Sequential Monte Carlo Sampling for DSGE Models

Edward P. Herbst and Frank Schorfheide

Sequential Monte Carlo Sampling for DSGE Models

Edward Herbst

Federal Reserve Board

Frank Schorfheide[*]

University of Pennsylvania

CEPR, and NBER

June 14, 2013

Abstract

We develop a sequential Monte Carlo (SMC) algorithm for estimating Bayesian dynamic stochastic general equilibrium (DSGE) models, wherein a particle approximation to the posterior is built iteratively through tempering the likelihood. Using three examples–an artificial state-space model, the Smets and Wouters (2007) model, and Schmitt-Grohé and Uribe's (2012) news shock model–we show that the SMC algorithm is better suited for multimodal and irregular posterior distributions than the widely-used random-walk Metropolis-Hastings algorithm. We find that a more diffuse prior for the Smets and Wouters (2007) model improves its marginal data density and that a slight modification of the prior for the news shock model leads to important changes in the posterior inference about the importance of news shocks for fluctuations in hours worked. Unlike standard Markov chain Monte Carlo (MCMC) techniques, the SMC algorithm is well suited for parallel computing.

JEL CLASSIFICATION: C11, C15, E10

KEY WORDS: Bayesian Analysis, DSGE Models, Monte Carlo Methods, Parallel Computing

1 Introduction

Bayesian methods are now widely used to estimate dynamic stochastic general equilibrium (DSGE) models. Bayesian methods combine the likelihood function of a DSGE model with a prior distribution for its parameters to form a posterior distribution that can then be used for inference and decision making. Because it is infeasible to compute moments of the posterior distribution analytically, simulation methods must be used to characterize the posterior. Starting with Schorfheide (2000) and Otrok (2001), the random walk Metropolis-Hastings (RWMH) algorithm – an iterative simulation technique belonging to the class of algorithms known as Markov chain Monte Carlo (MCMC) algorithms – has been the workhorse simulator for DSGE models. Herbst (2011) reports that 95% of papers published from 2005 to 2010 in eight top economics journals use the RWMH algorithm to implement Bayesian estimation of DSGE models.

While the complexity of DSGE models has increased over time, the efficacy of the RWMH algorithm has declined. It is well documented, e.g., Chib and Ramamurthy (2010) and Herbst (2011), that the sequences of DSGE model parameter draws generated by the RWMH algorithm can be very slow to converge to the posterior distribution. This problem is not limited to DSGE model applications; it is important for many areas of applied Bayesian research. Parameter draws may exhibit high serial correlation such that averages of these draws converge very slowly to moments of posterior distributions, or the algorithm may get stuck near local mode and fail to explore the posterior distribution in its entirety (see, for instance, Neal (2003)).

In this paper, we explore an alternative class of algorithms, namely, so-called sequential Monte Carlo (SMC) algorithms, to generate draws from posterior distributions associated with DSGE models. SMC techniques are typically used for solving intractable integration problems (such as filtering nonlinear state space systems); however, they can be used to estimate static model parameters – a point raised by Chopin (2002). The SMC method employed here amounts to recursively constructing importance samplers for a sequence of distributions that begin at an easy-to-sample initial distribution and end at the posterior, supplemented by a series of intermediate "bridge" distributions. The draws from these distributions are called particles and each particle is associated with an importance weight. The particles that represent bridge distribution $n-1$ are "mutated" into particles for bridge distribution n using the Metropolis-Hastings (MH) algorithm.

The contributions of this paper are threefold. First, we tailor a generic SMC algorithm to make it suitable for the analysis of a large-scale DSGE model. More specifically, building on insights from

the application of RWMH algorithms to DSGE models, we use random blocking of parameters as well as mixture proposal distributions during the MH mutation step of the algorithm. Without these modifications, the algorithm failed to explore the posterior surface of large-scale DSGE models. We also made the proposal distribution adaptive, that is, its mean and covariance matrix in iteration n is a function of the particles generated in iteration $n-1$. Second, we present a strong law of large numbers (SLLN) and a central limit theorem (CLT) for the specific version of our algorithm. In particular, we show that, under some regularity conditions, the adaptive determination of the proposal distribution in the mutation step of our algorithm does not affect the limit distribution of the SMC approximation of posterior moments.

Third, we provide two empirical illustrations involving large-scale DSGE models, namely the widely used Smets and Wouters (2007) model (hereafter SW model) and a DSGE model with anticipated (news) shocks developed by Schmitt-Grohé and Uribe (2012) (hereafter SGU model). We find that the SMC algorithm is more stable than the RWMH algorithm if applied repeatedly to generate draws from the same posterior distribution, providing a better approximation of multimodal posterior distributions, in particular. We estimate the SW model under the prior used by Smets and Wouters (2007) as well as a more diffuse prior. While the fit of the model, measured by the marginal data density, substantially improves under the diffuse prior, the posterior surface becomes multimodal and the standard RWMH algorithm does a poor job in capturing this multimodality. We also introduce a small modification to the prior distribution used by SGU to estimate their news shock model and find that conclusions about the importance of news shocks for business cycles can change dramatically. While SGU report that anticipated shocks explain about 70% of the variance of hours worked, under our slightly modified prior the posterior distribution becomes bimodal and most of the posterior probability concentrates near the mode that implies that the contribution of anticipated shocks to hours is only 30%. The RWMH algorithm is unable to characterize this multimodal posterior reliably.

We build on several strands of the existing literature. There exists an extensive body of work in the statistical literature on applying SMC methods to posterior inference for static model parameters as in Chopin (2002). Textbook treatments can be found in Cappé, Moulines, and Ryden (2005) and Liu (2008) and a recent survey is provided by Creal (2012). The theoretical analysis in our paper builds heavily on Chopin (2004), by modifying his proofs to suit our particular version of the SMC algorithm. So far as we know, ours is the second paper that uses SMC to implement

Bayesian inference in DSGE models. Creal (2007) presents a basic algorithm, which he applies to the small-scale DSGE model of Rabanal and Rubio-Ramírez (2005). While we used his algorithm as a starting point, the application to large-scale DSGE models required substantial modifications including the more elaborate adaptive mutation step described above as well as a different sequence of bridge distributions. Moreover, our implementation exploits parallel computation to substantially speed up the algorithm, which is necessary for the practical implementation of this algorithm on modern DSGE models.

In complementary research, Durham and Geweke (2012) use an SMC algorithm to estimate an EGARCH model as well as several other small-scale reduced-form time series models. They employ a graphical processing unit (GPU) to implement an SMC algorithm using the predictive likelihood distribution as the bridge distribution. For our applications, the use of GPUs appeared impractical because the solution and likelihood evaluation of DSGE models involves high-dimensional matrix operations that are not readily available in current GPU programming languages. Durham and Geweke (2012) also propose an adaptive tuning scheme for the sequence of the bridge distributions and the proposal distributions used in the particle mutation step. They suggest determining the tuning parameters in a preliminary run of the SMC algorithm to ensure that the adaptive tuning scheme does not invalidate the CLT for the SMC approximation. Our theoretical analysis suggests that this is unnecessary, provided that the tuning scheme satisfies certain regularity conditions. The authors propose a version of the SMC algorithm that divides particles into groups and carries out independent computations for each group. The variation across groups provides a measure of numerical accuracy. By running the SMC multiple times, we employ an evaluation scheme in our paper that is similar in spirit to theirs.

Other authors have explored alternatives to the version of the RWMH algorithm that has become standard in DSGE model applications. Several papers that have tried to improve upon the RWMH, including Chib and Ramamurthy (2010), Curdia and Reis (2010), Kohn, Giordani, and Strid (2010), and Herbst (2011). These papers propose alternative MCMC algorithms that improve upon the standard single-block RWMH algorithm by grouping parameters into blocks and cycling over conditional posterior distributions (the so-called Metropolis-within-Gibbs algorithm) and by changing the proposal distribution that is used to generate proposed draws in the Metropolis steps. Each of these algorithms, however, is an MCMC technique and remains to some extent susceptible to the above criticism of highly correlated draws.

On the computational front, multiple processor and core environments are becoming more readily available. While likelihood evaluation routines and MCMC algorithms can be written to take advantage of this,[1] neither is "embarrassingly parallelizable," that is, neither naturally exploits the parallel computing framework. This computational challenge might bias researchers to simulate too few draws in their MCMC chains, exacerbating the statistical problem discussed above. SMC algorithms, on the other hand, can easily take advantage of a parallel processing environment. In the extreme case, each draw (or particle) from the initial distribution can be assigned to a separate processor and then converted into a sequence of draws from the "bridge" distributions. While, some communication between the processors is necessary to normalize the particle weights and to potentially eliminate particle degeneracy by a re-sampling step, but the most time-consuming task–namely the evaluation of the likelihood function–can be executed in parallel.

The remainder of this paper is organized as follows. In Section 2, we review some basic insights underlying SMC methods. The SMC algorithm tailored to DSGE model applications is presented in Section 3 and its theoretical properties are studied in Section 4. Section 5 contains three numerical illustrations, one pertaining to a stylized state space model, to motivate estimation problems inherent to DSGE models, and two based on DSGE model posteriors obtained from actual U.S. data. Section 6 concludes. The proofs for Section 4 and a detailed description of the DSGE models estimated in Section 5 as well as additional empirical results are relegated to the Online Appendix.

2   Sequential Monte Carlo Methods

Let $p(Y|\theta)$ denote the likelihood function and $p(\theta)$ the prior density. The object of interest is the posterior density $p(\theta|Y)$ given by

$$p(\theta|Y) = \frac{p(Y|\theta)p(\theta)}{p(Y)}, \quad \text{where } p(Y) = \int p(Y|\theta)p(\theta)d\theta. \tag{1}$$

The parameter vector θ has support Θ. To economize on notation, we abbreviate this density by $\pi(\theta) = p(\theta|Y)$. Moreover, we denote the numerator in (1) by $f(\theta) = p(Y|\theta)p(\theta)$ and the denominator by Z, which does not depend on θ. Using this more compact notation

$$\pi(\theta) = \frac{f(\theta)}{Z}. \tag{2}$$

---

[1]For instance, one can run separate Markov chains on each processor and subsequently merge the draws.

While for linearized DSGE models with Gaussian innovations and a regular prior density the function $f(\theta)$ can be evaluated to machine accuracy without the use of simulation approximations, the normalization constant Z is unknown and closed-form expressions for posterior moments under $\pi(\theta)$ are unavailable. Thus, posterior expectations of θ and transformations $h(\theta)$ have to be approximated numerically with Monte Carlo (MC) simulation methods. Most of the Bayesian DSGE literature applies Markov chain Monte Carlo (MCMC) techniques. As we will argue later, the increasing complexity of DSGE models combined with the emerging parallel framework for scientific computing makes MCMC less attractive for sampling. Instead, sequential Monte Carlo (SMC) methods, we will argue, are an appealing alternative simulation technique. We describe the basics of SMC below. More elaborate explications can be found in Chopin (2002), Del Moral, Doucet, and Jasra (2006), and Creal (2012).

One of the important steps in all SMC algorithms involves importance sampling: we might try to approximate $\pi(\cdot)$ using a different, tractable density $g(\theta)$ that is easy to sample from. Importance sampling is based on the identity

$$E_\pi[h(\theta)] = \int h(\theta)\pi(\theta)d\theta = \frac{1}{Z}\int_\Theta h(\theta)w(\theta)g(\theta)d\theta, \quad \text{where} \quad w(\theta) = \frac{f(\theta)}{g(\theta)}, \tag{3}$$

Suppose that $\theta^i \overset{iid}{\sim} g(\theta)$, $i = 1, \ldots, N$. Then, under suitable regularity conditions, see Geweke (1989), the Monte Carlo estimate

$$\bar{h} = \sum_{i=1}^N h(\theta^i)\tilde{W}^i, \quad \text{where} \quad \tilde{W}^i = \frac{w(\theta^i)}{\sum_{j=1}^N w(\theta^j)}, \tag{4}$$

converges almost surely (a.s.) to $E_\pi[h(\theta)]$ as $N \longrightarrow \infty$. The set of pairs $\{(\theta^i, \tilde{W}^i)\}_{i=1}^N$ provides a particle approximation of $\pi(\theta)$. The $\tilde{W}^i$'s are the (normalized) importance weights assigned to each particle value θ^i. The accuracy of the approximation is driven by the "closeness" of $g(\cdot)$ to $f(\cdot)$ and is reflected in the distribution of the weights. If the distribution of weights is very uneven, the Monte Carlo approximation $\bar{h}$ is inaccurate. Uniform weights arise if $g(\cdot) \propto f(\cdot)$, which means that we are sampling directly from $\pi(\theta)$.

Importance sampling was first used for posterior inference in DSGE models by DeJong, Ingram, and Whiteman (2000). However, in practice, it is extremely difficult to find densities $g(\theta)$ that lead to efficient importance samplers. This task is particularly challenging if the posterior has a non-normal shape, containing several peaks and valleys. The essence of the SMC methods employed in

this paper is to construct sequential particle approximations to intermediate distributions, indexed by n:[2]

$$\pi_n(\theta) = \frac{f_n(\theta)}{Z_n} = \frac{[p(Y|\theta)]^{\phi_n} p(\theta)}{\int [p(Y|\theta)]^{\phi_n} p(\theta) d\theta}, \quad n = 1, \ldots, \phi_{N_\phi} \tag{5}$$

where $\phi_1 = 0$ and $\phi_{N_\phi} = 1$. Note that $\pi_n(\theta) = p(\theta)$ for $n = 1$. Since priors in DSGE models are typically specified such that draws can either be generated by direct sampling or with an efficient acceptance sampler, the initialization of the SMC algorithm is straightforward. Thus, provided it is possible to use the approximation of $\pi_n(\theta)$ to assist in the construction of a particle approximation for $\pi_{n+1}(\theta)$, one can use iterative approximations to estimate $\pi_{N_\phi}(\theta) = \pi(\theta)$. A function (here the likelihood) raised to a power less than one is called a *tempered* function. The process of estimating the parameters of a function through a sequence of tempered functions is known as *simulated tempering*. This estimation framework has a long history in statistics; see Liu (2008) and the references therein. It is common to refer to the sequence of tempering parameters as the *tempering schedule* (or heating schedule, due to its connection to simulated annealing.)

3   An SMC Algorithm for DSGE Models

We begin with the description of the basic algorithm in Section 3.1. This algorithm consists of three steps, using Chopin (2004)'s terminology: *correction*, that is, reweighting the particles to reflect the density in iteration n; *selection*, that is, eliminating any particle degeneracy by resampling the particles; and *mutation*, that is, propagating the particles forward using a Markov transition kernel to adapt to the current bridge density. Section 3.2 provides details on the choice of the transition kernel in the mutation step, and the adaptive choice of various tuning parameters is discussed in Section 3.3. Finally, we provide a summary of the key aspects of our algorithm in Section 3.4.

3.1   The Basic Algorithm

To avoid confusion as to whether θ is drawn from $\pi_n(\cdot)$ or $\pi_{n+1}(\cdot)$, we equip the parameter vector with a subscript n. Thus, θ_n is associated with the density $\pi_n(\cdot)$.

Algorithm 1 (Simulated Tempering SMC)

---

[2]Using the notation that $Y_{t_1:t_2} = \{y_{t_1}, \ldots, y_{t_2}\}$ and $Y = Y_{1:T}$ one could define an integer-valued sequence ϕ_n with $\phi_1 = 0$ and $\phi_{N_\phi} = T$ and define $f_n(\theta) = p(Y_{1:\phi_n}|\theta)$. This data-point tempering approach is attractive for applications in which θ is sequentially estimated on an increasing sample.

1. **Initialization.** *($\phi_1 = 0$). Draw the initial particles from the prior:*

$$\theta_1^i \overset{iid}{\sim} p(\theta), \quad W_1^i = 1, \quad i = 1, \ldots, N.$$

2. **Recursion.** *For $n = 2, \ldots, N_\phi$,*

   (a) **Correction.** *Reweight the particles from stage $n - 1$ by defining the incremental and normalized weights*

   $$\tilde{w}_n^i = [p(Y|\theta_{n-1}^i)]^{\phi_n - \phi_{n-1}}, \quad \tilde{W}_n^i = \frac{\tilde{w}_n^i W_{n-1}^i}{\frac{1}{N} \sum_{i=1}^N \tilde{w}_n^i W_{n-1}^i}, \quad i = 1, \ldots, N.$$

   An approximation of $\mathbb{E}_{\pi_n}[h(\theta)]$ is given by

   $$\tilde{h}_{n,N} = \frac{1}{N} \sum_{i=1}^N h(\theta_{n-1}^i) \tilde{W}_n^i. \tag{6}$$

   (b) **Selection.** *Compute the effective sample size $ESS = N / \left(\frac{1}{N} \sum_{i=1}^N (\tilde{W}_i^n)^2 \right)$.*

   Case (i): If $ESS < N/2$, resample the particles via multinomial resampling. Let $\{\hat{\theta}\}_{i=1}^N$ denote N iid draws from a multinomial distribution characterized by support points and weights $\{\theta_{n-1}^i, \tilde{W}_n^i\}_{i=1}^N$ and set $W_n^i = 1$. Case (ii): If $ESS \geq N/2$, let $\hat{\theta}_n^i = \theta_{n-1}^i$ and $W_n^i = \tilde{W}_n^i$, $i = 1, \ldots, N$. An approximation of $\mathbb{E}_{\pi_n}[h(\theta)]$ is given by

   $$\hat{h}_{n,N} = \frac{1}{N} \sum_{i=1}^N h(\hat{\theta}_n^i) W_n^i. \tag{7}$$

   (c) **Mutation.** *Propagate the particles $\{\hat{\theta}_i, W_n^i\}$ via M steps of a MH algorithm with transition density $\theta_n^i \sim K_n(\theta_n | \hat{\theta}_n^i; \zeta)$ and stationary distribution $\pi_n(\theta)$ (see Algorithm 2 for details below). An approximation of $\mathbb{E}_{\pi_n}[h(\theta)]$ is given by*

   $$\bar{h}_{n,N} = \frac{1}{N} \sum_{i=1}^N h(\theta_n^i) W_n^i. \tag{8}$$

3. *For $n = N_\phi$ ($\phi_{N_\phi} = 1$) the final importance sampling approximation of $\mathbb{E}_\pi[h(\theta)]$ is given by:*

$$\bar{h}_{N_\phi, N} = \sum_{i=1}^N h(\theta_{N_\phi}^i) W_{N_\phi}^i. \tag{9}$$

Our basic SMC algorithm requires tuning in two dimensions. First, the user has to specify the number of particles N. Under suitable regularity conditions discussed in Section 4, the SMC approximations of posterior moments satisfy a CLT, which implies that the precision (i.e. inverse variance) of the approximation increases in proportion to N. Second, the user has to determine the tempering schedule $\{\phi_n\}_{n=1}^{N_\phi}$. All other things equal, increasing the number of stages of the tempering schedule, N_ϕ, will decrease the distance between bridge distributions and thus make it easier to maintain particle weights that are close to being uniform. The cost of increasing N_ϕ is that each stage requires additional likelihood evaluations. To control the shape of the tempering schedule, we introduce a parameter λ:

$$\phi_n = \left(\frac{n-1}{N_\phi - 1} \right)^\lambda.$$

A large value of λ implies that the bridge distributions will be very similar (and close to the prior) for small values of n and very different at a later stage when n is large. In the DSGE model applications, we found a value of $\lambda = 2$ to be very useful, because for smaller values the information from the likelihood function will dominate the priors too quickly and only a few particles will survive the correction and selection steps. Conversely, if λ is much larger than 2, it makes some of the bridge distributions essentially redundant and leads to unnecessary computations. The choice of λ does not affect the overall number of likelihood evaluations.

Algorithm 1 is initialized by generating N *iid* draws from the prior distribution. This initialization will work well as long as the prior is sufficiently diffuse to assign non-trivial probability mass to the area of the parameter space in which the likelihood function peaks.[3] If the particles are initialized based on a more general distribution with density $g(\theta)$, then for $n = 2$ the incremental weights have to be corrected by the ratio $p(\theta)/g(\theta)$. In the selection step, the resampling is only executed if the effective sample size ESS, which is a function of the variance of the particle weights, falls below some threshold. We discuss the rationale for this threshold rule in more detail in Section 4.

---

[3]There exist papers in the DSGE model estimation literature in which the *posterior mean* of some parameters is several *prior* standard deviations away from the *prior mean*. For such applications it might be necessary to choose $\phi_1 > 0$ and to use an initial distribution that is also informed by the tempered likelihood function $[p(Y|\theta)]^{\phi_1}$.

3.2 The Transition Kernel for the Mutation Step

The transition kernel $K_n(\theta_n|\hat{\theta}_n; \zeta)$ is generated through a sequence of M Metropolis-Hastings steps. It is indexed by a vector of tuning parameters ζ. The transition kernel is constructed such that for each ζ the posterior $\pi_n(\theta)$ is an invariant distribution. The MH steps are summarized in the following algorithm. Let θ^* denote a particular set of values for the parameter vector θ and Σ^* be a covariance matrix that is conformable with the parameter vector θ. In Section 3.3 we will replace θ^* and Σ^* by estimates of $\mathbb{E}_{\pi_n}[\theta]$ and $\mathbb{V}_{\pi_n}[\theta]$.

Algorithm 2 (Particle Mutation) *In Step 2(c) at iteration n of Algorithm 1:*

1. *Randomly partition[4] the parameter vector θ_n into N_{blocks} equally sized blocks, denoted by $\theta_{n,b}$, $b = 1, \ldots, N_{blocks}$. Moreover, let θ_b^* and Σ_b^* be the partitions of θ^* and Σ^* that correspond to the subvector $\theta_{n,b}$.*

2. *For each particle i, run M steps of the following MH algorithm. For $i = 1$ to M:*

   For $b = 1$ to N_{blocks}:

   (a) *Let $\theta_{n,b,m}^i$ be the parameter value for $\theta_{n,b}^i$ in the m-th iteration (initialization for $m = 1$: $\theta_{n,b,0}^i = \theta_{n-1,b}^i$) and let $\theta_{n,-b,m}^i = \left[\theta_{n,1,m}^i, \ldots, \theta_{n,b-1,m}^i, \theta_{n,b+1,m-1}^i, \ldots, \theta_{n,N_{blocks},m-1}^i\right]$.*

   (b) *Generate a proposal draw ϑ_b from the mixture distribution*

   $$
   \begin{aligned}
   \vartheta_b|(\theta_{n,b,m-1}^i, &\theta_{n,-b,m}^i, \theta_b^*, \Sigma_b^*) \qquad\qquad (10)\\
   \sim\ &\alpha N\left(\theta_{n,b,m-1}^i, c^2\Sigma_b^*\right) + \frac{1-\alpha}{2} N\left(\theta_{n,b,m-1}^i, c^2 diag(\Sigma_b^*)\right)\\
   &+ \frac{1-\alpha}{2} N\left(\theta_b^*, c^2\Sigma_b^*\right)
   \end{aligned}
   $$

   and denote the density of the proposal distribution by $q(\vartheta_b|\theta_{n,b,m-1}^i, \theta_{n,-b,m}^i, \theta_b^*, \Sigma_b^*)$.

   (c) *Define the acceptance probability*

   $$
   \begin{aligned}
   &\alpha(\vartheta_b|\theta_{n,b,m-1}^i, \theta_{n,-b,m}^i, \theta_b^*, \Sigma_b^*)\\
   &= \min\left\{1, \frac{p^{\phi_n}(Y|\vartheta_b, \theta_{n,-b,m}^i)p(\vartheta_b, \theta_{n,-b,m}^i)\big/q(\vartheta_b|\theta_{n,b,m-1}^i, \theta_{n,-b,m}^i, \theta_b^*, \Sigma_b^*)}{p^{\phi_n}(Y|\theta_{n,b,m-1}^i, \theta_{n,-b,m}^i)p(\theta_{n,b,m-1}^i, \theta_{n,-b,m}^i)\big/q(\theta_{n,b,m-1}^i|\vartheta_b, \theta_{n,-b,m}^i, \theta_b^*, \Sigma_b^*)}\right\}
   \end{aligned}
   $$

---

[4]We assign *iid* $U[0,1]$ draws to each parameter, sort the parameters according to the assigned random numbers, and then let the i-th block consist of parameters $(i-1)N_{blocks}, \ldots, iN_{blocks}$.

and let

$$\theta_{n,b,m}^{i} = \begin{cases} \vartheta_b & \text{with probability } \alpha(\vartheta_b | \theta_{n,b,m-1}^{i}, \theta_{n,-b}^{i}, \theta_b^*, \Sigma_b^*) \\ \theta_{n,b,m-1}^{i} & \text{otherwise} \end{cases}$$

3. Let $\theta_{n,b}^{i} = \theta_{n,b,M}^{i}$ for $b = 1, \ldots, N_{blocks}$.

The particle-mutation algorithm uses several insights from the application of RWMH algorithms to DSGE models. First, Chib and Ramamurthy (2010) and Herbst (2011) have documented that in DSGE model applications, blocking of parameters can dramatically reduce the persistence in Markov chains generated by the MH algorithm. N_{blocks} determines the number of partitions of the parameter vector. For simple models with elliptical posteriors, blocking the parameter vector is unnecessary. When the posterior becomes complex or the dimensionality of the parameter vector is large, however, moving all the parameters in a single block precludes all but the smallest moves, which could hamper the mutation step. Second, Kohn, Giordani, and Strid (2010) have proposed an adaptive MH algorithm for DSGE model posteriors in which the proposal density is a mixture of a random-walk proposal, an independence proposal, and a t-copula estimated from previous draws of the chain. Adopting this general idea, our proposal density in Step 2(a) of Algorithm 2 is also a mixture. The first component corresponds to a random-walk proposal with non-diagonal covariance matrix, the second component is a random-walk proposal with diagonal covariance matrix, and the third component is an independence proposal density.

The tuning parameter $\alpha \in [0, 1]$ controls the weight of the mixture components. The parameter c scales the covariance matrices of the proposal densities.[5] The number of MH steps M affects the probability that the particle mutation step is successful in the sense that $\theta_n^i \neq \hat{\theta}_n^i$. In practice, the effect of increasing M turned out to be similar to the effect of raising N_ϕ and thereby reducing the distance between bridge distributions. In the applications in Section 5, we set $M = 1$.

3.3   Adaption of the Transition Kernel

To achieve a good performance of the SMC algorithm, it is important to choose some of the tuning parameters adaptively, that is, tuning parameters for iteration n are chosen based on the particles

---

[5]In principle one could use separate scaling factors for the three mixture components but in our applications a single scaling factor was sufficient.

generated in iteration $n - 1$. We collect the adaptively chosen parameters in the vector ζ:

$$\zeta = \left[c, \theta^{*\prime}, vech(\Sigma^*)^\prime \right]^\prime. \tag{11}$$

In the implementation of the mutation algorithm we fix the remaining tuning parameters, which are M, N_{blocks}, and α. We also fix the tuning parameters N, N_ϕ, and λ for Algorithm 1 ex ante. We use importance sampling approximations of $\mathbb{E}_{\pi_n}[\theta]$ and $\mathbb{V}_{\pi_n}[\theta]$ to specify θ^* and Σ^* and we adjust the scaling factor c to ensure that the acceptance rate in the MH step is approximately 25%, along the lines of Durham and Geweke (2012). At each iteration n, we replace ζ in (11) with

$$\hat{\zeta}_n = \left[\hat{c}_n, \tilde{\theta}_n, vech(\tilde{\Sigma}_n)^\prime \right]^\prime, \tag{12}$$

where $\hat{\zeta}_n$ is measurable with respect to the σ-algebra $\mathcal{F}_{n-1,N}$ generated by the particles $\{\theta^i_{n-1}, \tilde{W}^i_n\}^N_{i=1}$.[6] The following algorithm describes how $\hat{\zeta}_n$ is constructed at each iteration n.

Algorithm 3 (Adaptive Particle Mutation) *Prior to Step 1 of Algorithm 2:*

1. *Compute importance sampling approximations $\tilde{\theta}_n$ and $\tilde{\Sigma}_n$ of $\mathbb{E}_{\pi_n}[\theta]$ and $\mathbb{V}_{\pi_n}[\theta]$ based on the particles $\{\theta^i_{n-1}, \tilde{W}^i_n\}^N_{i=1}$.*

2. *Compute the average empirical rejection rates $\hat{R}_{n-1}(\hat{\zeta}_{n-1})$, based on the Mutation step in iteration $n - 1$. The averages are computed across the N_{blocks} blocks and across the three mixture components separately.*

3. *Adjust the scaling factor according to*

$$\hat{c}_n = \hat{c}_{n-1} f\left(\hat{R}_{n-1}(\hat{\zeta}_{n-1}) \right),$$

   where $f(\cdot)$ is given by
$$f(x) = 0.95 + 0.10 \frac{e^{16(x-0.25)}}{1 + e^{16(x-0.25)}}.$$

4. *Execute Algorithm 2 by replacing ζ with $\hat{\zeta}_n = \left[\hat{c}_n, \tilde{\theta}_n, vech(\tilde{\Sigma}_n)^\prime \right]^\prime$.*

---

[6] We use "tilde" instead of "hat" for θ and Σ because the approximations are based on the correction step in Algorithm 1.

Note that $f(0.25) = 1$, which means that the scaling factor stays constant whenever the target acceptance rate is achieved. If the acceptance rate is below (above) 25% the scaling factor is decreased (increased). To satisfy the regularity conditions for the theoretical analysis in Section 4, we chose a function $f(x)$ that is differentiable.

3.4   Further Discussion

Our implementation of the SMC algorithm differs from that in Creal (2007) in two important dimensions. First, our mutation step is more elaborate. The mixture proposal distribution proved to be important for adapting the algorithm to large-scale DSGE models with complicated multimodal posterior surfaces. The random blocking scheme is important for avoiding bad blocks as the correlation structure of the tempered posterior changes. Moreover, the introduction of the tempering parameter, λ, is crucial. Creal (2007) uses a linear cooling schedule (i.e., $\lambda = 1$). Even for a large number of stages (n_ϕ) at $n = 1$, the information contained in $[p(Y|\theta)]^{\phi_1}$ dominates the information contained in the prior. This means that initializing the algorithm from the prior is impractical, as sample impoverishment occurs immediately. In light of this, Creal (2007) initializes the simulator from a student-t distribution centered at the posterior mode. The initialization presumes prior knowledge of the mode(s) of the posterior and is not well suited in applications where there are many modes or the posterior is severely non-elliptical. Instead, by using a $\lambda > 1$, we are able to add information from the likelihood to the prior slowly. This allows us to initialize the algorithm from the prior, working without presupposition about the shape of the posterior. Collectively, these additional tuning parameters yield a degree of flexibility that is crucial for the implementation of SMC algorithms on large-scale DSGE models.

Second, our implementation exploits parallel computing to substantially speed up the algorithm. The potential for parallelisation is a key advantage of SMC over MCMC routines. In Algorithm 2, step 2 involves $N \times M \times N_{blocks}$ MH steps. Here, the principal computational bottleneck is the evaluation of the likelihood itself.[7] The total number of likelihood evaluations in the SMC algorithm is equal to $N \times N_\phi \times N_{blocks} \times M$. For some of the examples considered in this paper, that number can be in the tens of millions. In our experience, the standard number of draws in an MCMC estimation of a DSGE model is rarely more than one million. So it would appear that the SMC algorithm would take a prohibitively long time to run. With SMC methods, however, one can exploit the fact

---

[7]We speed up the evaluation of the likelihood of the DSGE model by using the Chandrasekhar recursions to compute the predictive decomposition of the likelihood. See Herbst (2012) for details.

that the $M \times N_{blocks}$ likelihood evaluations in Step 2 of Algorithm 2 can be executed independently for each of the N particles. Given access to a multiple processor[8] environment, this feature is very useful. Because multiple core/processor setups are quite common, parallelization can be achieved easily in many programming languages, e.g., via the MATLAB command `parfor`, even on a single machine.

In our implementation, we use multiple machines (distributed memory) communicating via a Message Passing Interface (MPI). To get a sense of gains from parallelization, let $T(P)$ be the running time of an algorithm using P processors. The relative speedup is given by $S(P) = T(1)/T(P)$. For example, for the standard SW model in Section 5.2, $S(24) \approx 15$, which means that using 24 processors speeds up the algorithm by a factor of 15. A back-of-the-envelope calculation suggests that, for the specific model and computational setup used here, about 97% of the SMC algorithm's time is spent in Step 2 of Algorithm 2. This means that parallelization is extremely efficient.[9]

4 Theoretical Properties

We proceed with a formal analysis of Algorithm 1 and provide some regularity conditions under which the SMC approximations satisfy a SLLN and CLT. The main results are summarized in four theorems. The first three theorems establish a SLLN/CLT for the correction, selection, and mutation steps of Algorithm 1. The fourth theorem shows that the adaptive choice of the transition kernel discussed in Section 3.3 does not, under some regularity conditions, affect the asymptotic variance of the SMC approximation. The first three theorems (and their proofs) essentially correspond to Lemmas A.1, A.2, and A.3 in Chopin (2004). We make three modifications: (i) while our proofs closely follow the proofs in Chopin (2004), we use our specific assumptions about the prior and likelihood function; (ii) our algorithm executes the three steps in a different order; and (iii) we resample only if ESS falls below the threshold $N/2$, which requires an adjustment in some of the asymptotic variance formulas. Although the first three theorems are not new, we think that it is worthwhile to reproduce them, because the asymptotic variance formulas provide valuable insights for practitioners into the behavior of the algorithm. To the best of our knowledge, the fourth

---

[8]We use the term processor to refer to the basic physical unit capable of executing a single thread.

[9]We perform this calculation as follows. Divide the algorithm instructions in two parts: a fraction $1 - F$ of serial (i.e., non-parallelizable) instructions and F instructions which can be executed in parallel. Amdahl's Law, see Amdahl (1967), states that the best expected improvement from using P processors/cores/threads is $T(P) = T(1)(1 - F + F/P)$. Using the runtimes for our algorithm for various Ps, we can come up with an (admittedly crude) estimate for F.

theorem is new. It states that the adaptive choice of ζ has no effect on the limit distribution of the SMC approximation. This asymptotic variance, of course, depends on ζ, which we define to be the probability limit of $\hat{\zeta}_n$ in (12). Detailed proofs of the four theorems are provided in the Online Appendix. We begin with some assumptions on the prior and the likelihood function:[10]

Assumption 1 *Suppose that (i) the prior is proper:* $\int p(\theta)d\theta < \infty$; *(ii) the likelihood function is uniformly bounded:* $\sup_{\theta \in \Theta} p(Y|\theta) < M < \infty$; *and (iii)* $\int [p(Y|\theta)]^{\phi_2}p(\theta)d\theta > 0$.

For the analysis of a DSGE model, the restriction to proper prior distributions does not pose a real constraint on the researcher. In fact, most priors used in the literature are fairly informative and many parameters have a bounded domain. We also assume that the likelihood function is bounded from above and that there exists a set of parameters with non-zero measure under the prior distribution for which the likelihood function is strictly positive, meaning that the marginal tempered data density is positive at the observed data. Throughout this section, we will assume that the object of interest is the posterior mean of the scalar function $h(\theta)$. Extensions to vector-valued functions are straightforward. The convergence of the sequential Monte Carlo approximations requires the existence of moments. We define the classes of functions $\mathcal{H}_1$ and $\mathcal{H}_2$ as

$$
\begin{aligned}
\mathcal{H}_1 &= \left\{ h(\theta) \,\middle|\, \exists\, \delta > 0 \text{ s.t. } \int |h(\theta)|^{1+\delta}p(\theta)d\theta < \infty \right\}, \\
\mathcal{H}_2 &= \left\{ h(\theta) \,\middle|\, \exists\, \delta > 0 \text{ s.t. } \int |h(\theta)|^{2+\delta}p(\theta)d\theta < \infty \right\}
\end{aligned}
$$

and our convergence results will apply to functions in these two classes. Assumption 1 implies that for functions in $\mathcal{H}_j$ $j + \delta$ posterior moments exist for all values of the tempering parameter ϕ.[11]

We follow Chopin (2004) and state the convergence results in an inductive manner, starting from the following assumption:

Assumption 2   (i) $\bar{h}_{n-1,N} \xrightarrow{a.s.} \mathbb{E}_{\pi_{n-1}}[h]$ *for every* $h \in \mathcal{H}_1$.
     (ii) $\sqrt{N}(\bar{h}_{n-1,N} - \mathbb{E}_{\pi_{n-1}}[h]) \Longrightarrow N(0, \Omega_{n-1}(h))$ *for every* $h \in \mathcal{H}_2$.

It is straightforward to verify that Assumption 2 is satisfied for $n = 2$. Recall that $\phi_1 = 0$. According to Step 1 of Algorithm 1, the particles $\{\theta_1^i, W_1^i\}$ are obtained using an *iid* sample from

---

[10] Our Assumptions correspond to Conditions 3 and 4 in Durham and Geweke (2012).

[11] In principle the classes of functions $\mathcal{H}_j$ can be enlarged to functions for which posterior moments exist only if $\phi \geq \underline{\phi} > 0$. In this case one could start the algorithm from $\phi_1 = \underline{\phi}$ using a proposal density that delivers uniformly bounded importance weights.

the prior distribution. Thus, for $h \in \mathcal{H}_1$ and $h \in \mathcal{H}_2$, the moment conditions for the SLLN and the CLT for *iid* random variables are satisfied and $\Omega_1(h) = \mathbb{E}_{\pi_1}\big[(h - \mathbb{E}_{\pi_1}[h])^2\big]$. To present the subsequent results, it is convenient to normalize the incremental weights as follows. Let

$$v_n(\theta) = \frac{Z_{n-1}}{Z_n}[p(Y|\theta)]^{\phi_n - \phi_{n-1}}$$

such that $\mathbb{E}_{\pi_{n-1}}[h(\theta)v_n(\theta)] = \mathbb{E}_{\pi_n}[h(\theta)]$. Theorem 1 summarizes the convergence results for the Monte Carlo approximations obtained after the correction step:

Theorem 1 *[Correction Step] Suppose that Assumptions 1 and 2 are satisfied. Then, $\tilde{h}_{n,N}$ defined in (6) converges as follows: (i) $\tilde{h}_{n,N} \xrightarrow{a.s.} \mathbb{E}_{\pi_n}[h]$ for every $h \in \mathcal{H}_1$; (ii) $\sqrt{N}(\tilde{h}_{n,N} - \mathbb{E}_{\pi_n}[h]) \implies N(0, \tilde{\Omega}_n(h))$, where $\tilde{\Omega}_n(h) = \Omega_{n-1}\big(v_n(\theta)\big(h(\theta) - \mathbb{E}_{\pi_n}[h]\big)\big)$, for every $h \in \mathcal{H}_2$.*

The asymptotic variance $\tilde{\Omega}_n(h)$ has the familiar form of the variance of an importance sampling approximation where the $v_n(\theta)$s correspond to the importance sampling weights. As discussed, for instance, in Liu (2008), a crude approximation of $\tilde{\Omega}_n(h)$ is given by $(N/ESS)\Omega_{n-1}\big(h(\theta) - \mathbb{E}_{\pi_n}[h]\big)$, which provides a rationale for monitoring ESS (see the selection step of Algorithm 1).[12] However, ESS does not directly measure the overall asymptotic variance of the SMC approximation. For the subsequent selection step, we distinguish between the case in which the particles are resampled, i.e. $ESS < N/2$, and the case in which the resampling is skipped.

Theorem 2 *[Selection Step] Suppose that Assumptions 1 and 2 are satisfied. Then, $\hat{h}_{n,N}$ defined in (7) converges as follows: (i) $\hat{h}_{n,N} \xrightarrow{a.s.} \mathbb{E}_{\pi_n}[h]$ for every $h \in \mathcal{H}_1$; (ii) $\sqrt{N}(\hat{h}_{n,N} - \mathbb{E}_{\pi_n}[h]) \implies N(0, \hat{\Omega}_n(h))$ for every $h \in \mathcal{H}_2$. Case (a): if the particles are resampled, then $\hat{\Omega}_n(h) = \tilde{\Omega}_n(h) + \mathbb{V}_{\pi_n}[h]$. Case (b): if the particles not are resampled, then $\hat{\Omega}_n(h) = \tilde{\Omega}_n(h)$.*

A comparison of $\hat{\Omega}_n(h)$ in cases (a) and (b) highlights that the resampling adds noise $\mathbb{V}_{\pi_n}[h]$ to the Monte Carlo approximation. However, it also equalizes the particle weights and thereby reduces the variance in the correction step of iteration $n + 1$. The rule of resampling whenever $ESS < N/2$ tries to strike a balance between this trade-off. To obtain the convergence results for the mutation step, we need the following additional assumption on the transition kernel:

---

[12]Creal, Koopman, and Shephard (2009) suggest various visual indicators of weight performance: (i) a scatter plot of the top 100 weights should be free of outliers; (ii) the histogram of the remaining weights should not be sharply skewed, indicating that many of the particles have very little weight; and (iii) recursive estimates of the variance of the weights should be stable.

Assumption 3 $\pi_n(\theta)$ *is an invariant distribution associated with the transition kernel, that is:* $\int_{\hat{\theta}} K_n(\theta|\hat{\theta};\zeta)\pi_n(\hat{\theta})d\hat{\theta} = \pi_n(\theta)$.

Theorem 3 *[Mutation Step] Suppose that Assumptions 1, 2, and 3 are satisfied. Then, $\bar{h}_{n,N}$ defined in (8) converges as follows: (i) $\bar{h}_{n,N} \xrightarrow{a.s.} \mathbb{E}_{\pi_n}[h]$ for every $h \in \mathcal{H}_1$; (ii) $\sqrt{N}(\bar{h}_{n,N} - \mathbb{E}_{\pi_n}[h]) \Longrightarrow N(0, \Omega_n(h))$ for every $h \in \mathcal{H}_2$. Case (a): if the particles were resampled in the preceeding selection step, then $\Omega_n(h) = \mathbb{E}_{\pi_n}\left[\mathbb{V}_{K_n(\cdot|\theta;\zeta)}[h]\right] + \hat{\Omega}_n\left(\mathbb{E}_{K_n(\cdot|\hat{\theta}_n;\zeta)}[h]\right)$. Case (b): If the last resampling was executed in iteration $n-p$, then $\Omega_n(h) = \mathbb{E}\left[W^2\mathbb{V}_{K_n(\cdot|\theta;\zeta)}[h]\right] + \hat{\Omega}_n\left(\mathbb{E}_{K_n(\cdot|\hat{\theta}_n;\zeta)}[h]\right)$, where*

$$\mathbb{E}\left[W^2\mathbb{V}_{K_n(\cdot|\theta;\zeta)}[h]\right]$$
$$= \int_{\theta_{n-1}} \cdots \int_{\theta_{n-p}} \mathbb{V}_{K_n(\cdot|\theta_{n-1};\zeta)}[h] \left(\prod_{l=1}^{p} v_{n+1-l}^2(\theta_{n-l})\right) \left(\prod_{l=1}^{p-1} K_{n-l}(\theta_{n-l}|\theta_{n-l-1};\zeta)\right)$$
$$\times \pi_{n-p}(\theta_{n-p})d\theta_{n-p}\cdots d\theta_{n-1}.$$

The asymptotic variance $\Omega_n(h)$ consists of two terms. The first term captures the average conditional variance of the Markov transition kernel $K_n(\cdot|\theta;\zeta)$. If the particle weights are not equal to one, then the conditional variance needs to be scaled by the weights, which are functions of the particles from the previous iterations. The second term captures the variance of the average of the conditional expectations $\mathbb{E}_{K_n(\cdot|\hat{\theta}_n^i;\zeta)}[h]$, which are functions of the resampled particles $\hat{\theta}_n^i$. The variance $\Omega_n(h)$ depends on the tuning parameter ζ of the Markov transition kernel.

To study the effect of choosing the tuning parameters ζ adaptively by replacing it with $\hat{\zeta}_n$, where $\hat{\zeta}_n$ is measurable with respect to the σ-algebra generated by $\{\theta_{n-1}^i, \tilde{W}_n^i\}_{i=1}^{N}$, we use the following decomposition

$$\mathbb{E}_{K_n(\cdot|\hat{\theta};\zeta)}[h] = h(\hat{\theta}) + \Psi(\hat{\theta}, \zeta; h). \tag{13}$$

Since $\mathbb{E}_{\pi_n}\left[\mathbb{E}_{K_n(\cdot|\hat{\theta};\zeta)}[h]\right] = \mathbb{E}_{\pi_n}[h]$ by Assumption 3, we deduce that $\mathbb{E}_{\pi_n}[\Psi(\hat{\theta}, \zeta; h)] = 0$ for all ζ. We now make the high-level assumption that $\hat{\zeta}_n$ approaches a limit value ζ as $N \longrightarrow \infty$ and that $\Psi(\hat{\theta}, \zeta; h)$ is differentiable with respect to ζ.

Assumption 4 *Suppose the following conditions are satisfied:*

(i) $\sqrt{N}(\hat{\zeta}_n - \zeta) = O_p(1)$

(ii) $\Psi(\theta, \zeta; h) \in \mathcal{H}_1$ *and* $\Psi(\theta, \zeta; h^2) \in \mathcal{H}_1$ *are twice differentiable and there exist constants M_1 and M_2, independent of θ and ζ, such that $\|\Psi_{\zeta\zeta}(\theta, \zeta; h)\| < M_1$ and $\|\Psi_{\zeta\zeta}(\theta, \zeta; h^2)\| < M_2$.*

Theorem 4 *If Assumption 4 is satisfied then the fixed tuning parameters ζ in Algorithm 2 can be replaced by the adaptive tuning parameters $\hat{\zeta}_n$ in Algorithm 3 without affecting the limit distribution $\Omega_n(h)$ in Theorem 3.*

Since $\mathbb{E}_{\pi_n}[\Psi(\hat{\theta},\zeta;h)] = 0$ for all ζ, it is also the case that the expected values of the derivatives with respect to ζ are equal to zero: $\mathbb{E}_{\pi_n}[\Psi_\zeta(\hat{\theta},\zeta;h)] = 0$ and $\mathbb{E}_{\pi_n}[\Psi_{\zeta\zeta}(\hat{\theta},\zeta;h)] = 0$. The proof relies on the insight that

$$
\begin{aligned}
\frac{1}{\sqrt{N}}\sum_{i=1}^{N}\left(\mathbb{E}_{K_n(\cdot|\hat{\theta}_n^i;\hat{\zeta}_n)}[h] - \mathbb{E}_{K_n(\cdot|\hat{\theta}_n^i;\zeta)}[h]\right) &= \left(\frac{1}{N}\sum_{i=1}^{N}\Psi_\zeta(\hat{\theta}_n^i,\zeta;h)W_n^i\right)\sqrt{N}(\hat{\zeta}_n - \zeta) + o_p(1) \\
&= o_p(1)O_p(1) + o_p(1).
\end{aligned}
$$

The first $o_p(1)$ is obtained by noting that according to Theorem 2, $\frac{1}{N}\sum_{i=1}^{N}\Psi_\zeta(\hat{\theta}_n^i,\zeta;h)W_n^i \xrightarrow{a.s.} \mathbb{E}_{\pi_n}[\Psi_\zeta(\hat{\theta},\zeta;h)] = 0$. In the remainder of this section, we will verify Assumption 4 for a special case of Algorithm 2.

Example: Suppose that $M = 1$, $N_{blocks} = 1$, and $\alpha = 1$. The transition density can be expressed as

$$
K_n(\theta|\hat{\theta};\zeta) = \alpha_n(\theta|\hat{\theta};\zeta)q(\theta|\hat{\theta};\zeta) + r_n(\hat{\theta};\zeta)\delta(\theta|\hat{\theta}), \tag{14}
$$

where $q(\theta|\hat{\theta};\zeta)$ is the proposal density, $\delta(\theta|\hat{\theta})$ is the dirac function,[13] and

$$
\alpha_n(\theta|\hat{\theta};\zeta) = \min\left\{1, \frac{\pi_n(\theta)/q(\theta|\hat{\theta};\zeta)}{\pi_n(\hat{\theta})/q(\hat{\theta}|\theta;\zeta)}\right\}, \quad r_n(\hat{\theta};\zeta) = 1 - \int \alpha_n(\theta|\hat{\theta};\zeta)q(\theta|\hat{\theta};\zeta)d\theta.
$$

The function $r_n(\hat{\theta};\zeta)$ is the probability that the proposed draw is rejected. The function $\Psi(\hat{\theta},\zeta;h)$ is given by

$$
\Psi(\hat{\theta},\zeta;h) = \int h(\theta)\alpha_n(\theta|\hat{\theta};\zeta)q(\theta|\hat{\theta};\zeta)d\theta - h(\hat{\theta})\int \alpha_n(\theta|\hat{\theta};\zeta)q(\theta|\hat{\theta};\zeta)d\theta. \tag{15}
$$

Under the restriction $\alpha = 1$, the conditional acceptance probability does not depend on ζ, that is, $\alpha_n(\theta|\hat{\theta};\zeta) = \alpha_n(\theta|\hat{\theta})$, and the proposal density takes the form

$$
q(\theta|\hat{\theta};\zeta) = (2\pi)^{-k/2}|c_1^2\Sigma_*|^{-1/2}\exp\left\{-\frac{1}{2}(\theta - \hat{\theta})'(c_1^2\Sigma_*)^{-1}(\theta - \hat{\theta})\right\} = \exp\left\{l(\theta|\hat{\theta};\zeta)\right\}.
$$

---

[13]It has the properties that $\delta(\theta|\hat{\theta}) = 0$ for $\theta \neq \hat{\theta}$ and $\int \delta(\theta|\hat{\theta})d\theta = 1$.

Its derivatives are given by

$$q_\zeta(\theta|\hat\theta;\zeta) = l_\zeta(\theta|\hat\theta;\zeta)q(\theta|\hat\theta;\zeta) \quad \text{and} \quad q_{\zeta\zeta}(\theta|\hat\theta;\zeta) = \left[l_\zeta(\theta|\hat\theta;\zeta)l_\zeta'(\theta|\hat\theta;\zeta) + l_{\zeta\zeta}(\theta|\hat\theta;\zeta)\right]q(\theta|\hat\theta;\zeta).$$

The derivatives of the log density with respect to ζ, $l_\zeta(\theta|\hat\theta;\zeta)$, and $l_{\zeta\zeta}(\theta|\hat\theta;\zeta)$ are polynomial functions of θ and therefore integrable with respect to θ under the Gaussian proposal density $q(\theta|\hat\theta;\zeta)$. Thus, the differentiability condition of Assumption 4 is satisfied.

We now verify that $\sqrt{N}(\hat\zeta_n - \zeta)$ is stochastically bounded. Let $\Sigma_* = \mathbb{V}_{\pi_n}[\theta]$ and define $\tilde\Sigma_n$ as the Monte Carlo approximation of Σ_* based on the particles $\{\theta_{n-1}^i, \tilde W_n^i\}_{i=1}^N$. Then, Theorem 1 implies that $\sqrt{N}\big(vech(\tilde\Sigma_n) - vech(\Sigma_*)\big) = O_p(1)$. The empirical rejection rate in iteration $n-1$ is given by

$$\hat R_{j,n-1}(\hat\zeta_{n-1}) = \frac{1}{N}\sum_{i=1}^{N}\{\theta_{n-1}^i = \hat\theta_{n-1}^i\}$$

and has expected value

$$\mathbb{E}[\hat R_{j,n-1}(\hat\zeta_{n-1})|\hat F_{n-1,N}] = \frac{1}{N}\sum_{i=1}^{N} r_{n-1}(\hat\theta_{n-1}^i; \hat\zeta_{n-1}).$$

The arguments used in the proof of Theorem 3 can be modified to verify that conditional on $\hat{\mathcal{F}}_{n-1,N}$, $\sqrt{N}\big(\hat R_{j,n-1}(\hat\zeta_{n-1}) - \mathbb{E}[\hat R_{j,n-1}(\hat\zeta_{n-1})|\hat F_{n-1,N}]\big)$ converges in distribution to a Gaussian limit and is therefore $O_p(1)$. Using the relationship between $\hat c_n$ and the rejection rate in iteration $n-1$ specified in Algorithm 3, we deduce that $\sqrt{N}(\hat c_n - c) = O_p(1)$. $\square$

Verifying the regularity conditions for the general transition kernel associated with Algorithm 2 is more tedious. For $M > 1$ or $N_{blocks} > 1$, the representation of the transition kernel's density in (14) involves additional point masses and the expression for $\Psi(\hat\theta, \zeta; h)$ in (15) becomes more complicated. For $\alpha < 1$, it is no longer true that the conditional acceptance probability $\alpha_n(\theta|\hat\theta;\zeta)$ is invariant to ζ. Due to the min operator in the definition of $\alpha_n(\cdot)$, there are points at which the function is no longer differentiable with respect to ζ. Thus, rather than verifying the differentiability of the integrants in (15) directly one has to integrate with respect to θ separately over the regions $\Theta_l = \{\theta \mid \pi_n(\theta)q(\hat\theta|\theta,\zeta) \le \pi_n(\hat\theta)q(\theta|\hat\theta,\zeta)\}$ and $\Theta_u = \Theta\backslash\Theta_l$ and show that the boundary $\partial\Theta_l$ is a smooth function of ζ.

5 Applications

We now consider three applications of the proposed SMC algorithm and provide comparisons to a standard RWMH algorithm. Section 5.1 evaluates the posterior distribution of a small stylized state-space model based on simulated data. In Section 5.2 we consider the SW model. Typically, the SW model is estimated under a fairly informative prior distribution that leads to a well-behaved posterior distribution when combined with U.S. data. However, under a less informative prior the posterior distribution becomes more irregular and provides an interesting application for our algorithm. Finally, in Section 5.3 we apply the algorithm to a real business cycle model proposed by Schmitt-Grohé and Uribe (2012) which, in addition to standard shocks, is driven by a large number of anticipated shocks, e.g., news about future changes in technology. These news shocks make parameter identification more difficult. The SW model and the SGU model are estimated based on actual U.S. data.

5.1 A Stylized State-Space Model

To illustrate the difficulties that can arise when generating draws from the posterior density $p(\theta|Y)$, consider the following stylized state-space model discussed in Schorfheide (2010):

$$y_t = [1\ 1]s_t, \quad s_t = \begin{bmatrix} \phi_1 & 0 \\ \phi_3 & \phi_2 \end{bmatrix} s_{t-1} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \epsilon_t, \quad \epsilon_t \sim iidN(0,1). \tag{16}$$

The mapping between some structural parameters $\theta = [\theta_1, \theta_2]'$ and the reduced-form parameters $\phi = [\phi_1, \phi_2, \phi_3]'$ is assumed to be

$$\phi_1 = \theta_1^2, \quad \phi_2 = (1 - \theta_1^2), \quad \phi_3 - \phi_2 = -\theta_1\theta_2. \tag{17}$$

The first state, $s_{1,t}$, looks like a typical exogenous driving force of a DSGE model, e.g., total factor productivity, while the second state $s_{2,t}$ evolves like an endogenous state variable, e.g., the capital stock, driven by the exogenous process and past realizations of itself. The mapping from structural to reduced-form parameters is chosen to highlight the identification problems endemic to DSGE models. First, θ_2 is not identifiable when θ_1 is close to 0, since it enters the model only multiplicatively. Second, there is a global identification problem. Root cancellation in the AR and MA lag polynomials for y_t causes a bimodality in the likelihood function. To illustrate this, we

Figure 1: STATE SPACE MODEL: LOG LIKELIHOOD FUNCTION AND POSTERIOR DRAWS

Contour Plot

Density Estimates of $\pi_n(\theta)$
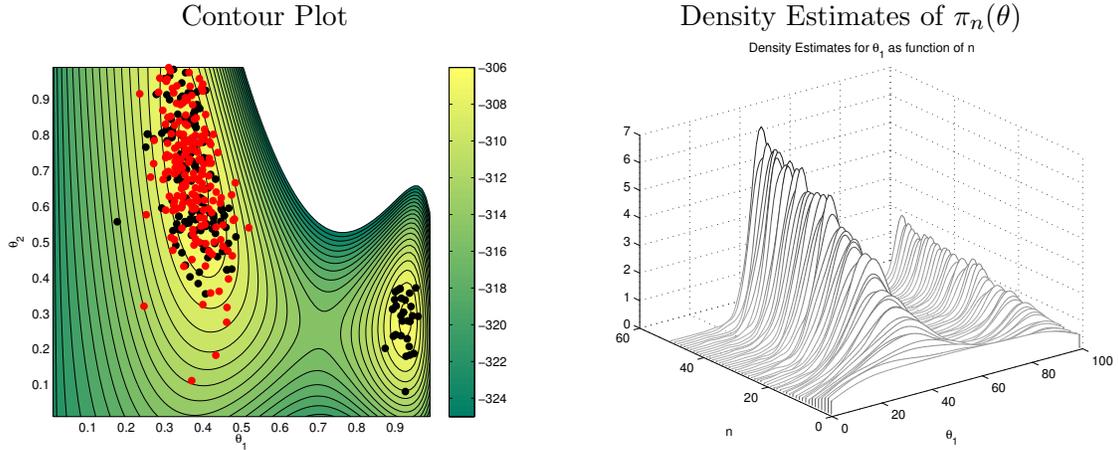
Notes: The left panel shows a contour plot of the log likelihood function overlaid with draws from the RWMH (red) and SMC (red) algorithms. The right panel shows sequential density estimates of $\pi_n(\theta_1)$ using output from the SMC algorithm. Here $n = 1, \ldots, 50$, $\phi_1 = 0$, and $\phi_{50} = 1$.

simulate $T = 200$ observations given $\theta = [0.45, 0.45]'$. This parameterization is observationally equivalent to $\theta = [0.89, 0.22]'$. Moreover, we use a prior distribution that is uniform on the square $0 \le \theta_1 \le 1$ and $0 \le \theta_2 \le 1$.

Tuning of Algorithms. The SMC algorithm is configured as follows. We set $N_\phi = 50$, $N = 1024$, and $\lambda = 1$. Since this is a very simple (and correctly specified) model, we use only $N_{blocks} = 1$ block, $M = 1$, and set $\alpha = 0.9$. The SMC algorithm works extremely well for this small problem, so changing the hyperparameters does not change the results or running time very much. For comparison, we also run the standard RWMH algorithm with a proposal density that is a bivariate independent normal scaled to achieve a 25% acceptance rate. Changing this proposal variance does not materially affect the results.

Results. The left panel of Figure 1 plots the contours of the posterior density overlaid with draws from the SMC algorithm (black) and the RWMH (red). It is clear that the RWMH algorithm fails to mix on both modes, while the SMC does a good job of capturing the structure of the posterior. To understand the evolution of the particles in the SMC algorithm, in the right panel of Figure 1 we show estimates of the sequence of (marginal) densities $\pi_n(\theta_1) \propto \int \left[p(Y|\theta_1, \theta_2) \right]^{\phi_n} p(\theta_1, \theta_2) d\theta_2$, where $\phi_n = (n-1)/49$ and $n = 1, \ldots 50$. The density for θ_1 starts out very flat, reflecting the fact that for low n the uniform prior dominates the scaled likelihood. As n increases, the bimodal

structure quickly emerges and becomes very pronounced as ϕ_n approaches one. This plot highlights some of the crucial features of SMC. The general characteristics of the posterior are established relatively quickly in the algorithm. For larger DSGE models, this feature will guide our choice set $\lambda > 1$, as the early approximations are crucial for the success of the algorithm. As the algorithm proceeds, the approximation at step n provides a good importance sample for the distribution at step $n + 1$.

5.2 The Smets-Wouters Model

The SW model is a medium-scale macroeconomic model that has become an important benchmark in the DSGE model literature. The model is typically estimated with data on output, consumption, investment, hours, wages, inflation, and interest rates. The details of the model, which we take exactly as presented in SW, are summarized in the Online Appendix. In our subsequent analysis, we consider two prior distributions. The first prior, which we refer to as the *standard prior*, is the one used by SW and many of the authors that build on their work. Our second prior is less informative than SW's prior and we will refer to it as the *diffuse prior*. However, the second prior is still proper.

Comparison Among Algorithms. We compute posterior moments based on our proposed SMC algorithm as well as a standard RWMH algorithm. To assess the precision of the Monte Carlo approximations, we run both algorithms 20 times and compute means and standard deviations of posterior moment estimates across runs. To the extent that the number of particles is large enough for the CLTs presented in Section 4 to be accurate, the standard deviations reported below for the SMC can be interpreted as simulation approximations of the asymptotic variance (for $n = N_\phi$) that appears in Theorem 3. We have taken some care to ensure an "apples-to-apples" comparison by constraining the processing time to be roughly the same across algorithms. Given our choice of tuning parameters (see below), the SMC algorithm for the SW model with the standard prior (diffuse prior), runs about 1 hour and 40 minutes (2 hours and 30 minutes) using two twelve-core Intel Xeon X5670 CPUs (24 processors in total) in parallel. The code is written in Fortran 95 and uses the distributed-memory communication interface MPI.

When comparing the SMC and RWMH algorithms, it is important to avoid giving the SMC algorithm an advantage simply because it can more easily exploit parallel programming techniques. In principle, we could instead run 24 copies of the RWMH on separate processor cores and merge

the results afterwards. This may reduce sampling variance if each of the RWMH chains has reliably converged to the posterior distribution. However, if there is a bias in the chains – because of, say, the failure to mix on a mode in a multimodal posterior or simply a slowly converging chain – then merging chains will not eliminate that bias. Moreover, choosing the length of the "burn-in" phase may become an issue as discussed in Rosenthal (2000). Instead, we use a poor-man's parallelization of the RWMH algorithm. It is possible to parallelize MH algorithms via pre-fetching as discussed in Strid (2009). Pre-fetching tries to anticipate the points in the parameter space that the MH algorithm is likely to visit in the next k iterations and executes the likelihood evaluation for these parameter values in parallel. Once the likelihood values have been computed one can quickly determine the next k draws. While coding the parallel MCMC algorithm efficiently is quite difficult, the simulation results reported in Strid (2009) suggest that a parallelization using 24 processors would lead to a speedup factor of eight at best. Thus, in our poor-man's parallelization, we simply increase the running time of the RWMH algorithm on a single CPU by a factor of eight. This results in approximately 10 million draws.

Tuning of Algorithms. The hyperparameters of the SMC algorithm for the estimation under the standard prior are $N = 12,000$, $N_\phi = 500$, $\lambda = 2.1$, $N_{blocks} = 3$, $M = 1$, and $\alpha = 0.9$. The total product of the number of particles, stages, and blocks was chosen by the desired run time of the algorithm. The choice of N_ϕ at 500 was somewhat arbitrary, but it ensured that the bridge distributions were never too "different." The parameter λ was calibrated by examining the correction step at $n = 1$. Essentially, we increased λ until the effective sample size after adding the first piece of information from the likelihood was at least $10,000$; roughly speaking, 80% of the initial particles retained substantial weight. We settled on the number of blocks by examining the behavior of the adaptive scaling parameter c in a preliminary run. Setting $N_{blocks} = 3$ ensured that the proposal variances were never scaled down too much for sufficient mutation. For the estimation under the diffuse prior, we increase the number of blocks to $N_{blocks} = 6$. For the RWMH algorithm, we scale the proposal covariance to achieve an acceptance rate of approximately 30% over 5 million draws after a burn-in period of 5 million. Each RWMH chain was initialized with a draw from the prior distribution.

Results from the Standard Prior. It turns out that with the standard prior, the results for the RWMH algorithm and the SMC algorithm are close, both for the posterior means and for the 5% and 95% quantiles of the posterior distribution. The Online Appendix contains a table

Figure 2: SW Model: Estimates of Posterior Means and Quantiles
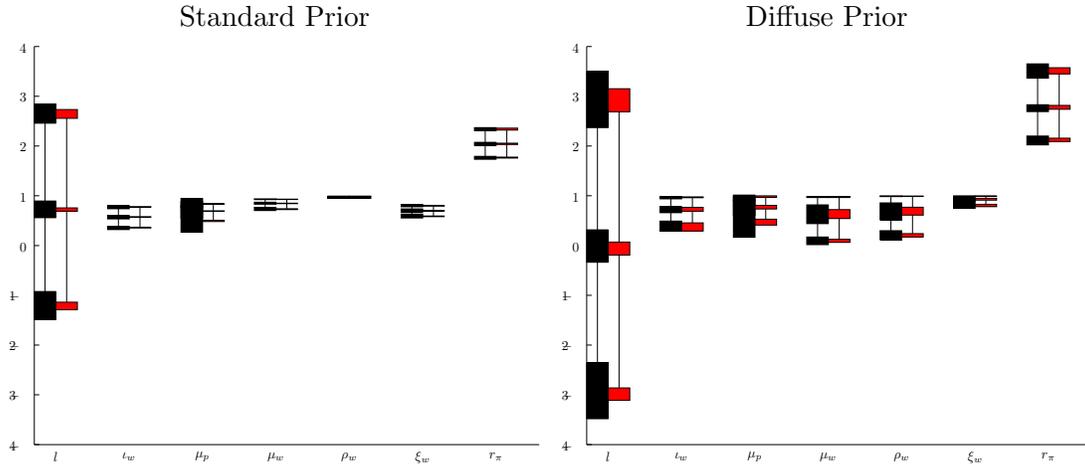
Notes: This figure shows estimates of the mean, 5th and 95th percentile for the RWMH (black) and SMC (red) simulators for the SW model under the standard prior (left) and diffuse prior (right). The boxes are centered at the mean estimate (across 20 simulations) for each statistic while the shaded region shows plus and minus two standard deviations around this mean.

with posterior means as well as 90% equal-tail-probability credible intervals obtained from the two algorithms. A visual comparison of the RWMH and SMC algorithm under the standard prior is provided in the left panel of Figure 2. For seven selected parameters, Figure 2 presents the estimates of the mean, 5th, and 95th percentiles of the posterior distribution. The shaded region covers plus and minus two standard deviations (across runs) around the point estimates for each of the three statistics. The RWMH estimates are shown in black, while the SMC estimates are in red. The figure shows that both simulators are capturing the distribution, although the SMC algorithm is more precise, as indicated by the size of the red boxes relative to the black ones.

The parameter where this difference is most stark is μ_p, which is the moving average term in the exogenous ARMA(1,1) price-markup shock process. For μ_p, the black boxes representing the noise around estimates of the mean, 5th, and 95th percentiles overlap, which indicates substantial convergence problems. A close inspection of the output from the posterior simulators shows the reason for this. On two of the twenty runs for the RWMH algorithm, the simulator becomes trapped in a region with relatively low posterior density. That is, there is a local mode on which the RWMH gets stuck. Note that this mode is small, so that it can be safely ignored when characterizing the posterior distribution, unlike the multimodal posterior distributions seen below. Perhaps most

Table 1: SW Model: Log MDD Estimates

Algorithm (Method)	MEAN(Log MDD)	STD(Log MDD)
Standard Prior		
SMC (Particle Estimate)	-901.739	0.33
RWMH (Modified Harmonic Mean)	-902.626	2.34
Diffuse Prior		
SMC (Particle Estimate)	-873.46	0.24
RWMH (Modified Harmonic Mean)	-874.56	1.20

Notes: Means and standard deviations are over 20 runs for each algorithm.

disturbing is that output from these simulators "looks" fine, in the sense that there are no abrupt shifts in acceptance rates, recursive means, or other summary statistics routinely used as indications of convergence. The SMC algorithm easily avoids getting trapped in the vicinity of this mode, as the relatively small weight on particles in this region ensures they are dropped during a resampling step or mutated.

In addition to the posterior distribution of the parameters, the so-called marginal data density (MDD) $p(Y) = \int p(Y|\theta)p(\theta)d\theta$ plays an important role in DSGE model applications, because it determines the posterior model probability. The MDD is the normalization constant that appears in the denominator of (1) and as constant Z in (2). The top panel of Table 1 shows estimates of the marginal data densities (MDD) associated with the posterior simulators. While the SMC algorithm delivers an estimate of the MDD as a by-product of the simulation, for the RWMH an auxiliary estimator must be used. We use the modified harmonic mean estimator of Geweke (1999). The estimates in Table 1 indicate that the SMC algorithm is a more stable estimator of the MDD. The high standard deviation of the RWMH-based estimate is driven by the two runs that get stuck in a local mode. These runs produce MDD estimates of about -902.6, a much worse fit relative to the SMC algorithm.

The Diffuse Prior. Some researchers argue that the prior distributions used by SW are implausibly tight, in the sense that they seem hard to rationalize based on information independent of the information in the estimation sample. For instance, the tight prior on the steady-state inflation rate is unlikely to reflect *a priori* beliefs of someone who has seen macroeconomic data only from the 1950s and 1960s. At the same time, this prior has a strong influence on the empirical performance of the model, as discussed in Del Negro and Schorfheide (2013). Closely related, Müller (2011)

derives an analytical approximation for the sensitivity of posterior means to shifts in prior means and finds evidence that the stickiness of prices and wages is driven substantially by the priors.

One side benefit of tight prior distributions is that they tend to smooth out the posterior surface by down-weighting areas of the parameter space that exhibit local peaks in the likelihood function but are deemed unlikely under the prior distribution. Moreover, if the likelihood function contains hardly any information about certain parameters and is essentially flat with respect to these parameters, tight priors induce curvature in the posterior. In both cases, the prior information stabilizes the posterior computations. For simulators such as the RWMH this is crucial, as they work best when the posterior is well-behaved. With a view toward comparing the effectiveness of the different posterior simulators, we relax the priors on the SW model substantially. For parameters which previously had a Beta prior distribution, we now use a uniform distribution. Moreover, we scale the prior variances of the other parameters by a factor of three – with the exception that we leave the priors for the shock standard deviations unchanged. A table with the full specification of the prior is available in the Online Appendix.

Results from the Diffuse Prior. Table 2 summarizes the output of the posterior simulators under the diffuse prior, focusing on parameters for which the standard deviation (STD) of the posterior mean approximation under the RWMH algorithm is greater than 0.02 (the results for the remaining parameters are reproduced in the Online Appendix). While the average estimated mean of the posterior seems to be roughly the same across the algorithms, the variance across simulations is substantially higher under the RWMH algorithm. For instance, the standard deviation of the estimate of the mean for ρ_w, the autoregressive coefficient for the wage markup, is 0.09. Given the point estimate of 0.69, this means that any given run of the simulator could easily return a mean between 0.5 and 0.9, even after simulating a Markov chain of length 10 million. In almost all cases, the SMC estimates are about half as noisy, usually substantially less so.

Suppose we were able to generate M *iid* draws from the marginal posterior distribution for a parameter θ. The variance of the mean of these draws would be given by $\mathbb{V}(MEAN) = \mathbb{V}_\pi/M$. Thus, we define $N_{eff} = \hat{\mathbb{V}}_\pi[\theta]/STD^2$, where $\hat{\mathbb{V}}_\pi[\theta]$ is an estimate of the posterior variance of θ obtained from the output of the SMC algorithm and STD^2 is the variance of the posterior mean estimate across the 20 runs of each algorithm. The effective sample sizes N_{eff} are also reported in Table 2. The N_{eff} differences across the two algorithms are striking. The SMC algorithm is substantially more efficient, generating effective sample sizes that are one to two

Table 2: SW Model with Diffuse Prior: Posterior Comparison

		SMC				RWMH		
Parameter	Mean	$[0.05, 0.95]$	STD(Mean)	N_{eff}	Mean	$[0.05, 0.95]$	STD(Mean)	N_{eff}
σ_l	3.06	[1.40, 5.26]	0.04	1058	3.04	[1.41, 5.14]	0.15	60
l	-0.06	[-2.99, 2.92]	0.07	732	-0.01	[-2.92, 2.93]	0.16	177
ι_p	0.11	[0.01, 0.29]	0.00	637	0.12	[0.01, 0.29]	0.02	19
h	0.70	[0.59, 0.78]	0.00	522	0.69	[0.58, 0.78]	0.03	5
Φ	1.71	[1.50, 1.94]	0.01	514	1.69	[1.48, 1.91]	0.04	10
r_π	2.78	[2.12, 3.52]	0.02	507	2.76	[2.11, 3.51]	0.03	159
ρ_b	0.19	[0.03, 0.44]	0.01	440	0.21	[0.03, 0.48]	0.08	3
φ	8.12	[4.27, 12.59]	0.16	266	7.98	[4.16, 12.50]	1.03	6
σ_p	0.14	[0.09, 0.23]	0.00	126	0.15	[0.11, 0.20]	0.04	1
ξ_p	0.72	[0.60, 0.82]	0.01	91	0.73	[0.62, 0.82]	0.03	5
ι_w	0.73	[0.37, 0.97]	0.02	87	0.72	[0.39, 0.96]	0.03	36
μ_p	0.77	[0.47, 0.98]	0.02	77	0.80	[0.54, 0.96]	0.10	3
ρ_w	0.69	[0.21, 0.99]	0.04	49	0.69	[0.21, 0.99]	0.09	11
μ_w	0.63	[0.09, 0.97]	0.05	49	0.63	[0.09, 0.98]	0.09	11
ξ_w	0.93	[0.80, 0.99]	0.01	43	0.93	[0.82, 0.99]	0.02	8

Notes: Means and standard deviations are over 20 runs for each algorithm. The RWMH algorithms use 10 million draws with the first 5 million discarded. The SMC algorithms use 12,000 particles and 500 stages. The two algorithms utilize approximately the same computational resources. We define $N_{eff} = \hat{\mathbb{V}}_\pi[\theta]/STD^2$.

orders of magnitudes larger. N_{eff} for the inverse Frisch elasticity of labor supply σ_l is 1058 for the SMC algorithm and only 60 for the RWMH. The least efficient numerical approximation is obtained for the posterior mean of the parameters governing the wage (ξ_w, ι_w, ρ_w, μ_w) and price (ξ_p, μ_p, σ_p) stickiness. Under the SMC algorithm, N_{eff} ranges from 43 to 126, whereas for the RWMH algorithm, the effective sample size ranges from 1 to 36.

We return to Figure 2, which graphically illustrates the precision of the algorithms estimates of the mean, 5th, and 95th percentiles of selected parameters. For the diffuse prior, the differences across algorithms are more pronounced. First, note that relative to the analysis with the standard prior, the posterior distributions are now more spread out. Second, the precision of the estimates is much lower for the RWMH simulations than for the SMC simulations, as the black boxes are generally much larger than the red boxes. In fact, the performance of the RWMH algorithm is so poor that for the parameters restricted to the unit intervals, the boxes overlap substantially, meaning that there is sufficient noise in the RWMH algorithm to tangle the, say, 5th and 95th percentile of the posterior distribution for μ_p. Moreover, the figure confirms the message from

Figure 3: SW Model with Diffuse Prior: Bivariate Contour Plots
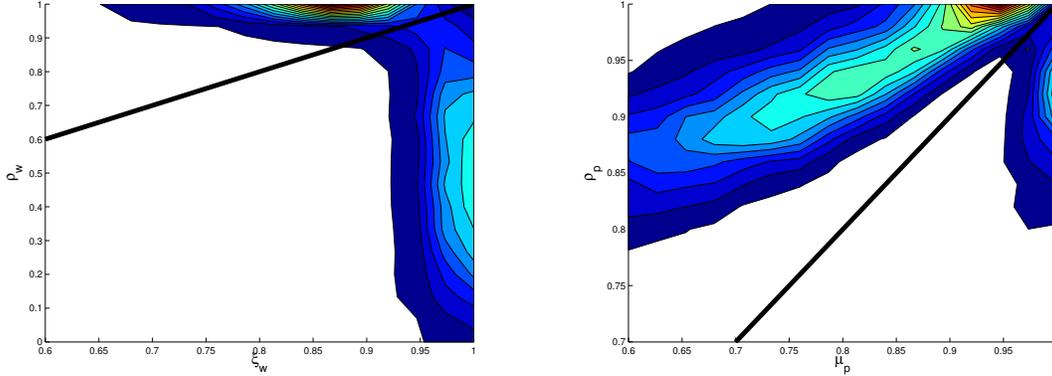
Notes: This figure shows contour plots for bivariate kernel density estimates of the posteriors for $[\rho_w, \xi_w]$ (left) and $[\rho_p, \mu_p]$ (right) from the SMC simulator. The black solid line is the 45° line.

Table 2: some of the worst-performing aspects of the RWMH algorithm are associated with the parameters controlling the movements of wages and prices.

To examine the posterior distribution more closely, we plot joint kernel density estimates of parameters from the wage and price block of the model. The two panels of Figure 3 display joint density estimates of $[\rho_w, \xi_w]$ and $[\rho_p, \mu_p]$. ρ_w is the autocorrelation parameter in the exogenous wage-markup shock process, whereas ξ_w is the wage-Calvo parameter that determines the degree of nominal wage rigidity. The parameters ρ_p and μ_p are the autoregressive and moving-average coefficients of the exogenous price-markup shock process. The black line shows the 45° line. It is clear that both panels display multimodal, irregular posterior distributions, exacerbated by hard parameter boundaries.[14] For the left panel, it is clear that when the autoregressive coefficient for the exogenous wage-markup parameter, ρ_w, is very close to one, the endogenous wage stickiness of the model, embodied in the wage-Calvo parameter ξ_w, is less important. On the other hand, when ξ_w is close to one, the exogenous persistence of wage movements is much lower. The right panel in Figure 3 shows identification problems related to the coefficients of the ARMA(1,1) price-markup shock process.

Motivated by the bimodal features of the bivariate posteriors displayed in Figure 3, we graph estimates of posterior probabilities associated with particular regions in the parameter space in Figure 4. According to Figure 3 there exists a mode for which $\rho_w > \xi_w$ ($\rho_p > \mu_p$) and a mode for

[14]We did some exercises using parameters transformed to unbounded space. The results were essentially unchanged.

Figure 4: SW MODEL WITH DIFFUSE PRIOR: POSTERIOR PROBABILITY STATEMENTS
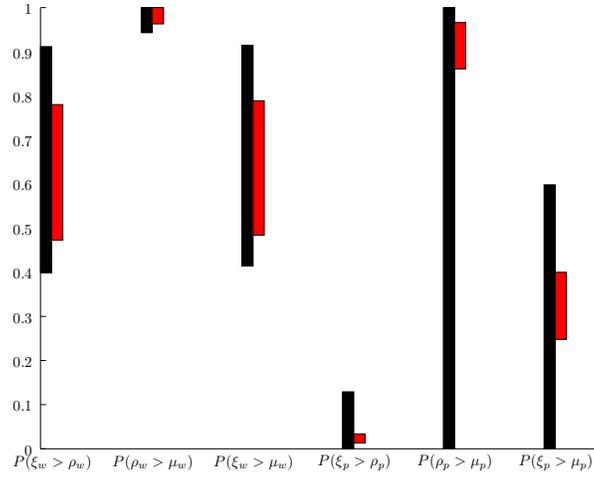
Notes: This figure shows estimates of various posterior probability statements. The black (RWMH) and red (SMC) boxes are centered at the mean (across the 20 simulations) estimate of each probability statement. The shaded region shows plus and minus two standard deviations around this mean.

Table 3: SW MODEL WITH DIFFUSE PRIOR: TWO MODES

Parameter	Mode 1	Mode 2
ξ_w	0.844	0.962
ι_w	0.812	0.918
ρ_w	0.997	0.394
μ_w	0.978	0.267
ξ_p	0.591	0.698
ι_p	0.001	0.085
ρ_p	0.909	0.999
μ_p	0.612	0.937
Log Posterior	-804.14	-803.51

which $\rho_w < \xi_w$ ($\rho_p < \mu_p$). To measure the posterior probabilities associated with these regions it is necessary for the posterior simulator to mix draws from the two modal regions in the right proportion. Figure 4 shows that the RWMH algorithm has severe difficulties approximating the posterior probabilities precisely. The RWMH estimates are highly variable, and in the case of $\mathbb{P}_\pi\{\rho_p \geq \mu_p\}$ range from 0 to 1, whereas the SMC algorithm fairly precisely determines this probability to be around 0.95.

To further explore the multimodal shape of the posterior under the diffuse prior, Table 3 lists the values of the key wage and price parameters at two modes of the joint posterior density captured by the SMC algorithm.[15] With respect to wages, the modes identify two different drivers of fit. At Mode 1, the values of the wage Calvo parameter, ξ_w, and wage indexation parameter, ι_w, are relatively low, while the parameters governing the exogenous wage markup process, ρ_w and μ_w, are high. That is, model dynamics for wages near this mode are driven by the *exogenous* persistence of the shock. At Mode 2, the relative importance of the parameters is reversed. The persistence of wages is driven by the parameters that control the strength of the endogenous propagation, echoing the shape in Figure 3. The exogenous wage markup process is much less persistent. We see that the data support both endogenous and exogenous persistence mechanisms. Thus, a researcher's priors can heavily influence the assessment of the importance of nominal rigidities, echoing a conclusion reached by Del Negro and Schorfheide (2008).

Finally, we return to the marginal data densities, which for the SW model with diffuse prior are reported in the bottom panel of Table 1. The SMC estimate is much more precise, which is unsurprising, given the difficulties of using the modified harmonic mean estimator on a bimodal distribution. The modified harmonic mean estimator actually performs better on the model with diffuse prior than on model with standard prior, mostly because the "height" of the different modes is more similar. Still, the high standard deviation of the estimate indicates severe problems in the algorithm. Finally, as a substantive point, it should be noted that the diffuse prior model has an MDD that is about 30 log points higher than the standard prior SW model. This is a large difference in favor of the diffuse prior model and it highlights the strong tension between the standard prior and the data. This point may apply more generally to DSGE models.

5.3   The Schmitt-Grohe and Uribe News Model

The final application is SGU's "news" model, which has been used to examine the extent to which anticipated shocks (i.e., news) to technology, government spending, wage markups, and preferences explain movements in major macroeconomic time series at business cycle frequencies. The core of the SGU model is a real business cycle model with real rigidities in investment, capital utilization, and wages. While there is not enough space to go through the complete model here, we summarize

---

[15]Note that these two modes of the joint posterior are different from the two modes in the marginal posteriors for $[\rho_w, \xi_w]$ and $[\rho_p, \mu_p]$ seen in Figure 3.

the key features below. The complete equilibrium conditions for this model are summarized in the Online Appendix.

Important Model Features. We focus on two key features of the SGU model. First, the model has seven exogenous processes in total: stationary and nonstationary technology, stationary and nonstationary investment specific technology, government spending, wage markup, and preference shocks. Each of the seven processes is driven by three (independent) innovations, one of which is realized contemporaneously, and two of which are realized four and eight quarters ahead. So, the determination of a given exogenous process x_t looks like:

$$x_t = \rho_x x_{t-1} + \epsilon_t^{x,0} + \epsilon_{t-4}^{x,4} + \epsilon_{t-8}^{x,8}.$$

Each innovation $\epsilon^{x,h}$ is scaled by its own variance, σ_x^h, for horizons $h = 0, 4, 8$. Given the model's seven exogenous processes, there are 21 shocks in total, plus a measurement error for output. This framework allows one to capture anticipated changes in, say, productivity as a driver of the business cycle. This can lead to expectation-driven booms, which are, strictly speaking, ruled out in most DSGE models. In forward-looking models, anticipated shocks can have large effects, so it is crucial for the econometrician to recover plausible values for the size of these shocks. The second key feature of the SGU model is that households have nonstandard preferences. Specifically, the representative agent has constant-relative-risk-aversion (CRRA) preferences defined over V_t, a bundle of consumption (C_t), labor (L_t), and the additional variable S_t:

$$V_t = C_t - bC_{t-1} - \psi L_t^\theta S_t.$$

The parameter b controls the degree to which households display (internal) habit formation, and the parameters ψ and θ are related to the level and elasticity of labor supply, respectively. The variable S_t reflects a geometric average of past habit-adjusted consumption:

$$S_t = (C_t - bC_{t-1})^\gamma S_{t-1}^{1-\gamma}.$$

The presence of S_t, owing to Jaimovich and Rebelo (2009), nests two popular specifications for preferences. As $\gamma \longrightarrow 1$, preferences take the form introduced by King, Plosser, and Rebelo (1988). Here, the labor choice will be related to the intertemporal consumption-savings decision; that is, there will be a wealth effect on labor supply. This is the standard form of preferences used in most

estimated DSGE models. In a model with anticipated shocks, it might be problematic because, if there is a large wealth effect, hours today might fall in response to positive news about future economic conditions. On the other hand, as $\gamma \longrightarrow 0$, preferences take the form of Greenwood, Hercowitz, and Huffman (1988). Labor supply depends only on the current real wage: the wealth effect on labor supply is eliminated. Jaimovich and Rebelo (2009) highlight how crucial this style of preference – coupled with specific adjustment costs – is to generating comovements in response to the news about technology shocks. It is important to note that γ must lie strictly above zero to be compatible with a balanced growth path for the economy.

The model is estimated on seven economic time series: real GDP growth, real consumption growth, real investment growth, real government spending growth, hours, TFP growth and the relative price of investment. The estimation period is 1955:Q1 to 2006:Q4. This model is an interesting application for the SMC algorithm for several reasons. First, the scale of the model is quite large. There are almost 100 states and 35 estimated parameters. Second, the model contains many parameters for which the priors are quite diffuse, relative to standard priors used in macroeconometrics. In particular, this is true for the variances of the shocks. Also, the prior for the Jaimovich-Rebelo parameter, γ, is uniform across $[0, 1]$. We estimate the SGU model under the prior specified by SGU (tabulated in the Online Appendix) and a modified version of this prior in which we change the distribution of the preference parameter γ.

Tuning of Algorithms. We run the same comparison as in previous sections. For each posterior simulator, we compute 20 runs of the algorithm. For the SMC runs, we use $N = 30,048$ particles, $N_\phi = 500$ stages, $\lambda = 2.1$, $N_{blocks} = 6$ blocks, $M = 1$, and $\alpha = 0.9$. We choose the tuning parameters in a way similar to that for the SW model. We use many more particles because the parameter space is larger and the prior is more diffuse compared to the SW model. Given the high-dimensional parameter vector, we increase the number of blocks from 3 to 6 to ensure that mutation can still occur. For the RWMH algorithm, we simulate chains of length 10 million, considering only the final 5 million draws. For the proposal density we use a covariance matrix computed from an earlier run, scaled so that the acceptance rate is roughly 32%. The SMC algorithm takes about 14 hours to run while the RWMH algorithm takes about 4 days. This suggests that a parallelized RWMH algorithm with a speed-up factor of 6.85 would take roughly as long as the SMC algorithm. This is in line with the estimates of Strid (2009), so we take it that the algorithms are on roughly equal footing in terms of computing power.

Table 4: Posterior Comparison for News Model

	SMC				RWMH			
Parameter	Mean	$[0.05, 0.95]$	STD(Mean)	N_{eff}	Mean	$[0.05, 0.95]$	STD(Mean)	N_{eff}
σ_{zi}^4	3.14	$[\,0.21, 7.98]$	0.04	4190	3.08	$[\,0.21, 7.80]$	0.24	108
σ_g^8	0.41	$[\,0.03, 0.99]$	0.01	1830	0.41	$[\,0.03, 0.98]$	0.03	108
σ_{zi}^8	5.59	$[\,0.75, 10.59]$	0.09	1124	5.68	$[\,0.87, 10.54]$	0.30	102
θ	4.13	$[\,3.19, 5.18]$	0.02	671	4.14	$[\,3.22, 5.19]$	0.05	146
σ_{zi}^0	12.27	$[\,9.07, 15.84]$	0.09	640	12.36	$[\,9.05, 16.12]$	0.09	616
σ_μ^0	1.04	$[\,0.06, 2.79]$	0.04	626	0.92	$[\,0.06, 2.39]$	0.04	625
σ_g^0	0.62	$[\,0.06, 1.08]$	0.01	609	0.60	$[\,0.06, 1.07]$	0.03	111
κ	9.32	$[\,7.48, 11.33]$	0.05	578	9.33	$[\,7.49, 11.40]$	0.09	208
σ_ζ^4	2.43	$[\,0.15, 5.95]$	0.09	406	2.44	$[\,0.15, 6.04]$	0.04	2066
σ_ζ^0	3.82	$[\,0.50, 6.77]$	0.10	384	3.80	$[\,0.51, 6.78]$	0.22	73
σ_ζ^8	2.65	$[\,0.17, 6.22]$	0.11	335	2.62	$[\,0.17, 6.07]$	0.18	126
σ_μ^4	4.26	$[\,0.28, 5.91]$	0.24	49	4.33	$[\,0.84, 5.92]$	0.49	12
σ_μ^8	1.36	$[\,0.03, 5.14]$	0.24	46	1.34	$[\,0.04, 4.83]$	0.49	11

Notes: Means and standard deviations are over 20 runs for each algorithm. The RWMH algorithms use 10 million draws with the first 5 million discarded. The SMC algorithms use 30,048 particles and 500 stages. The two algorithms utilize approximately the same computational resources. We define $N_{eff} = \hat{\mathbb{V}}_\pi[\theta]/STD^2$.

Results from the SGU Prior. Table 4 displays the posteriors for some of the parameters of the SGU model with the standard prior. It is clear that the average posterior means from the RWMH and SMC algorithms are roughly the same, with a few exceptions, but that the posteriors generated by the RWMH algorithm are substantially noisier. Mean estimates of parameters associated with news about the wage markup, σ_μ^4 and σ_μ^8, have standard deviations about twice as large under the RWMH simulator relative to the SMC simulator. Related, in terms of coverage, the posterior credible interval for the standard deviation of the contemporaneous wage markup shock, σ_μ^0, has a 95% quantile estimate of 2.79 under the SMC algorithm and 2.39 under the RWMH algorithm. Furthermore, the estimate of the mean of the Jaimovich-Rebelo parameter, γ, under the SMC algorithm is outside the 90% credible interval generated by the RWMH algorithm (not listed in Table 4). Estimates related to news about the investment specific technology shock, σ_{zi}^4 and σ_{zi}^8, similarly contain much more Monte Carlo noise under the RWMH. Using the numeric efficiency measure discussed in Section 5.2, there are about 3 times as many independent draws in the SMC samples relative to the RWMH samples, based on the median parameter.

To shed light on the relative performance of the simulators, Figure 5 displays histograms for the

Figure 5: NEWS MODEL: RWMH AND SMC HISTOGRAM OF WAGE MARKUP PROCESS
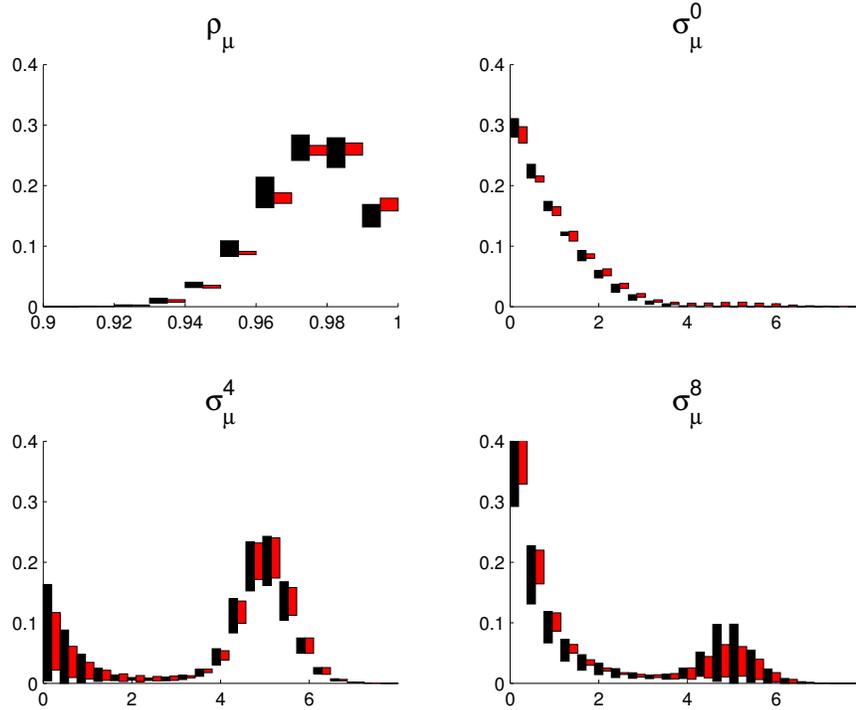
Notes: The boxes show estimates of histograms from each of the simulators. The RWMH algorithm is in red and the SMC is in black. Each box is centered at the mean of the number of elements in each, while the height of the box indicates plus or minus two standard deviations around this mean.

marginal posterior of parameters associated with the wage markup process. For each of the 20 runs we compute separate histograms. Each box is centered at the mean relative histogram height across the 20 runs for the RWMH (black) and SMC (red) algorithms. The shaded rectangles span plus and minus two standard deviations (across the 20 runs) about this mean. We choose evenly-sized bins for the histograms, restricting the span of the bins to a range with non-trivial posterior density. In the upper left panel, the histogram estimates for ρ_μ, the persistence of the wage markup, shows that the RWMH and SMC algorithm capture roughly the same posterior shape. The size of the black boxes relative to the red boxes shows that the SMC estimator is much more precise. For the parameter σ_μ^0, the standard deviation of unanticipated markup shock, the posteriors appear very slightly different, as the SMC algorithm has a long tail, consistent with the difference in credible intervals discussed above.

Finally, the bottom two panels of Figure 5 indicate that the RWMH has the most trouble with the multimodalities and irregularities associated with the variances of the news shocks. Looking

Figure 6: NEWS MODEL: RWMH AND SMC HISTOGRAMS OF γ
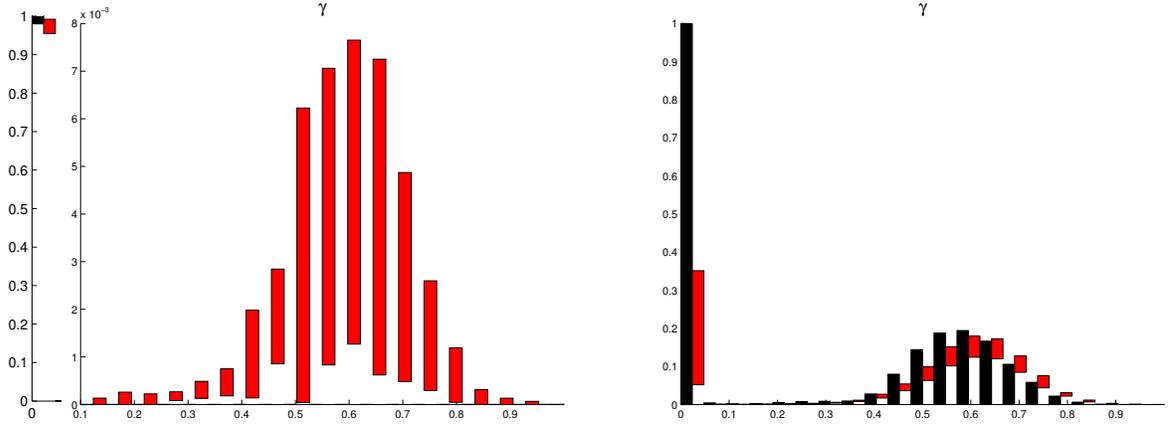
Notes: The boxes show estimates of histograms from each of the simulators. The RWMH algorithm is in red and the SMC is in black. The left panel shows output from the standard prior model and the right panel shows output from the diffuse prior model. Each box is centered at the mean of the number of elements in each, while the height of the box indicates plus or minus two standard deviations around this mean.

at the size of the boxes, it is clear that these bimodal distributions are more precisely estimated using SMC. For instance, σ_μ^4, the size of the four-quarter-ahead news about wage markups, has two sharp peaks, one roughly at 0.35, and another at 4.9. The reason for the two peaks is a (global) identification problem. The model has difficulty distinguishing markup news at 4-quarter-ahead frequency from 8-quarter ahead. This is borne out by the density of σ_μ^8, which has a similar bimodal shape as the density of σ_μ^4, although the heights are peaked differently.[16] Indeed, the posterior correlation of the two parameters is -0.9. The Online Appendix plots a kernel density estimate of the bivariate posterior. In this particular instance, the economic effects of anticipated shocks are unchanged by the failure of the RWMH to mix properly, precisely because of the identification problem: the effects of an eight- and four-periods-ahead shock are very similar. On the other hand, fine parsings of the structural results are incorrect under RWMH. For instance, Table 6 of SGU reports that eight-periods-ahead shocks are responsible for about five percent of the variation in hours worked. Our RWMH runs that do not mix on the second σ_μ^8 are consistent with that result. The SMC algorithm, which properly reflects the bimodal structure of the shock variances, puts that number close to 20% and, in general, places more importance on longer run news about wages.

With respect to the Jaimovich-Rebelo parameter, γ, the posteriors generated by the RWMH

---

[16]To examine whether the bimodality is specific to the U.S. data that are used to estimate the SGU model, we re-estimated the model based on simulated data. The posterior obtained from simulated data exhibits a similar bimodal shape as the posterior given the actual data.

and the SMC algorithm yield small but interesting differences. The left panel of Figure 6 displays the histograms for the posterior for γ. The histograms are set up as in Figure 5, with the exception that the left panel has a different scale for the smallest bin. The RWMH algorithm essentially finds a "pointmass" very close to zero. All of the draws from the RWMH runs are less than 0.03, which we refer to as Mode 1. The SMC algorithm also finds significant posterior probability mass near zero, but in addition it detects a small mode (Mode 2) around $\gamma = 0.6$, comprising about 3% of the total posterior. This can be seen again by looking at the red boxes to the right of 0.1, vs. the non-existent black boxes. Keep in mind though, that the scale for Mode 2 is magnified relative to Mode 1.

The two modes have very different economic implications. Near Mode 2 the wealth effect associated with an increase in income is nonzero. This changes the dynamics of the news model: positive news about the future tend to decrease labor supplied today, because consumers anticipate higher income in the future. Indeed, a parameter value of $\gamma \approx 0.6$ implies that the importance of anticipated shocks for hours is substantially diminished. To compensate for the reduced importance of anticipated shocks, values of $\gamma \approx 0.6$ are associated with high standard deviations of the unanticipated wage-markup shock. Overall, a variance decomposition implies that conditional on $\gamma \approx 0.6$, unanticipated movements in the wage markup account for about 50% of the movement in hours, compared with just 10% of the movement when $\gamma \approx 0$. Thus, for parameters near Mode 2, movements in hours are not principally driven by news.

Results from a Modified Prior. To highlight this potential pitfall more clearly, we re-estimate the news model under a different prior for γ, namely, $\gamma \sim \text{Beta}(2,1)$. Relative to the uniform prior, this prior places more mass on the region near 1. Its probability density function is a 45-degree line. A possible justification for this prior is that it places more weight on the "standard" utility functions used in the macroeconomics literature. Mechanically, this prior causes Mode 2 in the original SGU model to become more important. Indeed, now most of the mass resides in this region, although the peak at $\gamma \approx 0$ is still important. The heights of the two modes in the posterior distribution of γ is approximately equal under the modified prior. The right panel of Figure 6 displays the histogram estimates for γ under the new prior for 20 runs each of the RWMH (black) and the SMC (red) algorithms. One can see that the estimates for γ are now extremely noisy, place anywhere from 0 to 1 probability mass in the region $[0, 0.05]$ for the RWMH, indicating that some runs of the RWMH did not mix on both modes. The reason for this is that the peak around

Figure 7: News Model: Anticipated Shocks' Variance Shares for Hours
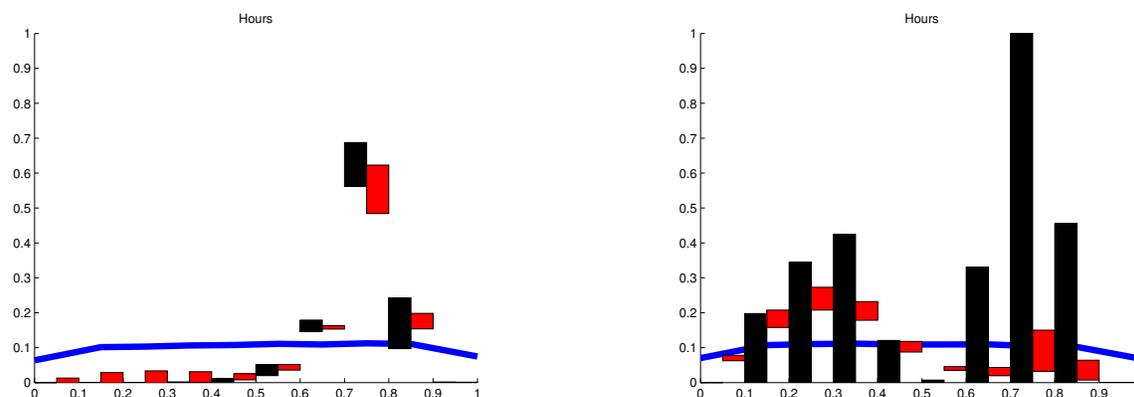
Notes: The boxes show estimates of histograms for the share of long-run variance in hours attributable to news shocks from each of the simulators. The RWMH algorithm is in red and the SMC is in black. The prior share is shown with the blue line. The left panel shows output from the standard prior model and the right panel shows output from the modified prior model. Each box is centered at the mean of the number of elements in each, while the height of the box indicates plus or minus two standard deviations around this mean.

$\gamma \approx 0$ is extremely sharp and the valley around is very deep. If the RWMH gets to this region, it is extremely unlikely to ever leave it. On the other hand, the SMC algorithm mixes on both modes easily.

To see how this translates into problems for substantive economic inference, Figure 7 plots the histograms for the share of the long-run variance in hours accounted for by anticipated shocks under both priors along with the prior shares. Under the standard prior, both simulators generate roughly the same posterior variance shares, although the SMC has a slightly longer left tail due to the small second mode, and to be sure, this tail is relatively imprecisely estimated (as it is small). Once the prior for γ is changed to a $\mathrm{Beta}(2, 1)$ distribution, the problems associated with the RWMH are exacerbated and inference is substantially changed. As with the posterior for γ, the RWMH places anywhere from 0 to 1 probability mass on the anticipated shocks accounting for 80% of the long-run movements in hours. The bimodal shape of the posterior distribution distribution of the news shock share under the modified prior is much more precisely estimated by the SMC algorithm and the economic inference is not overwhelmed by Monte Carlo noise. Relative to the standard prior, news shocks are much less important in determining the behavior of hours.

Table 5: Log MDD Estimates for News Model

Algorithm (Method)	MEAN(Log MDD)	STD(Log MDD)
SGU Prior		
SMC (Particle Estimate)	-1802.15	0.23
RWMH (Modified Harmonic Mean)	-1800.77	0.35
Modified Prior		
SMC (Particle Estimate)	-1805.00	0.11
RWMH (Modified Harmonic Mean)	-1836.42	7.72

Notes: Means and standard deviations are over 20 runs for each algorithm.

Finally, we report log marginal data density estimates in Table 5. As in the case of the Smets-Wouters model, the modified harmonic mean estimate computed from the output of the RWMH algorithm is more noisy, in particular under the modified prior. It turns out that the data slightly favor the model specification with the original SGU prior distribution, but further modifications, e.g. larger prior variances, may easily overturn this ranking. While the analysis under the modified prior was designed to highlight the merit of the SMC sampler, it makes an empirical point as well: Conclusions about the importance of news as a drivers of the business cycle are very sensitive to prior choices. By pushing the prior for γ towards more standard values in the literature, we were able to reduce the effect of news on hours substantially. This result signals cautious interpretation of DSGE model-based estimates of the importance (or lack thereof) of news on the business cycle.

6 Conclusion

This paper has presented an alternative simulation technique for estimating Bayesian DSGE models. We showed that, when properly tailored to DSGE models, a sequential Monte Carlo technique can be more effective than the random-walk Metropolis Hastings algorithm. The RWMH is poorly suited to dealing with complex and large posteriors, while SMC algorithms, by slowly building a particle approximation of the posterior, can overcome the problems inherent in multimodality. This is important for DSGE models, where the parameters enter the likelihood in a highly nonlinear fashion and identification problems may be present. We have seen in both the SW model and the SGU model that when priors are not very informative, the posterior can possess multimodalities. It is difficult to correctly characterize the posterior with the RWMH in this case. Moreover, the

SMC algorithm has an embarrassingly parallelizable structure. Within each tempering iteration, likelihoods can be evaluated simultaneously. In most programming languages, this kind of parallelism can be employed easily, for example, MATLAB's `parfor` command. Finally, we have shown that the sequential Monte Carlo methods can be used for very large DSGE models. Indeed, it is on large complex distributions that returns to using an SMC algorithm over an MCMC algorithm are highest.

References

AMDAHL, G. (1967): "Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities," *AFIPS Conference Proceedings*, 30, 483–485.

CAPPÉ, O., E. MOULINES, AND T. RYDEN (2005): *Inference in Hidden Markov Models*. Springer Verlag.

CHIB, S., AND S. RAMAMURTHY (2010): "Tailored Randomized Block MCMC Methods with Application to DSGE Models," *Journal of Econometrics*, 155(1), 19–38.

CHOPIN, N. (2002): "A Sequential Particle Filter for Static Models," *Biometrika*, 89(3), 539–551.

——— (2004): "Central Limit Theorem for Sequential Monte Carlo Methods and its Application to Bayesian Inference," *Annals of Statistics*, 32(6), 2385–2411.

CREAL, D. (2007): "Sequential Monte Carlo Samplers for Bayesian DSGE Models," *Unpublished Manuscript, Vrije Universitiet*.

——— (2012): "A Survey of Sequential Monte Carlo Methods for Economics and Finance," *Econometric Reviews*, 31(3), 245–296.

CREAL, D., S. J. KOOPMAN, AND N. SHEPHARD (2009): "Testing the Assumptions Behind Importance Sampling," *Journal of Econometrics*, 149, 2–11.

CURDIA, V., AND R. REIS (2010): "Correlated Disturbances and U.S. Business Cycles," *Manuscript, Columbia University and FRB New York*.

DEJONG, D. N., B. F. INGRAM, AND C. H. WHITEMAN (2000): "A Bayesian Approach to Dynamic Macroeconomics," *Journal of Econometrics*, 98(2), 203 – 223.

DEL MORAL, P., A. DOUCET, AND A. JASRA (2006): "Sequential Monte Carlo Samplers," *Journal of the Royal Statistical Society, Series B*, 68(Part 3), 411–436.

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(7), 1191–1208.

——— (2013): "DSGE Model-Based Forecasting," in *Handbook of Economic Forecasting*, ed. by G. Elliott, and A. Timmermann, vol. 2, forthcoming. North Holland, Amsterdam.

DURHAM, G., AND J. GEWEKE (2012): "Adaptive Sequential Posterior Simulators for Massively Parallel Computing Environments," *Unpublished Manuscript*.

GEWEKE, J. (1989): "Bayesian Inference in Econometric Models Using Monte Carlo Integration," *Econometrica*, 57(6), 1317–1399.

——— (1999): "Using Simulation Methods for Bayesian Econometric Models: Inference, Development, and Communication," *Econometric Reviews*, 18(1), 1–126.

GREENWOOD, J., Z. HERCOWITZ, AND G. HUFFMAN (1988): "Investment, Capacity Utilization, and the Real Business Cycle," *American Economic Review*, 78(3), 402–417.

HERBST, E. (2011): "Gradient and Hessian-based MCMC for DSGE Models," *Unpublished Manuscript, University of Pennsylvania*.

——— (2012): "Using the "Chandrasekhar Recursions" for Likelihood Evaluation of DSGE Models," *FEDS Paper*, (2012-11).

JAIMOVICH, N., AND S. REBELO (2009): "Can News about the Future Drive the Business Cycle?," *American Economic Review*, 9(4), 1097–1118.

KING, R. G., C. I. PLOSSER, AND S. REBELO (1988): "Production, Growth, and Business Cycles: I The Basic Neoclassical Model," *Journal of Monetary Economics*, 21(2-3), 195–232.

KOHN, R., P. GIORDANI, AND I. STRID (2010): "Adaptive Hybrid Metropolis-Hastings Samplers for DSGE Models," *Working Paper*.

LIU, J. S. (2008): *Monte Carlo Strategies in Scientific Computing*. Springer Verlag.

MÜLLER, U. (2011): "Measuring Prior Sensitivity and Prior Informativeness in Large Bayesian Models," *Manuscript, Princeton University.*

NEAL, R. (2003): "Slice Sampling (with discussion)," *Annals of Statistics*, 31, 705–767.

OTROK, C. (2001): "On Measuring the Welfare Costs of Business Cycles," *Journal of Monetary Economics*, 47(1), 61–92.

POLLARD, D. (2002): *A User's Guide to Measure Theoretic Probability.* Cambridge University Press.

RABANAL, P., AND J. F. RUBIO-RAMÍREZ (2005): "Comparing New Keynesian Models of the Business Cycle: A Bayesian Approach," *Journal of Monetary Economics*, 52(6), 1151–1166.

ROSENTHAL, J. S. (2000): "Parallel Computing and Monte Carlo Algorithms," *Far East Journal of Theoretical Statistics*, 4, 207–236.

SCHMITT-GROHÉ, S., AND M. URIBE (2012): "What's News in Business Cycles?," *Econometrica*, forthcoming.

SCHORFHEIDE, F. (2000): "Loss Function-based Evaluation of DSGE Models," *Journal of Applied Econometrics*, 15, 645–670.

——— (2010): "Estimation and Evaluation of DSGE Models: Progress and Challenges," *NBER Working Paper.*

SMETS, F., AND R. WOUTERS (2007): "Shocks and Frictions in US Business Cycles: A Bayesian DSGE Approach," *American Economic Review*, 97, 586–608.

STRID, I. (2009): "Efficient Parallelisation of Metropolis-Hastings Algorithms Using a Prefetching Approach," *Computational Statistics and Data Analysis*, in press.