1:T_1})$ because the premise of the analysis is that this posterior has been computed in the past and is available to the user at the time when the estimation sample is extended by the observations

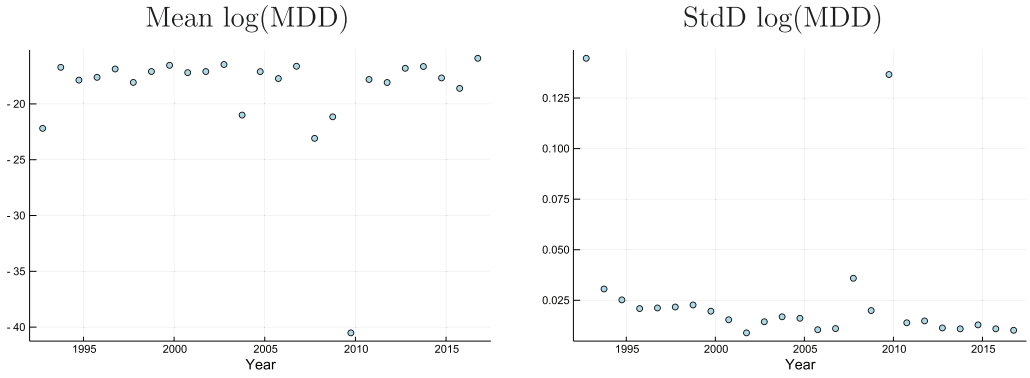

Figure 4. AS model: log MDD increments $\log(p(y_{T_{s-1}+1:T_s}|y_{1:T_{s-1}}))$.
Notes: Results are based on $N_{run} = 200$ runs of the SMC algorithm with $N_{MH} = 1$ and $\alpha = 0.98$.

$y_{T_1+1:T}$. For both models, the accuracy of the log MDD approximation remains roughly the same for a given α when using generalized tempering, but the reduction in runtime is substantial. To put it differently, generalized tempering shifts the SMC time–accuracy curve to the left, which of course is the desired outcome.

Scenario 2. Rather than adding observations in a single block, we now add four observations at a time. This corresponds to a setting in which the DSGE model is re-estimated once a year, which is a reasonable frequency in central bank environments. More formally, we partition the sample into the subsamples $y_{1:T_1}$, $y_{T_1+1:T_2}$, $y_{T_2+1:T_3}$, …, where $T_s - T_{s-1} = 4$. After having approximated the posterior based on observations $y_{1:T_1}$, we use generalized tempering to compute the sequence of densities $p(y_{1:T_s})$ for $s = 2, \ldots, S$. At each step s we initialize the SMC algorithm with the particles that represent the posterior $p(\theta|y_{1:T_{s-1}})$. To assess the accuracy of this computation, we repeat it $N_{run} = 200$ times. We fix the tuning parameter for the adaptive construction of the tempering schedule at $\alpha = 0.98$. The first sample, $y_{1:T_1}$, ranges from 1966:Q4 to 1991:Q3, and the last sample ends in 2016:Q3.

Figure 4 depicts the time series of the mean and the standard deviation of the log MDD increments

$$\log \hat{p}(y_{T_{s-1}+1:T_s}|y_{1:T_{s-1}}) = \log \hat{p}(y_{1:T_s}) - \log \hat{p}(y_{1:T_{s-1}}),$$

across the 200 SMC runs. The median (across time) of the average (across repetitions) log MDD increment is −17.6. The median standard deviation of the log MDD increments is 0.02, and the median of the average runtime is 0.11 minutes or 7 seconds. The largest deviation from these median values occurred during the Great Recession, when we added the 2009:Q4 to 2010:Q3 observations to the sample. During this period, the log MDD increment was only –40.5 and the standard deviation jumped up to 0.14 because these four observations lead to a substantial shift in the posterior distribution. In this period, the runtime of the SMC algorithm increased to 0.58 minutes, or 35 seconds.

Figure 5 depicts the evolution of posterior means and coverage intervals for two parameters, τ and σ_R (the inverse intertemporal elasticity of substitution and the standard deviation of shocks to the interest rate, respectively.) The τ sequence exhibits a clear blip in 2009, which coincides with the increased runtime of the algorithm. Most of the posteriors exhibit drifts rather than

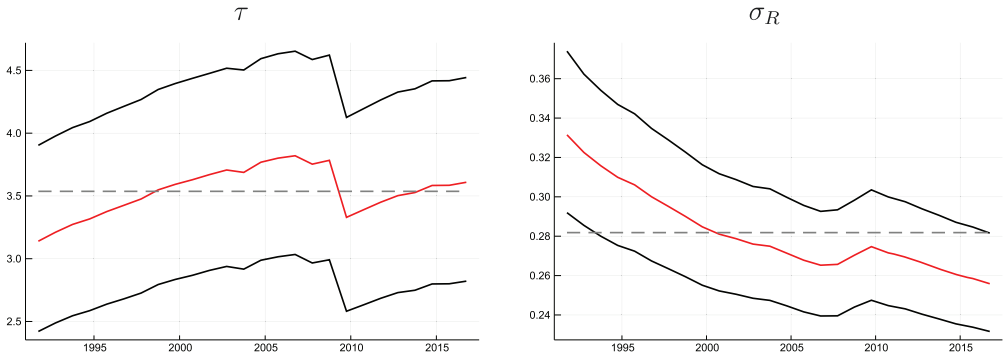

Figure 5. AS model: evolution of posterior means and coverage bands.
Notes: Sequence of posterior means (red line) and 90% coverage bands (black lines). The dashed line indicates the temporal average of the posterior means. We use $N_{MH} = 1$ and $\alpha = 0.98$ for the SMC algorithm.

sharp jumps, and the time-variation in the posterior mean is generally small compared with the overall uncertainty captured by the coverage bands. Overall, generalized tempering provides a framework for online estimation of DSGE models that is substantially more accurate than simple data tempering, at little additional computational cost, even when the additional data substantially change the posterior. Importantly, generalized tempering can also seamlessly handle the data revisions inherent in macroeconomic time series.

4.3. Exploring multimodal posteriors

An important advantage of SMC samplers over standard RWMH samplers is their ability to characterize multimodal posterior distributions. Multimodality may arise because the data are not informative enough to be able to disentangle internal versus external propagation mechanisms, e.g., Calvo price (wage) indexation and persistence of exogenous price (wage) markup shocks. Herbst and Schorfheide (2014) provided an example of a multimodal posterior distribution obtained in a SW model that is estimated under a diffuse prior distribution. Below, we document that a multimodal posterior may also arise if the SW model is estimated on a shorter sample with the informative prior originally used by Smets and Wouters (2007). Capturing this bimodality correctly will be important for the accurate computation of predictive densities that are generated as part of the real-time forecast applications in Section 5.

Figure 6 depicts various marginal bivariate posterior densities for parameters of the SW model estimated based on a sample from 1960:Q1 to 1991:Q3.[11] The plots in the left column of the figure correspond to the 'standard' prior for the SW model. The joint posteriors for the parameters ι_p and ρ_{λ_f} (weight on the backward-looking component in the New Keynesian price Phillips curve, and persistence of price markup shock), on the one hand, and h and ρ_{λ_f} (the former determines the degree of habit formation in consumption), on the other hand, exhibit clear bimodal features,

[11] For these results we match the sample used in Section 5, where we discuss the predictive ability of the various DSGE models. See footnote 14 for a description of why the samples used in Subsections 4.1 and 4.2 differ slightly from the sample in Section 5.

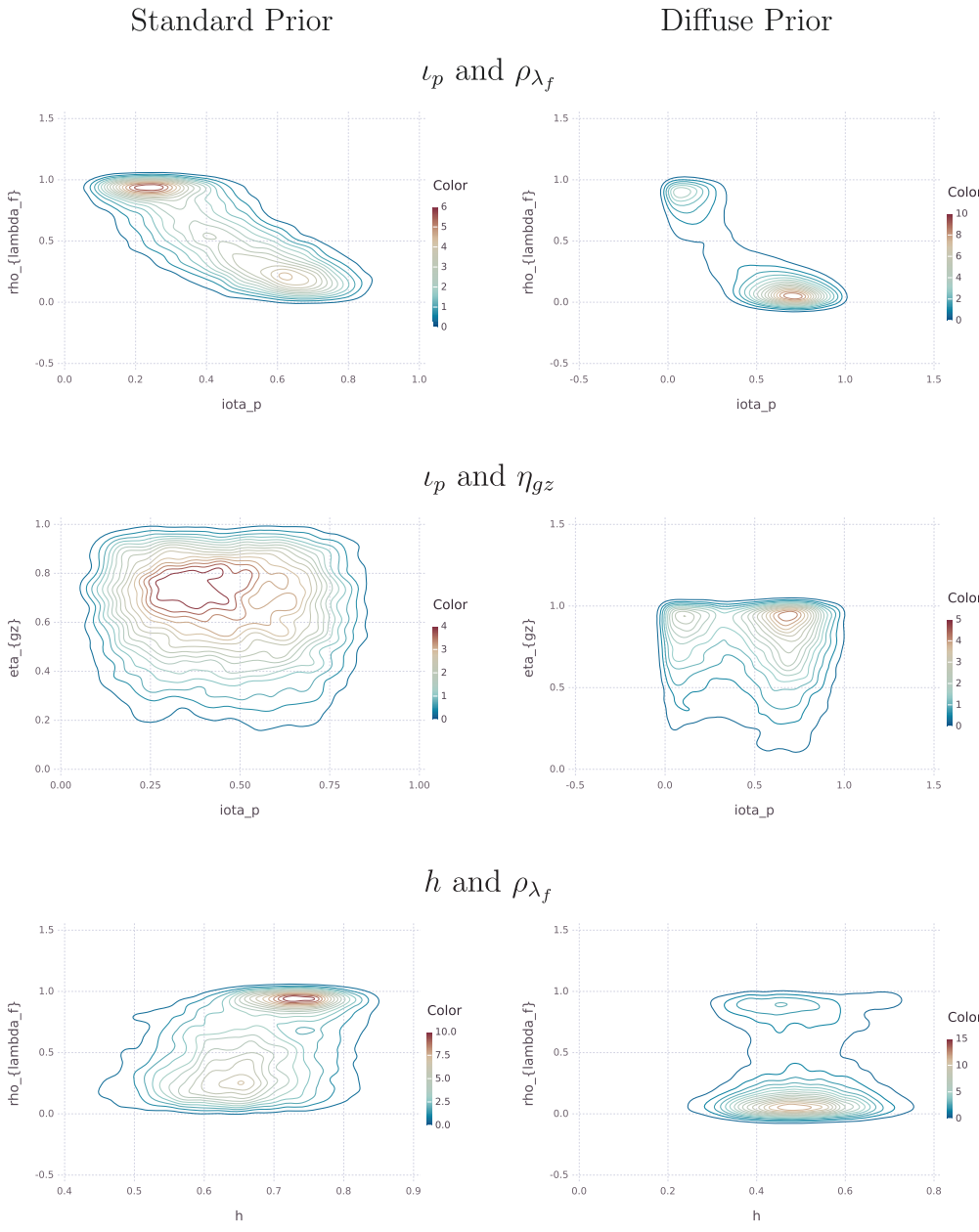

Figure 6. SW model: posterior contours for selected parameter pairs.
Notes: Estimation sample is 1960:Q1 to 1991:Q3. We use $N_{MH} = 1$ and $\alpha = 0.98$ for the SMC algorithm. Plots show a two-dimensional visualization of the full-dimension joint posterior.

albeit one mode dominates the other. For the parameter pair ι_p and η_{gz} (the loading of government spending on technology shock innovations), the bimodality is less pronounced.

The right column of Figure 6 shows posteriors for the same estimation sample but the 'diffuse' prior of Herbst and Schorfheide (2014). This prior is obtained by increasing the standard deviations for parameters with marginal Normal and Gamma distributions by a factor of three and using uniform priors for parameters defined on the unit interval.[12] Under the diffuse prior, the multimodal shapes of the bivariate posteriors are more pronounced, and for the first two parameter pairs both modes are associated with approximately the same probability mass. Thus, using a sampler that correctly captures the non-elliptical features of the posterior is essential for valid Bayesian inference and allows researchers to estimate DSGE models under less informative priors than have traditionally been used in the literature. In Subsection 5.3 we will examine the forecasting performance of the SW model under the diffuse prior.

5. PREDICTIVE DENSITY EVALUATIONS

In this section we compare the forecast performance of two DSGE models, one without (SW) and one with (SWFF) financial frictions. We focus on log predictive density scores, a widely used criterion to compare density forecasts across models (see Del Negro et al., 2016, and Warne et al., 2017, in the context of DSGE model forecasting), rather than root-mean-squared errors, the standard metric for point forecast evaluation used in the literature. We add to the existing literature by studying the predictive ability of these models under a prior that is much more diffuse than the one typically used. For density forecast evaluation, an accurate characterization of the posterior distribution is even more important than for point forecasts. It is clear from the results in Subsection 4.3 that SMC techniques are necessary for this task, given the severe multimodalities present in the posterior, especially under the diffuse prior. We therefore apply the generalized tempering approach to SMC described in Subsection 4.2 when estimating these models recursively using real-time data.

5.1. Real-time dataset and DSGE forecasting setup

This section provides a quick overview of the data series used for the DSGE model estimation, the process of constructing a real-time dataset, which follows the approach of Edge and Gürkaynak (2010) and Del Negro and Schorfheide (2013, subsection 4.1), as well as the forecast setup. Since much of this information is also provided in Cai et al. (2019), we refer to that paper and to Subsection B.4 of the Online Appendix for a more detailed discussion. As mentioned before, the SW model is estimated using data on the growth rate of aggregate output, consumption, investment, the real wage, hours worked, GDP deflator inflation, and the Federal funds rate. The SWFF model is estimated on these same observables plus long-run inflation expectations and the Baa-10-year Treasury spread. All data series start in 1960:Q1 or the first quarter in which the series becomes available.[13]

[12] The prior on the shock standard deviations is the same for the diffuse and standard prior, as this prior is already very loose under the standard specification (an inverse Gamma with only two degrees of freedom). Table B2 of the Online Appendix describes in detail both the standard and the diffuse prior.

[13] A careful reader will notice that this sample start-date is different from the one used in Subsections 4.1 and 4.2 (1964:Q1). In Subsections 4.1 and 4.2, we matched the sample used in Herbst and Schorfheide (2014) so that all series are non-missing. This is required to optimize the Kalman filter algorithm so that it is feasible to run a large number of

The forecast evaluation is based on forecast origins that range from January 1992 (this is the beginning of the sample used in Edge and Gürkaynak, 2010, and Del Negro and Schorfheide, 2013) to January 2017, the last quarter for which eight period-ahead forecasts could be evaluated against realized data. For each forecast origin, we construct a real-time dataset that starts in 1960:Q1, using data vintages available on the 10th of January, April, July, and October of each year, which we obtained from the St. Louis Fed's ALFRED database. Our convention, which follows Del Negro and Schorfheide (2013), is to call the end of the estimation sample T. This is the last quarter for which we have NIPA data, that is, GDP, the GDP deflator, etc. For instance, we use $T = 2010$:Q4 for forecasts generated using data available on April 10, 2011. Even though 2011:Q1 has passed, the Q1 NIPA data are not yet available. We re-estimate the DSGE model parameters once a year with observations ranging from 1960:Q1 to T, using the January vintages. Because financial data are published in real time, we use $T + 1$ financial information to sharpen our inference on the $T + 1$ state of the economy when generating the forecasts. However, the $T + 1$ financial information is excluded from the parameter estimation.

We focus on projections that are conditional on external interest rate forecasts, following Del Negro and Schorfheide (2013, Subsection 5.4). In order to construct the conditional projections, we augment the measurement equations by

$$R^e_{T+k|T} = R_* + \mathbb{E}_T[R_{T+k}], \qquad k = 1, \ldots, K$$

where $R^e_{T+k|T}$ is the observed k-period-ahead interest rate forecast, $\mathbb{E}_T[R_{T+k}]$ is the model-implied interest rate expectation, and R_* is the steady-state interest rate. The interest rate expectations observables $R^e_{T+1|T}, \ldots, R^e_{T+K|T}$ come from the Blue Chip Financial Forecasts (BCFF) forecast released in the first month of quarter $T + 1$. In order to provide the model with the ability to accommodate federal funds rate expectations, the policy rule in the model is augmented with anticipated policy shocks; see Section B in the Online Appendix for additional details.[14]

The projections discussed below are also conditional on nowcasts—that is, forecasts of the current quarter, which we obtain from the Blue Chip Economic Indicators (BCEI) consensus forecasts—of GDP growth and the GDP deflator inflation. We treat the nowcast for $T + 1$ as a perfect signal of y_{T+1}. Finally, the forecasts are evaluated using the latest data vintage, following much of the existing literature on DSGE forecast evaluation. Specifically, for the results shown below we use the vintage downloaded on April 18, 2019. The predictive densities for real GDP growth are computed on per capita data.

5.2. *Log predictive density scores with standard prior*

Figure 7 shows the logarithm of the predictive densities for real GDP growth, GDP deflator inflation, and both variables jointly (left, middle, and right column, respectively) computed over the full and the post-recession samples for the two DSGE models we are considering: SW (blue solid lines) and SWFF (red solid lines). Both models are estimated using what we have referred to as the 'standard' prior, that is, the prior used in previous work, which for most parameters amounts to the prior used in Smets and Wouters (2007). For each forecast origin T, the predictive densities for model m are computed for $h = 2$, 4, 6, and 8 quarters ahead, using the aforementioned

estimations; see Herbst (2015). However, in Subsections 4.3 and 5 we match the sample used in Cai et al. (2019), which starts in 1960:Q1.

[14] As in Del Negro and Schorfheide (2013), but differently from in Cai et al. (2019), we do not use the expanded dataset containing interest rate forecasts in the estimation of the model's parameters beginning in 2008:Q4—the start of the zero-lower-bound period.

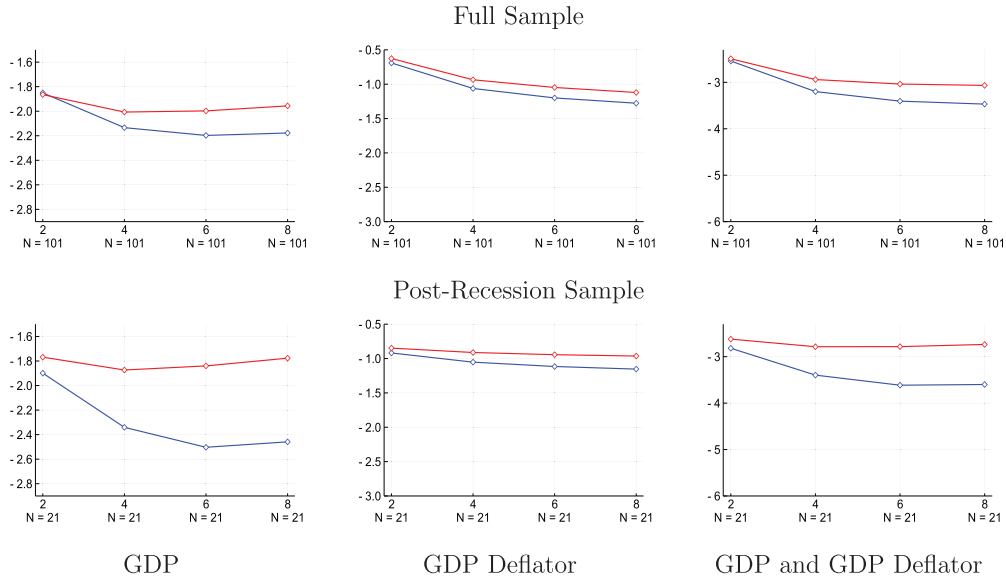

Figure 7. Average log predictive scores for SW versus SWFF.

Notes: These panels compare the log predictive densities from the SW DSGE model (blue diamonds) with those from the SWFF DSGE model (red diamonds) averaged over two, four, six, and eight quarter horizons for output growth and inflation individually, and for both together. The forecasts associated with these predictive densities are generated conditioning on nowcasts and federal funds rate expectations. In the top row, forecast origins from January 1992 to January 2017 only are included in these calculations. In the bottom row, forecast origins from April 2011 to April 2016 only are included in these calculations.

information set $\mathcal{I}_t^m$ that includes interest rate projections, $T+1$ nowcasts, and $T+1$ financial variables.

Before commenting on the results, a few details on the computation of the predictive densities are in order. The objects being forecasted are the h-period averages of the variables of interest j,

$$\bar{y}_{j;T+h,h} = \frac{1}{h}\sum_{s=1}^{h} y_{j;T+s}.$$

While the previous literature on forecasting with DSGE models generally focused on h-period-ahead forecasts of $y_{j;T+h}$, we choose to focus on averages as they arguably better capture the relevant object for policy-makers: accurately forecasting inflation behaviour over the next two years is arguably more important than predicting inflation eight quarters ahead. The time t h-period-ahead posterior predictive density for model m is approximated by

$$p(\bar{y}_{j;T+h,h}|\mathcal{I}_T^m, \mathcal{M}_m) = \frac{1}{N}\sum_{i=1}^{N} p(\bar{y}_{j;T+h,h}|\theta^i, \mathcal{I}_T^m, \mathcal{M}_m),$$

where N is the number of SMC particles and $p(\bar{y}_{j;T+h,h}|\theta^i, \mathcal{I}_T^m, \mathcal{M}_m)$ is the predictive density conditional on the particle θ^i (Subsection A.1 in the Online Appendix provides the

computational details). The values plotted in Figure 7 are the average of the time T log predictive densities across the sample $[T_0, T_1]$, namely

$$\frac{1}{T_1 - T_0} \sum_{T=T_0}^{T_1} \log\ p(\bar{y}_{j;T+h,h} | \mathcal{I}_T^m, \mathcal{M}_m).$$

The top row of Figure 7 shows that over the full sample the SWFF model performs better than the SW model, regardless of the variable being forecasted, and for any forecast horizon higher than two quarters (since we are conditioning on nowcasts, the predictive densities for two quarters ahead are virtually indistinguishable). Quantitatively, the gaps between the two models shown in the first row are nonnegligible for GDP growth at longer horizons, but very small for inflation and for inflation and GDP growth jointly. If we convert the difference in log predictive densities as posterior odds ratios for the two models, a gap of 2 implies that model SWFF is about seven times more likely than model SW. If we focus on the post-recession sample—the sample studied in Cai et al. (2019), which starts in April 2011 and ends in April 2016—we see that the gap between the two models in terms of output growth forecasts becomes larger than 4 for any horizon greater than two (see bottom row), implying posterior odds ratios above 50 to 1. In fact, the time series of the log predictive scores (see Subsection C.2 of the Online Appendix) shows that the superior forecasting performance of SWFF is due to better forecasting accuracy from the Great Recession onward.

In Cai et al. (2019), we explain these results in terms of (i) the failure of the SW model to explain the Great Recession, which affects its forecasting performance thereafter, and (ii) the so-called 'forward guidance puzzle' (Del Negro et al., 2012; Carlstrom et al., 2015). The latter refers to the fact that rational expectations representative agent models tend to overestimate the impact of forward guidance policies, an issue particularly severe for the SW model, leading to projections that were overoptimistic when forward guidance was in place.

5.3. Log predictive density scores with diffuse priors

As illustrated in Subsection 4.3, the prior used in the estimation of DSGE models is often quite informative, in the sense that it affects the posterior distribution. From an econometric point of view, informative priors are not necessarily problematic. They may incorporate a priori information gleaned from other studies; see for instance the discussion in Del Negro and Schorfheide (2008). However, in cases in which the information embedded in the priors is controversial, the use of informative priors has been criticized; see, for instance, the critique in Romer (2016) of the priors used in Smets and Wouters (2007).

In view of this criticism, it is interesting to examine the effect of relaxing a relatively tight prior on the forecasting performance of the model. We therefore compute the log predictive scores also under the diffuse prior described in Subsection 4.3. We previously only considered a diffuse prior for the SW model. For parameters that are common to the SW and the SWFF model, we adopt the same diffuse prior as previously used for the SW model. For parameters that are unique to the SWFF model, we leave the priors unchanged, with the exception that we replace the prior for the autocorrelation of the financial shock by a uniform distribution; see Table B2 in the Online Appendix for details.

From a frequentist perspective, increasing the prior variance reduces the bias of the Bayes estimator while increasing its variability. The net effect on the mean-squared estimation (and hence forecast) error is therefore ambiguous. In models for which not all parameters are identified,

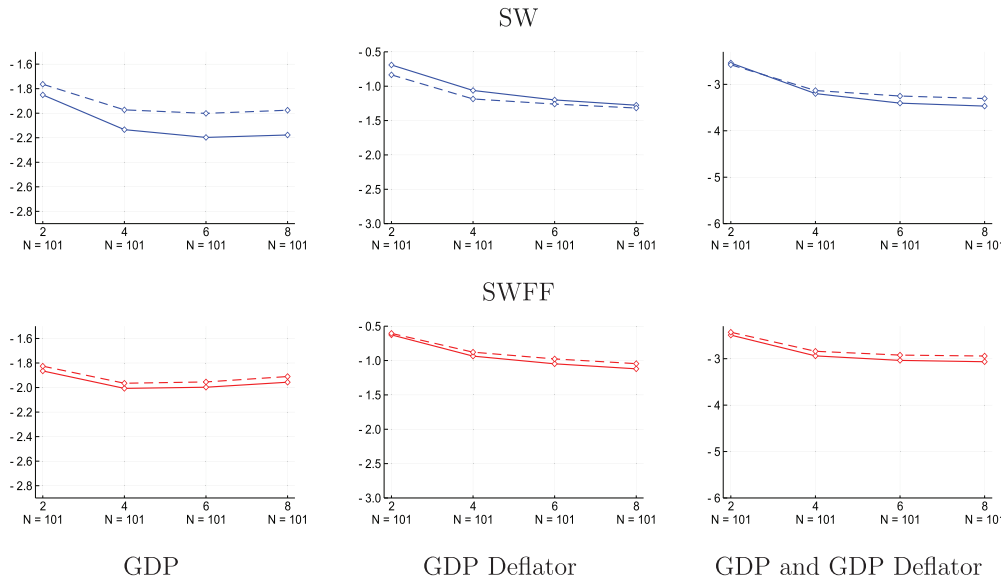

Figure 8. Comparison of predictive densities under standard and diffuse priors.
Notes: These panels compare the predictive densities estimated with the standard (solid line) and the diffuse (dashed line) prior for the SW (blue, top row) and SWFF (red, bottom row) DSGE models. The predictive densities are averaged over two, four, six, and eight quarter horizons for output growth and inflation individually, and for both together. Forecast origins from January 1992 to January 2017 only are included in these calculations. The forecasts associated with these predictive densities are generated conditioning on nowcasts and federal funds rate expectations.

the prior serves as a 'tie-breaker', and introduces curvature into the posterior in directions in which the likelihood function is flat. If a prior mainly adds information where the likelihood is uninformative, and parameters are not identified based on the sample information, then making the prior more diffuse will not have a noticeable effect on the forecasting performance of DSGE models because it mainly selects among parametrizations that track the data equally well.

Figure 8 compares the average log predictive scores obtained with the standard (solid line) and the diffuse (dashed line) prior for the SW (blue, top row) and SWFF (red, bottom row) DSGE models. The panels in Figure 8 show that the differences in average log predictive scores between the standard and the diffuse prior are generally small, with the largest differences arising in the SW model. For this model, the inflation forecast deteriorates as the prior becomes more diffuse, while the GDP forecast improves. This result is consistent with previous findings in the literature. A tight prior on the constant inflation target around 2% in the SW model is key for estimating a target that is consistent with the low inflation in the latter part of the sample. If the prior variance is increased, then the posterior mean shifts closer to the average inflation rate during the estimation period, which is substantially higher than inflation during and after the mid-1990s.

For the SWFF model, replacing the standard prior by the diffuse prior does not lead to a deterioration of inflation forecasts. The reason for this result is unrelated to the financial frictions. It is due to the introduction of the time-varying inflation target, which adapts to the drop in

inflation in the 1980s and therefore generates inflation forecasts in the 1990s and 2000s that are not contaminated by the high inflation rates in the 1960s and 1970s. Overall, the density forecast accuracy of the SWFF is essentially the same under the two prior distributions.

We conclude this section with a word of caution. The result that the prior specification has no influence on the forecasting performance does not imply that the two versions of the model deliver the same policy predictions and historical shock decompositions. To the extent that a tighter prior amplifies and downweights competing modes in the likelihood function—which is apparent from the multimodal posterior densities plotted in Figure 6—posterior inference on the relative importance of the endogenous and exogenous propagation mechanism can be very different. In turn, this can lead to different impulse response functions for structural shocks, and the predicted effect of policy interventions may vary across estimations.

6. CONCLUSION

As the DSGE models used by central banks and researchers become more complex, improved algorithms for Bayesian computations are necessary. This paper provides a framework for performing online estimation of DSGE models using SMC techniques. Rather than starting from scratch each time a DSGE model must be re-estimated, the SMC algorithm makes it possible to mutate and re-weight posterior draws from an earlier estimation so that they approximate a new posterior based on additional observations that have become available since the previous estimation. The algorithm minimizes computational time, requires little user supervision, and can handle the irregular distributions common in the posteriors of DSGE model parameters. The same approach could also be used to transform posterior draws for one model into posterior draws for another model that shares the same parameter space, for example a linear and a nonlinear version of a DSGE model.

REFERENCES

An, S. and F. Schorfheide (2007). Bayesian analysis of DSGE models. *Econometric Reviews 26*, 113–72.

Bernanke, B., M. Gertler and S. Gilchrist (1999). The financial accelerator in a quantitative business cycle framework. In J. Taylor and M. Woodford (Eds.), *Handbook of Macroeconomics, Volume 1*, 1341–93, Amsterdam, North-Holland.

Cai, M., M. D. Negro, M. P. Giannoni, A. Gupta, P. Li and E. Moszkowski (2019). DSGE forecasts of the lost recovery. *International Journal of Forecasting 35*, 1770–89.

Cappé, O., E. Moulines and T. Ryden (2005). *Inference in Hidden Markov Models*. Springer Verlag, New York.

Carlstrom, C. T., T. S. Fuerst and M. Paustian (2015). Inflation and output in New Keynesian models with a transient interest rate peg. *Journal of Monetary Economics 76*, 230–43.

Chopin, N. (2002). A sequential particle filter for static models. *Biometrika 89*, 539–51.

Chopin, N. (2004). Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference. *Annals of Statistics 32*, 2385–411.

Christiano, L. J., M. Eichenbaum and C. L. Evans (2005). Nominal rigidities and the dynamic effects of a shock to monetary policy. *Journal of Political Economy 113*, 1–45.

Christiano, L. J., R. Motto and M. Rostagno (2003). The great depression and the Friedman–Schwartz hypothesis. *Journal of Money, Credit and Banking 35*, 1119–97.

Christiano, L. J., R. Motto and M. Rostagno (2009). Financial factors in economic fluctuations. ECB Working Paper No 1192, May 2010.

Christiano, L. J., R. Motto and M. Rostagno (2014). Risk shocks. *American Economic Review 104*(1), 27–65.

Creal, D. (2007). Sequential Monte Carlo samplers for Bayesian DSGE models. Manuscript, University Chicago Booth, mimeo.

Creal, D. (2012). A survey of sequential Monte Carlo methods for economics and finance. *Econometric Reviews 31*, 245–96.

Del Negro, M. and F. Schorfheide (2008). Forming priors for DSGE models (and how it affects the assessment of nominal rigidities). *Journal of Monetary Economics 55*, 1191–208.

Del Negro, M., and F. Schorfheide (2013). DSGE model-based forecasting. In G. Elliott and A. Timmermann (Eds.), *Handbook of Economic Forecasting, Volume 2*, 57–140. Elsevier.

Del Negro, M., M. P. Giannoni and C. Patterson (2012). The forward guidance puzzle. FRBNY Staff Report, 574.

Del Negro, M., R. B Hasegawa and F. Schorfheide (2016). Dynamic prediction pools: an investigation of financial frictions and forecasting performance. *Journal of Econometrics 192*, 391–405.

Durham, G. and J. Geweke (2014). Adaptive sequential posterior simulators for massively parallel computing environments. *Advances in Econometrics, Volume 34*, 1–44, Emerald Group Publishing Limited.

Edge, R. and R. Gürkaynak (2010). How useful are estimated DSGE model forecasts for central bankers. *Brookings Papers of Economic Activity 41*, 209–59.

Galí, J. (2008). *Monetary Policy, Inflation, and the Business Cycle: An Introduction to the New Keynesian Framework*. Princeton University Press, Princeton, NJ.

Geweke, J. and B. Frischknecht (2014). Exact optimization by means of sequentially adaptive Bayesian learning. Mimeo.

Gordon, N., D. J. Salmond and A. F. E. Smith (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *Radar and Signal Processing, IEE Proceedings F 140*, 107–13.

Graeve, F. D. (2008). The external finance premium and the macroeconomy: US Post-WWII evidence. *Journal of Economic Dynamics and Control 32*, 3415–40.

Greenwood, J., Z. Hercowitz and P. Krusell (1998). Long-run implications of investment-specific technological change. *American Economic Review 87*, 342–36.

Herbst, E. (2015). Using the Chandrasekhar recursions for likelihood evaluation of DSGE models. *Computational Economics 45*, 693–705.

Herbst, E. and F. Schorfheide (2014). Sequential Monte Carlo sampling for DSGE models. *Journal of Applied Econometrics 29*, 1073–98.

Herbst, E. and F. Schorfheide (2015). *Bayesian estimation of DSGE models*, Princeton University Press, Princeton, NJ.

Jasra, A., D. A. Stephens, A. Doucet and T. Tsagaris (2011). Inference for levy-driven stochastic volatility models via adaptive sequential Monte Carlo. *Scandinavian Journal of Statistics 38*, 1–22.

Laseen, S. and L. E. O. Svensson (2011). Anticipated alternative policy-rate paths in policy simulations. *International Journal of Central Banking 7*, 1–35.

Liu, J. S. (2001). *Monte Carlo Strategies in Scientific Computing*. Springer Verlag, New York.

Moral, P. D., A. Doucet and A. Jasra (2012). An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistical Computing 22*, 1009–20.

Romer, P. (2016). The trouble with macroeconomics. *American Economist 20*, 1–20.

Schäfer, C. and N. Chopin (2013). Sequential Monte Carlo on large binary sampling spaces. *Statistical Computing 23*, 163–84.

Smets, F. and R. Wouters (2003). An estimated dynamic stochastic general equilibrium model of the Euro Area. *Journal of the European Economic Association 1*, 1123–75.

Smets, F. and R. Wouters (2007). Shocks and frictions in US business cycles: a Bayesian DSGE approach. *American Economic Review 97*(3), 586–606.

Warne, A., G. Coenen and K. Christoffel (2017). Marginalized predictive likelihood comparisons of linear Gaussian state-space models with applications to DSGE, DSGE-VAR, and VAR models. *Journal of Applied Econometrics 32*, 103–119.

Woodford, M. (2003). *Interest and Prices: Foundations of a Theory of Monetary Policy*. Princeton University Press, Princeton, NJ.

Zhou, Y., A. M. Johansen and J. A. D. Aston (2016). Towards automatic model comparison: an adaptive sequential Monte Carlo approach. *Journal of Computational and Graphical Statistics*, *25*(3), 701–26.

SUPPORTING INFORMATION

Additional Supporting Information may be found in the online version of this article at the publisher's website:

Online Appendix
Replication Package

Managing editor Jaap Abbring and guest editor Jeffrey Campbell handled this manuscript.