WORKING PAPER NO. 12-27
SEQUENTIAL MONTE CARLO SAMPLING FOR DSGE MODELS

Edward Herbst
Federal Reserve Board

Frank Schorfheide
University of Pennsylvania and Visiting Scholar,
Federal Reserve Bank of Philadelphia

November 7, 2012

RESEARCH DEPARTMENT, FEDERAL RESERVE BANK OF PHILADELPHIA
Ten Independence Mall, Philadelphia, PA 19106-1574 • www.philadelphiafed.org/research-and-data/

Sequential Monte Carlo Sampling for DSGE Models

Edward Herbst
Federal Reserve Board

Frank Schorfheide*
University of Pennsylvania
CEPR, NBER, and Visiting Scholar,
Federal Reserve Bank of Philadelphia

November 7, 2012

*Correspondence: E. Herbst: Board of Governors of the Federal Reserve System, 20th Street and Constitution Avenue N.W., Washington, D.C. 20551. Email: edward.p.herbst@frb.gov. F. Schorfheide: Department of Economics, 3718 Locust Walk, University of Pennsylvania, Philadelphia, PA 19104-6297. Email: schorf@ssc.upenn.edu. Thanks to Stephanie Schmitt-Grohé and Martín Uribe for graciously sharing their code. The views expressed in this paper are those of the authors and do not necessarily reflect the views of the Federal Reserve Bank of Philadelphia, the Federal Reserve Board of Governors, or the Federal Reserve System. This paper is available free of charge at www.philadelphiafed.org/research-and-data/publications/working-papers/.

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 consisting of 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 multi-modal and irregular posterior distributions than the widely-used random walk Metropolis-Hastings algorithm. 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 distribution embodied in a DSGE model with a prior distribution for the parameters to yield a posterior distribution that can then be used for inference. Since it is typically infeasible to compute moments of the posterior distribution analytically, simulation methods must be used for posterior inference and decision making. Since 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 - 2010 in eight top economics journals use the RWMH algorithm to facilitate Bayesian estimation of DSGE models.

As the complexity of DSGE models that are being analyzed with Bayesian methods has increased over time, the efficacy of the RWMH algorithm has declined. For instance, it is well documented for DSGE models, 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)).

The contribution of this paper is to 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 usually associated with 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 (2004). 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 a Metropolis-Hastings step.

We tailor the SMC algorithm toward DSGE model applications and illustrate its performance in three applications: the estimation of a stylized small-scale state-space model with cross-coefficient restrictions that generate an identification problem; the estimation of the widely used Smets and Wouters (2007) model (hereafter SW model); and the estimation of a DSGE model with anticipated technology (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 multi-modal posterior distributions, in particular. This is important in complex DSGE models, such as the SGU model and the SW model (under relatively uninformative prior distributions).

Ours is not the first paper to explore alternatives to the version of the RWMH algorithm that has become standard in DSGE model applications. Several papers have tried to address the problematic aspects of 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. Moreover, since MCMC algorithms are inherently sequential, the evaluation of the likelihood function becomes a bottleneck in the estimation of DSGE models as the model complexity increases.

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

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

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. True, 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.

An extensive survey of the theory and applications of SMC methods is provided by Creal (2012). 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). We extend his algorithm in two important ways. First, our algorithm offers more flexibility during the particle mutation phase, which is crucial to be able to explore the complicated posterior surfaces that arise in large-scale DSGE models such as the SW and the SGU model. By grouping parameters into blocks and generating proposal draws from mixture distributions, we are building on insights from the literature on MCMC methods for DSGE models. Second, we make explicit the use of a large parallel framework, which is necessary for the practical implementation of this algorithm on modern DSGE models.

Durham and Geweke (2012) employ a graphical processing unit (GPU) to implement an SMC algorithm using the predictive likelihood distribution as the bridge distribution. 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. 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 apply their SMC algorithm to an EGARCH model as well as several other small-scale reduced-form time series models, but not to large-scale DSGE models. In our experience, the DSGE model application necessitates a more tailored

mutation step than the one included in Durham and Geweke's (2012) generic version of the SMC algorithm.

The remainder of this paper is organized as follows. To motivate the use of SMC algorithms, we provide a stylized example of a state-space model with cross-coefficient restrictions in Section 2. This model generates a complicated posterior density that is difficult to explore with MCMC methods. In Section 3 we review the basic insights underlying SMC methods. The SMC algorithm tailored to DSGE model applications is presented in Section 4. Section 5 contains three numerical illustrations, one pertaining to the stylized model of Section 2 and two based on DSGE model posteriors obtained from actual U.S. data. Section 6 concludes.

2 Why Is Sampling from DSGE Model Posteriors Difficult?

The term *DSGE model* is typically used to refer to a broad class of dynamic macroeconomic models that spans the standard neoclassical growth model discussed in King, Plosser, and Rebelo (1988) as well as the monetary model with numerous real and nominal frictions developed by Christiano, Eichenbaum, and Evans (2005). A common feature of these models is that decision rules of economic agents are derived from assumptions about preferences and technologies by solving intertemporal optimization problems. Moreover, agents potentially face uncertainty with respect to total factor productivity, for instance, or the nominal interest rate set by a central bank. This uncertainty is generated by exogenous stochastic processes that shift technology, for example, or generate unanticipated deviations from a central bank's interest-rate feedback rule.

Conditional on distributional assumptions for the exogenous shocks, the DSGE model generates a joint probability distribution for the endogenous model variables such as output, consumption, inflation, and interest rates. From an econometric perspective the DSGE model solution takes the form of a state-space model with cross-coefficient restrictions. While, in general, this state-space model may be nonlinear, in our illustrations we focus on the case

in which the DSGE model has been solved using a linear approximation technique. The state-transition equation of the resulting state-space system can be written as

$$s_t = \Phi_1(\theta)s_{t-1} + \Phi_\epsilon(\theta)\epsilon_t \tag{1}$$

and the measurement equation is

$$y_t = \Psi_0(\theta) + \Psi_1(\theta)t + \Psi_2(\theta)s_t. \tag{2}$$

Here, y_t denotes the observables and s_t is a vector of (unobserved) state variables whose law of motion is described by (1). The vector ϵ_t represents innovations to the exogenous shock processes. Most importantly, the system matrices $\Phi_i(\cdot)$ and $\Psi_j(\cdot)$ are complicated nonlinear functions of the underlying DSGE model parameters θ. If the vector of innovations ϵ_t has a normal distribution, then the likelihood function, which we denote by $p(Y|\theta)$, can be evaluated with the Kalman filter.[2] For Bayesian inference the likelihood function $p(Y|\theta)$ is combined with a prior density, which we denote by $p(\theta)$. The main object of interest is the posterior density, which is given by

$$p(\theta|Y) = \frac{p(Y|\theta)p(\theta)}{p(Y)}, \quad \text{where } p(Y) = \int p(Y|\theta)p(\theta d\theta. \tag{3}$$

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):

$$y_t = [1\ 1]s_t, \quad s_t = \begin{bmatrix} \phi_1 & 0 \\ \phi_3 & \phi_2 \end{bmatrix} s_{t-1} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \epsilon_t, \quad \epsilon_t \sim iidN(0,1) \tag{4}$$

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

$$\phi_1 = \theta_1^2, \quad \phi_2 = (1 - \theta_1^2), \quad \phi_3 - \phi_2 = -\theta_1\theta_2. \tag{5}$$

[2]Detailed treatments of the Bayesian estimation of DSGE models can be found in the survey by An and Schorfheide (2007) or in the handbook chapter by Del Negro and Schorfheide (2011).

The first state, $s_{1,t}$, looks like a typical exogenous driving force of a DSGE model, while the transition equation $s_{2,t}$ acts as an endogenous variable, driven by the exogenous process and past realizations of itself. This example embodies the standard structure of DSGE models, while the mapping from structural to reduced form parameters is chosen to highlight the identification problems endemic to DSGE models.

Insert Figure 1 about here.

In order to understand the identification problems that arise in this example, it is helpful to rewrite the state-space model as an ARMA(2,1) process for y_t:

$$(1 - \phi_1 L)(1 - \phi_2 L)y_t = (1 - (\phi_2 - \phi_3)L)\epsilon_t. \tag{6}$$

As discussed in Schorfheide (2010), this specification suffers from two identification problems. First, θ_2 is not identifiable when θ_1 is close to 0, since it only enters the model multiplicatively. Second, there is a global identification problem. Root cancellation in the AR and MA lag polynomials for y_t in (6) causes a bimodality in the likelihood function. To illustrate this, we simulate $T = 200$ observations given $\theta = [0.45, 0.45]'$. This parameterization is observationally equivalent to $\theta = [0.89, 0.22]'$. Figure 1 plots the contours of the likelihood function, which can be easily evaluated with the Kalman filter. In order to fully explore the posterior distribution with an MCMC algorithm, the Markov chain has to travel through the (potentially very deep) valley between the two modes, which, in more complicated models, may be prohibitively difficult. In the remainder of this paper we examine the extent to which SMC algorithms can be helpful in accomplishing this task.

3 Sequential Monte Carlo Methods

The object of interest is the posterior density $p(\theta|Y)$ given in (3) with support Θ. To economize on notation, we abbreviate this density by $\pi(\theta) = p(\theta|Y)$. Moreover, we denote the numerator in (3) by $f(\theta) = p(Y|\theta)p(\theta)$. For linearized DSGE models with Gaussian

innovations and a regular prior density the function $f(\theta)$ can be evaluated numerically. Finally, we denote the denominator in (3) by Z, which does not depend on θ. Using this more compact notation

$$\pi(\theta) = \frac{f(\theta)}{Z}. \tag{7}$$

As mentioned above, for DSGE models the posterior density $\pi(\theta)$ is intractable in the sense that it is not possible to characterize its moments analytically. Thus, the posterior moments of θ and transformations $h(\theta)$ are typically computed with Monte Carlo (MC) simulation methods. Most of the Bayesian DSGE literature focuses on using Markov chain Monte Carlo (MCMC) simulations. 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 (2004), Del Moral, Doucet, and Jasra (2006), and Creal (2012).

The keystone of all SMC algorithms is importance sampling: we might try to approximate $\pi(\cdot)$ by using a different, tractable density that is easy to sample from. Let $g(\theta)$ be a positive density with respect to $d\theta$ on the same parameter space Θ. Importance sampling is based on the identity

$$E_\pi[h(\theta)] = \int h(\theta)\pi(\theta)d\theta = \frac{1}{Z}\int_\Theta h(\theta)w(\theta)g(\theta)d\theta, \quad \text{where} \quad w(\theta) = \frac{f(\theta)}{g(\theta)}, \tag{8}$$

This identity holds for any (measurable) function h. If the support of $g(\cdot)$ contains the support of $f(\cdot)$, for any independent sample $\{\theta^i\}_{i=1}^{N_{part}}$ from the distribution with density $g(\cdot)$ the Monte Carlo estimate

$$\bar{h} = \sum_{i=1}^{N_{part}} h(\theta^i)\hat{w}(\theta^i), \quad \text{where} \quad \hat{w}(\theta^i) = \frac{w(\theta^i)}{\sum_{j=1}^{N_{part}} w(\theta^j)}, \tag{9}$$

converges to $E_\pi[h(\theta)]$ as $N_{part} \longrightarrow \infty$ under some mild regularity conditions; see Geweke (1989).

If one chooses $h(\theta) = \mathcal{I}\{\theta \leq \tau\}$, where $\mathcal{I}$ is the indicator function indexed by τ, then (9) implies that we have constructed a "particle" approximation to the distribution associated with $\pi(\cdot)$, which is given by the set of pairs $\{(\theta^i, \hat{w}^i)\}_{i=1}^N$. The $\hat{w}$'s are the (normalized) importance weights assigned to each particle. The suitability of the approximation is driven by the "closeness" of $g(\cdot)$ to $f(\cdot)$ and is reflected in the behavior of the weights. If $g(\cdot)$ is very different from $f(\cdot)$, in particular if $f(\cdot)$ has fatter tails than $g(\cdot)$, the distribution of the weights will be highly skewed, with only a few particles receiving substantive weights. In this case, the approximation will be poor. The estimate $\bar{h}$ will depend only on a few particles, inducing substantial Monte Carlo variance. On the other hand, if $g(\cdot) = f(\cdot)$, that is, we are perfectly sampling from $\pi(\cdot)$, the distribution of the weights is uniform.

In practice, importance sampling from the posterior of a DSGE model is extremely difficult because the shape of $f(\cdot)$ can only be explored numerically and the approximations through tractable densities $g(\cdot)$ tend to be quite poor. This task is even more challenging if the posterior has a non-normal shape containing several peaks and valleys, in which case a suitable density $g(\cdot)$ tends to be elusive. The essence of the SMC methods employed in this paper is to construct sequential particle approximations to intermediate distributions, indexed by n:[3]

$$\pi_n(\theta) = \frac{f_n(\theta)}{Z_n} = \frac{[p(Y|\theta)]^{\phi_n} p(\theta)}{\int [p(Y|\theta)]^{\phi_n} p(\theta) d\theta}, \quad n = 1, \ldots, \phi_{N_\phi} \tag{10}$$

where

$$\phi_1 = 0 \quad \text{and} \quad \phi_{N_\phi} = 1.$$

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 $\pi_n(\cdot)$ to assist in the construction of a particle approximation for $\pi_{n+1}(\cdot)$, one can use iterative approximations to estimate $\pi_{N_\phi} = \pi(\theta)$. We call a function (here the likelihood) raised to a power less than one, a *tempered* function. The process of estimating the parameters of a function through a sequence of tempered functions is known

[3]Using the notation that $Y_{t_1:t_2} = \{y_{t_1}, \ldots, y_{t_2}\}$ and $Y = Y_{1:T}$ one could set $N_\phi = T$ and define $f_n(\theta) = p(Y_{1:n}|\theta)$. This approach is attractive for real-time applications in which the model needs to be re-estimated every period.

as *simulated tempering*. We refer to the sequence tempering parameters as the *tempering schedule* (or heating schedule, due to its connection to simulated annealing.)

Suppose we have an importance sample of $\pi_n(\cdot)$ and we want to use this sample to inform our particle approximation of $\pi_{n+1}(\cdot)$. To avoid confusion as to whether θ is drawn from $\pi_n(\cdot)$ or $\pi_{n+1}(\cdot)$, we equip the parameter vector by a subscript n. Thus, θ_n is associated with the density $\pi_n(\cdot)$. SMC algorithms propagate the particles $i = 1, \ldots, N_{part}$ forward using a Markov transition kernel

$$\theta_{n+1}^i \sim K_{n+1}(\theta_{n+1}|\theta_n^i), \tag{11}$$

which leads to an importance distribution,

$$g_{n+1}(\theta_{n+1}) = \int K_{n+1}(\theta_{n+1}|\theta_n) g_n(\theta_n) d\theta_{n-1}. \tag{12}$$

Unfortunately, the integral in (12) is extremely complex to compute ($O(N_{part}^2)$), and hence the computation of the importance weights $w_{n+1}(\theta) = f_{n+1}(\theta)/g_{n+1}(\theta)$ is essentially infeasible.

Del Moral, Doucet, and Jasra (2006) show that one can augment the target distribution $\pi_n(\cdot)$ to simplify the calculation of the importance weights considerably. Define the backward transition kernels

$$L_{n-k}(\theta_{n-k}|\theta_{n-k+1}), \quad k = 1, \ldots, n-1$$

which represent properly normalized conditional densities of θ_{n-k} given θ_{n-k+1}. Moreover, define the augmented target distributions as

$$\tilde{\pi}_n(\theta_{1:n}) = \frac{\tilde{f}_n(\theta_{1:n})}{Z_n}, \quad \text{where} \quad \tilde{f}_n(\theta_{1:n}) = f_n(\theta) \prod_{k=1}^{n-1} L_{n-k}(\theta_{n-k}|\theta_{n-k+1}), \tag{13}$$

where $\theta_{1:n} = [\theta_1, \ldots, \theta_n]$. The sequential importance sampler is based on the relationship

$$\begin{aligned}
\mathbb{E}_{\pi_n}[h(\theta_n)] &= \frac{1}{Z_n} \int h(\theta_n) \frac{\tilde{f}_n(\theta_{1:n})}{\tilde{g}_n(\theta_{1:n})} \tilde{g}_n(\theta_{1:n}) d\theta_{1:n} \\
&= \left[\frac{1}{Z_n} \int h(\theta_n) f_n(\theta_n) d\theta_n \right] \int \prod_{k=1}^{n-1} L_{n-k}(\theta_{n-k}|\theta_{n-k+1}) d\theta_{1:n-1} \\
&= \frac{1}{Z_n} \int h(\theta_n) f_n(\theta_n) d\theta_n.
\end{aligned}$$

Here $\tilde{g}_n(\cdot)$ is the proposal distribution of the importance sampler which we describe below. The second equality holds because the $\tilde{g}_n(\cdot)$ terms cancel and the transition kernels $L_{n-k}(\cdot|\cdot)$ are normalized to integrate to one. Thus, by construction of $\tilde{f}_n(\theta_{1:n})$ the density of interest $\pi_n(\cdot)$ arises as a marginal density from $\tilde{\pi}_n(\cdot)$.

The sequence of augmented proposal distributions is defined as the product of the transition kernels introduced in (11):

$$\tilde{g}_n(\theta_{1:n}) = g_1(\theta_1) \prod_{k=1}^{n-1} K_{n-k+1}(\theta_{n-k+1}|\theta_{n-k}). \tag{14}$$

The resulting important weights have a convenient recursive representation

$$\frac{\tilde{f}_n(\theta_{1:n})}{\tilde{g}_n(\theta_{1:n})} = \tilde{w}_1(\theta_1) \prod_{k=1}^{n-1} \tilde{w}_{n-k+1}(\theta_{n-k+1}|\theta_{n-k}) \tag{15}$$

with incremental weights

$$\tilde{w}_n(\theta_n|\theta_{n-1}) = \frac{f_n(\theta_n) L_{n-1}(\theta_{n-1}|\theta_n)}{f_{n-1}(\theta_{n-1}) K_n(\theta_n|\theta_{n-1})}. \tag{16}$$

If the backwards kernel is chosen to be

$$L_{n-1}(\theta_{n-1}|\theta_n) = \frac{\pi_n(\theta_{n-1}) K_n(\theta_n|\theta_{n-1})}{\pi_n(\theta_n)},$$

the incremental weights have the form,

$$\tilde{w}_n(\theta_n|\theta_{n-1}) = \frac{f_n(\theta_{n-1})}{f_{n-1}(\theta_{n-1})}. \tag{17}$$

Overall, this leaves us with the representation

$$\begin{aligned}\mathbb{E}_{\pi_n}[h(\theta_n)] &= \frac{1}{Z_n}\int h(\theta_n)\frac{\tilde{f}_n(\theta_{1:n})}{\tilde{g}_n(\theta_{1:n})}\tilde{g}_n(\theta_{1:n})d\theta_{1:n} \\ &= \frac{1}{Z_n}\int h(\theta_n)\left[\tilde{w}_1(\theta_1)\prod_{k=2}^{n}\frac{f_k(\theta_{k-1})}{f_{k-1}(\theta_{k-1})}\right]\left(g_1(\theta_1)\prod_{k=2}^{n}K_k(\theta_k|\theta_{k-1})\right)d\theta_{1:n}.\end{aligned}$$

The term in square brackets corresponds to the importance weights, which are obtained by combining (15) and (17), and the term in parentheses is the proposal distribution given by (12). By setting $\phi_1 = 0$, $\phi_{N_\phi} = 1$, drawing the first set of particles from the prior density, that is $g_1(\theta) = p(\theta)$, and using the definition of $f_n(\theta)$ in (10) we obtain

$$\mathbb{E}_\pi[h(\theta)] = \frac{1}{Z}\int h(\theta_{N_\phi})\left[\prod_{k=2}^{N_\phi}[p(Y|\theta_{k-1})]^{\phi_k-\phi_{k-1}}\right]\left(p(\theta_1)\prod_{k=2}^{N_\phi}K_k(\theta_k|\theta_{k-1})\right)d\theta_{1:N_\phi}. \tag{18}$$

Our choice of $g_1(\cdot)$ implies that $\tilde{w}_1(\theta_1) = 1$. The integral in (18) can be evaluated recursively by setting $W_1(\theta_1) = p(\theta_1)$ and

$$W_n(\theta_n) = \int [p(Y|\theta_{n-1})]^{\phi_n-\phi_{n-1}}W_{n-1}(\theta_{n-1})K_n(\theta_n|\theta_{n-1})d\theta_{n-1}, \quad n = 2,\ldots,N_\phi. \tag{19}$$

The normalization constant Z can be obtained by setting $h(\cdot) = 1$ and exploiting the fact that $\mathbb{E}_\pi[1] = \int \pi(\theta)d\theta = 1$. Overall, we obtain

$$\mathbb{E}_\pi[h(\theta)] = \frac{\int h(\theta_{N_\phi})W_{N_\phi}(\theta_{N_\phi})d\theta_{N_\phi}}{\int W_{N_\phi}(\theta_{N_\phi})d\theta_{N_\phi}}, \tag{20}$$

where the term in the denominator corresponds to the normalization constant (marginal data density)

$$Z = \int p(Y|\theta)p(\theta)d\theta.$$

The algorithms presented in the following section generate Monte Carlo approximations

of (19) and (20).

4 An SMC Algorithm for DSGE Models

The SMC algorithm consists of three steps, using Chopin's (2004) terminology: *correction*, that is, reweighting the particles to reflect the density in iteration n; *selection*, that is, eliminating any particle degeneracy; and *mutation*, that is, propagating the particles forward to adapt to the next bridge density.

Algorithm 1 (Simulated Tempering SMC)

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

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

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

   (a) **Correction.** *Reweight the particles from stage $n-1$ by first computing the incremental weights,*

$$\tilde{w}_n^i = [p(Y|\theta_{n-1}^i)]^{\phi_n - \phi_{n-1}}, \quad i = 1, \ldots, N_{part}.$$

   The total (unnormalized) weights are given by,

$$\tilde{W}_n^i = \tilde{w}_i^n W_{n-1}^i, \quad i = 1, \ldots, N_{part}.$$

   Finally, the normalized weights are

$$W_n^i = \frac{\tilde{W}_n^i}{\sum_{i=1}^{N_{part}} \tilde{W}_n^i}.$$

(b) **Selection.** *First, compute the effective sample size, ESS:*

$$ESS = \frac{1}{\sum_{i=1}^{N_{part}} (W_i^n)^2}.$$

If $ESS < N_{part}/2$*, resample the particles via systematic resampling (see, for example, Robert and Casella (2004) for details). Reset the weights to be uniformly* $1/N_{part}$*.*

(c) **Mutation.** *Propagate the particles* $\{\theta_{n-1}^i\}_{i=1}^{N_{part}} \longrightarrow \{\theta_n^i\}_{i=1}^{N_{part}}$ *via* M *steps of a Metropolis-Hastings algorithm (see below).*

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

$$\widehat{\mathbb{E}_\pi[h(\theta)]} = \sum_{i=1}^{N_{part}} h(\theta_{N_\phi}^i) W_{N_\phi}^i.$$

As is typical in importance samplers, the normalization of the importance weights in Step 2(a) ensures the computation of the normalization constant Z in (18). The recursive calculation of the weights W_n^i in Step 2(a) mimics the recursion in (19). The particle mutation in Step 2(c) amounts to generating draws from the transition kernel $K_n(\theta_n|\theta_{n-1})$ that appears in (19).

Algorithm 1 requires the specification of several tuning parameters. The number of particles $N_{particle}$ ultimately scales the precision of the Monte Carlo approximations. All other things equal, increasing the number of cooling stages, N_ϕ, will decrease the "distance" between bridge distribution and thus make it easier to maintain particle weights that are close to being uniform. The cost to increasing N_ϕ is that, for every additional stage, there must be $N_{part} \times N_{blocks} \times M$ additional likelihood evaluations. To control the shape of the sequence ϕ_n 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.

The transition kernel $K_n(\theta_n|\theta_{n-1})$ is generated through a sequence of M Metropolis-Hastings steps. The parameter M increases the likelihood that the mutation stage will be successful, in the sense that the new particle set resembles the posterior. In practice, the effect of M turned out to be similar to the effect of a change in N_ϕ and we set it equal to $M = 1$ in the applications in Section 5. The Metropolis-Hastings steps are summarized in the following algorithm.

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

1. *Based on* (θ^i_{n-1}, W^i_n) *compute an importance sampling estimate of* $\mathbb{E}_{\pi_n}[\theta]$*, denoted by* $\hat{\theta}$*, and of* $\mathbb{V}_{\pi_n}[\theta]$*, denoted by* $\hat{\Sigma}$*.*

2. *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* $\hat{\theta}_b$ *and* $\hat{\Sigma}_b$ *be the partitions of* $\hat{\theta}$ *and* $\hat{\Sigma}$ *that correspond to the subvector* $\theta_{n,b}$*.*

3. *For each particle* i*, run* M *steps of the following Metropolis-Hastings algorithm. Let* $\theta^i_{n,b,m}$ *be the parameter value for* $\theta^i_{n,b}$ *in the* m*-th iteration with the initialization* $\theta^i_{n,b,0} = \theta^i_{n-1,b}$*. In slight abuse of notation, let*

$$\theta^i_{n,-b,m} = \left[\theta^i_{n,1,m}, \ldots, \theta^i_{n,b-1,m}, \theta^i_{n,b+1,m-1}, \ldots, \theta^i_{n,N_{blocks},m-1}\right].$$

For $m = 1$ *to* M *and* $b = 1$ *to* N_{blocks}*:*

[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 $(i-1)N_{blocks}, \ldots, iN_{blocks}$.

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

$$\begin{aligned}\vartheta_b|(\theta^i_{n,b,m-1}, \theta^i_{n,-b,m}, \hat{\theta}_b, \hat{\Sigma}_b) \\ \sim \quad & \alpha N\left(\theta^i_{n,b,m-1}, c_1^2\hat{\Sigma}_b\right) + \frac{1-\alpha}{2}N\left(\theta^i_{n,b,m-1}, c_2^2 diag(\hat{\Sigma}_b)\right) \\ & + \frac{1-\alpha}{2}N\left(\hat{\theta}_b, c_3^2\hat{\Sigma}_b\right)\end{aligned}$$

and denote the density of the proposal distribution by $q(\vartheta_b|\theta^i_{n,b,m-1}, \theta^i_{n,-b,m}, \hat{\theta}_b, \hat{\Sigma}_b)$.

(b) Define the acceptance probability

$$\begin{aligned}&\alpha(\vartheta_b|\theta^i_{n,b,m-1}, \theta^i_{n,-b,m}, \hat{\theta}_b, \hat{\Sigma}_b) \\ &= \min\left\{1, \frac{p^{\phi_n}(Y|\vartheta_b, \theta^i_{n,-b,m})p(\vartheta_b, \theta^i_{n,-b,m})/q(\vartheta_b|\theta^i_{n,b,m-1}, \theta^i_{n,-b,m}, \hat{\theta}_b, \hat{\Sigma}_b)}{p^{\phi_n}(Y|\theta^i_{n,b,m-1}, \theta^i_{n,-b,m})p(\theta^i_{n,b,m-1}, \theta^i_{n,-b,m})/q(\theta^i_{n,b,m-1}|\vartheta_b, \theta^i_{n,-b,m}, \hat{\theta}_b, \hat{\Sigma}_b)}\right\}\end{aligned}$$

and let

$$\theta^i_{n,b,m} = \begin{cases} \vartheta_b & \text{with probability } \alpha(\vartheta_b|\theta^i_{n,b,m-1}, \theta^i_{n,-b}, \hat{\theta}_b, \hat{\Sigma}_b) \\ \theta^i_{n,b,m-1} & \text{otherwise}\end{cases}$$

4. Let $\theta^i_{n,b} = \theta^i_{n,b,M}$ *for* $b = 1, \ldots, N_{blocks}$.

Algorithm 2 also contains several tuning parameters. Chib and Ramamurthy (2010) and Herbst (2011) have documented that in DSGE model applications blocking of parameters can drastically reduce the persistence in Markov chains generated by the Metropolis-Hastings 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. This could hamper the mutation step. $\alpha \in [0, 1]$ controls the shape of the proposal distribution. For $\alpha = 1$ the Metropolis-Hastings step turns into a pure random walk Metropolis step. For $\alpha < 1$ there is some probability that the covariance matrix $\hat{\Sigma}_b$ is replaced by a diagonal matrix and some probability that the mean is replaced by $\hat{\theta}_b$, which is the Monte Carlo

estimate of $\mathbb{E}_{\pi_{n-1}}[\theta]$. The scaling factors c_1, c_2, and c_3 are chosen adaptively to ensure that the acceptance rate is reasonable, along the lines of Durham and Geweke (2012). The behavior of the scaling factors is used as a guide for whether the initial choice of N_{blocks} was reasonable.

Our implementation of the SMC Algorithm 1 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 multi-modal posterior surfaces. Moreover, the random blocking scheme is important for avoiding bad blocks as the correlation structure of the tempered posterior changes. Finally, 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 impratical, as sample impoverishment occurs immediately. In light of this, Creal (2007) initializes the simulator from a t distribution centered at the posterior mode. For one, this presumes prior knowledge of the mode(s) of the posterior. This approach is not well suited to cases where there are many modes or the posterior is not well characterized by a student-t distribution, as in some of our examples. 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 a sequential Monte Carlo algorithm on large-scale DSGE models.

Second, our implementation exploits advances in parallel computing. For any likelihood-based estimation of DSGE models, the principal computational bottleneck is the evaluation of the likelihood itself.[5] The total number of likelihood evaluations in the SMC algorithm is equal to

$$N_{part} \times N_{blocks} \times N_\phi \times M.$$

[5]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.

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. However, unlike MCMC, the likelihoods do not have to be evaluated in order – indeed they can be evaluated simultaneously. The key here is that Algorithm 2 can run independently for each particle i and thus the computational burden can easily be distributed across multiple processors. In computational terms, the evaluation of a likelihood function is well suited for *single instruction, multiple data* (SIMD) processing. That is, the bulk of the algorithm is spent evaluating the same function (the likelihood) under different data (the particles).

Finally, a discussion about performance measures is necessary. Essentially, SMC is iterative importance sampling, so the major driver of performance is the behavior of the particle weights. The optimal behavior is for the weights to be distributed evenly at every step of the algorithm. When only a few particles have substantial weight, the Monte Carlo variance is high and the performance of the algorithm is erratic. Creal, Koopman, and Shephard (2009), henceforth CKS, suggest various visual indicators of weight performance. First, they consider a scatter plot of the top 100 weights. The figure should be free of outliers. Next, CKS examine a histogram of the remaining weights. A sharply skewed histogram indicates that many of the particles have very little weight, which is problematic. Finally, CKS suggest inspecting recursive estimates of the variance of the weights. If the estimate is erratic, it indicates that a few particles are substantially different in terms of weight. Another effective visualization is the behavior of the effective sample size ESS throughout the simulation. The online Appendix reports some of these statistics for the applications presented in Section 5. In the main text we focus on the stability of the algorithm when it is being run multiple times.

5 Applications

We now consider three applications of the proposed SMC algorithm. Section 5.1 evaluates the posterior distribution of the small state-space model introduced in Section 2 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 less informative priors 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 with anticipated technology shocks proposed by Schmitt-Grohé and Uribe (2012). In addition to standard technology shocks, the model has a large number of anticipated shocks, i.e., news about future changes in technology, that make parameter identification more difficult. The SW model and the SGU model are estimated based on actual U.S. data.

In each of these applications we apply the proposed SMC Algorithm 1 as well as the widely used (in the DSGE model literature) RWMH algorithm. To assess the stability of the algorithm, we run each procedure five times and examine the variation in the simulation output across the repetitions. For our large applications, we make use of a parallel programming environment. We exploit the natural parallelism of the SMC by running it on four twelve-core Intel Xeon X5670 CPUs. The code is written in Fortran 95 and uses the distributed-memory communication interface MPI. On the other hand, we execute the RWMH algorithm on only one processor. In principle, we could instead run the 48 copies of the RWMH on separate processor cores and merge the results. 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 multi-modal 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 – see, for example, Rosenthal (2000). Finally, a single run of the RWMH is more reflective of the way practitioners use it.

5.1 A Stylized State-Space Model

We begin by re-visiting the state-space model defined through the state-space representation (4) and the cross-coefficient restrictions (5). As described in Section 2, we simulate $T = 200$ observations setting $\theta = [0.45, 0.45]'$. Moreover, we use a prior distribution that is uniform on the unit square defined by the constraints $0 \leq \theta_1 \leq 1$ and $0 \leq \theta_2 \leq 1$.

Tuning of Algorithms. The SMC algorithm is configured as follows. We set $N_\phi = 100$, $N_{part} = 1024$, and $\lambda = 1$. Since this is a very simple (and correctly specified) model, we use only 1 block and set $\alpha = 0.10$. The SMC algorithm works extremely well for this small problem, so changing the hyperparameters does not change the results or running time very much. The proposal for the RWMH algorithm is a bivariate independent normal scaled to achieve a 25% acceptance rate. Changing this proposal variance did not materially affect the results.

Insert Figure 2 about here.

Insert Figure 3 about here.

Results. Figure 2 plots the contours of the posterior density overlayed 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 Figure 3 we show density plots of θ_1 as functions of n, which determines the weight placed on the likelihood. To be clear, we are showing estimates of the sequence of (marginal) densities

$$\int p_n(\theta_1, \theta_2|Y)d\theta_2 \propto \int \left[p(Y|\theta_1, \theta_2)\right]^{\phi_n} p(\theta_1, \theta_2)d\theta_2, \quad \phi_n = \frac{n-1}{99}, \quad n = 1, \ldots 100.$$

The density for θ_1 starts out very flat, reflecting the fact that at low n, the uniform prior dominates the scaled likelihood. As n increases, the bimodal structure quickly emerges and becomes very pronounced as n approaches 1. 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 by 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 *diffuse prior*. However, the second prior is still proper.

Tuning of Algorithms. The hyperparameters of the SMC algorithm are $\lambda = 2.1$, $N_{part} = 12,000$, $N_\phi = 500$, and $N_{blocks} = 3$. 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 small 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 500,000 draws. For each algorithm, we run 5 simulations to assess stability. In terms of time, both algorithms run for about 20 minutes. For the SMC algorithm we use 48 processor cores for the likelihood evaluation.

Results from the Standard Prior. It turns out that the results for the RWMH algorithm and the SMC algorithm are nearly identical, both for the posterior means and for the 5% and 95% quantiles of the posterior distribution. The online Appendix contains a table with posterior means as well as 90% equal-tail-probability credible intervals obtained from the two algorithms. Across the five runs of the posterior simulators, the mean estimates are fairly stable, indicating there is little noise due to Monte Carlo variation across the five simulations. It seems that both algorithms have converged to the same posterior distribution. This is unsurprising, given the amount of attention paid to the SW model over the past five years. If anything, our results are contradictory to the the results of Chib and Ramamurthy (2010). They find that using an alternative simulator on the SW model results in larger credible sets for some parameters, in particular the steady-state labor supply $\bar{l}$. The credible sets from the SMC algorithm are (very slightly) smaller than those from the RWMH algorithm.

Insert Table 1 about here.

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 (3). The MDD also appears as constant Z in (18). 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). Since this only requires draws from the posterior, we also use it on the SMC algorithm. The estimates in Table 1 are all extremely close, consistent with the posterior results. While the numerical standard error is bigger for the SMC estimate, it still indicates that the MDD is estimated with reasonable precision.

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. And so, it is no surprise that the RWMH performs well on the standard SW model. Yet the SMC algorithm does not rely quite so heavily on a unimodal posterior, as shown in Section 5.1.

Insert Figure 4 about here.

With a view toward comparing the effectiveness of the different posterior simulators, we relax the priors on the SW model substantially. More specifically, parameters which have previously had beta distributions now have uniform distributions. The other parameters have had their variances scaled up by (roughly) a factor of three, except that the standard deviations of the shocks have their priors unchanged. In general, the priors are much less informative than the standard SW priors, although importantly they are still proper. Figure 4 shows the difference in the priors of the two models for four selected parameters. Most noticeably, parameters restricted to the unit interval now have a uniform prior over that region. A table with the full specification of the prior is available in the online Appendix.

Insert Table 2 about here.

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 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). In general, the Monte Carlo variance across simulations is substantially lower under the SMC algorithm. In particular, the parameters associated with the wage and price markup processes seem to behave quite differently across the two algorithms. Under the SMC algorithm, the posterior 90% credible set associated with ρ_w, the persistence of the wage markup, is $[0.18, 0.99]$ while it is much smaller $[0.62, 0.92]$ for the RWMH algorithm. Moreover, the Monte Carlo variance is quite large for the RWMH algorithm. The standard deviation of the mean of ρ_w across the five runs of RWMH is 0.23, yielding serious doubts about convergence of the algorithm. On the other hand, the Monte Carlo variance associated with the SMC runs suggests that the algorithm has reliably converged.

Insert Figure 5 about here.

A visual comparison of the RWMH and SMC algorithm under the different priors is provided in Figure 5. For eight selected parameters, Figure 5 presents the estimates of the mean, 5th, and 95th percentile 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 red, while the SMC estimates are in black. Under the standard prior, the behavior of the algorithms is roughly the same and reasonably precise. Under the diffuse prior, stark differences between the algorithms emerge. First, the precision of the estimates is much lower for the RWMH simulations than for the SMC simulations, as the size of the red boxes is generally much greater than the size of the black boxes. In fact, for 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 ι_w. Moreover, it is clear that some of the worst performing aspects of the RWMH algorithm are associated with the parameters controlling the movements of wages and prices.

Insert Figure 6 about here.

To shed light on this discrepancy, we plot density estimates from each run for the following parameters associated with the dynamics of wage and price markups: ρ_w, μ_w, ρ_p, and μ_p. Figure 6 displays the densities. There are five estimated densities for each algorithm, each associated with a different run of the algorithm. The high Monte Carlo variance of the RWMH algorithm is reflected in the wide variation of the black lines. It is clear that the results from the RWMH do not reflect convergence to an ergodic distribution. On the other hand, the SMC posteriors look nearly identical, almost appearing as one solid red line for each of the four parameters. Upon closer inspection, it is clear why this is the case. Both the wage markup autoregressive and moving average coefficients have a distinct bimodal structure. There is a mode near one with a sharp peak, while there is second mode in the middle of the long left tail of the distribution. This feature of the posterior is highly irregular, and one key reason that the RWMH algorithm has severe difficulty in approximating the posterior.

Insert Table 3 about here.

Insert Figure 7 about here.

To further explore the multi-modal shape of the posterior under the diffuse prior, Table 3 lists the values of the key wage and price parameters at the two modes, as suggested by the SMC algorithm. 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 mark-up 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. The exogenous wage markup process is much less persistent. Figure 7 shows a contour map of the joint posterior of the autoregressive coefficient for the exogenous wage markup, ρ_w, and the wage Calvo parameter, ξ_w. Here the bimodal structure is visually apparent. At Mode 1, ρ_w is extremely close to one and the wage Calvo parameter, ξ_w, is around 0.85. At Mode 2, the wage Calvo parameter is close to one, while the posterior

for ρ_w is much flatter, indicating that as the endogenous price stickiness goes to one, the exogenous markup is more or less unidentified. Using the SMC algorithm to facilitate an otherwise impractical exercise – estimating the SW model under loose priors – 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).

Insert Table 4 about here.

A final measure of the effectiveness of the algorithms in provided in Table 4. The SMC algorithm provides roughly the same estimate for the MDD using both the modified harmonic mean estimate and the natural estimate out of the sequential algorithm. However, the SMC estimate is much more precise, which is unsurprising, given the difficulties of using the modified harmonic mean estimator on a bimodal distribution. On the other hand, the RWMH gives a different point estimate with a much higher standard deviation, echoing the above concerns about convergence.

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 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 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 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, $\sigma^{x,h}$, for horizons $h = 0, 4, 8$. Given the model's seven exogenous processes, there are 21 total shocks, plus a measurement error for output. Second, households in the SGU model have preferences that were proposed by Jaimovich and Rebelo (2009), wherein a parameter, γ, controls the wealth elasticity of the labor supply. 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 $[0, 1]$. Given what we have seen happening to the SW posterior when we relaxed the prior, a diffuse prior here might yield an irregular posterior surface.

Tuning of Algorithms. We run the same comparison as in previous sections. For each posterior simulator, we compute five runs of the algorithm. For the SMC runs, we use $30,048$ particles, 500 stages, 6 blocks, and $\lambda = 2.1$. We choose the hyperparameters in a way similar to that for the SW model. We use many more particles because the parameter space is much bigger in terms of prior support with meaningful weight. Given a large parameter vector, we increase the number of blocks to ensure that mutation still can occur. For the RWMH, we simulate chains of length $1,000,000$, considering only the final $500,000$ draws. For the proposal density we use a covariance matrix computed from an earlier run, scaled so that the acceptance rate was roughly 27%. 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.

Insert Table 5 about here.

Results from the SGU Prior. Table 5 displays the posteriors under the various simulators. While the results are broadly similar, some important differences emerge. First, the Monte Carlo variance for the estimates of the mean is substantially reduced. One reason for this is that the RWMH moves quite slowly across the parameter space. One way to see this is by looking at the integrated autocorrelation for the chain. For a given parameter, θ_i, the inefficiency factor, κ_i, is a measure of the inefficiency of the Markov chain relative to independent draws from the posterior.

$$\kappa_i = 1 + 2\sum_{l=1}^{\infty} corr(\theta_{i,t}, \theta_{i,t-l}).$$

We approximate it by

$$\hat{\kappa}_i = 1 + 2\sum_{l=1}^{L} \left(1 - \frac{l}{L}\right) \hat{corr}(\theta_{i,t}, \theta_{i,t-1}),$$

where $\hat{corr}(\cdot)$ is the sample autocorrelation of the parameter draws and we choose the number of included lags to be $L = 5,000$. We apply this to each parameter for a single run of the RWMH.[6] Using the last 500,000 draws of the chain, we find that the median (across parameters) inefficiency factor is $1,943.9$. This means that, from our chain of 500,000, we have about 257 effectively independent draws from the posterior. Using the maximum inefficiency factor, $4,138.1$, we have only 120 draws.[7] Even if both algorithms reach the same posterior, estimates from the RWMH algorithm will be less precise given the huge discrepancy in the effective sample size. Recall that the SMC algorithm uses over $30,000$ particles. While resampling introduces a dependence, recall that we resample once the effective sample size is less than $N_{sim}/2$, which, very roughly speaking, puts a lower bound of about $15,000$ on the ESS. This number dwarfs the number of draws from the RWMH corrected by the inefficiency factors κ_i and translates into more precise estimates, which is borne out by Table 5.

[6] Using the other chains yields very similar results.

[7] Interestingly, the parameter with by far the lowest inefficiency factor is γ, the Jaimovich-Rebelo parameter. Given that the RWMH misses the small second mode, this lends caution to leaning too heavily on looking at inefficiency factors to assess algorithmic performance.

Insert Figure 8 about here.

Moreover, it is clear from Figure 8, which displays the parameters associated with the wage markup process, that the RWMH has the most trouble with the multimodalities and irregularities associated with the variances of the news shocks. For instance, σ_μ^4, the size of the four-quarters-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 an identification problem. When ρ_μ is very close to one, it is difficult to distinguish between the four-quarters-ahead news from the eight-quarters-ahead, as both have nearly permanent effects. This is borne out by the density of σ_μ^8, which is nearly a mirror image of σ_μ^4.[8] Indeed, in the posterior they have a correlation of -0.9.

Insert Figure 9 about here.

Figure 9 displays contours for a nonparametric estimate of the joint posterior of σ_μ^4 and σ_μ^8, conditional on $\gamma < 0.01$. The negative relationship is clear. Moreover, the joint posterior displays the clear bimodal structure, which is also evident from the marginals in Figure 8. The RWMH fails to hit this second mode on two of five runs and generally cannot mix on both in reliable proportions. This is reflected in the large differences of the fifth percentile of the various posteriors. 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 the 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.

[8] To examine whether the bi-modality 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.

Insert Figure 10 about here.

With respect to the Jaimovich-Rebelo parameter, γ, the posteriors generated by the RWMH and the SMC algorithm yield small but interesting differences. Figure 10 displays the posteriors for γ. The RWMH algorithm finds basically a degeneracy very close to zero. The SMC algorithm also finds this degeneracy, but it also finds a small mode, comprising about 4% of the total posterior around $\gamma = 0.6$. On this mode, the wealth effect associated with an increase in income is nonzero, which creates some problems for news models, in the sense that positive news about the future can decrease labor supplied today, as consumers anticipate higher income in the future. On this second mode, the importance of anticipated shocks about wages is substantially diminished. Instead, the persistence of the process comes very close to a unit root, and the mean of the unanticipated shock increases from about 1 to 4. At this mode, then, movements in hours are not principally driven by news. Indeed, a long-run variance decomposition suggests that unanticipated movements in the markup account for about half the movement in hours, compared with just 10% of the movement when $\gamma \approx 0$.

In this case, inference based on the central tendency of the distribution gives roughly correct results. This is because the RWMH performs poorly in 1) the tails of the distribution (as with γ) or in regions where its errors offset (mixing poorly on σ_μ^4 and σ_μ^8). The SMC algorithm provides a much better characterization of the distribution. It is clear that for models of similar scale, the RWMH is inadequate in certain dimensions.

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 pdf 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.

Insert Figure 11 about here.

Mechanically, this prior causes the small second mode 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, with the heights of the two modes approximately equal. Figure 11 displays the marginal posterior densities for γ under the new prior for five runs each of the RWMH (black) and the SMC (red) algorithms. The bimodality clearly poses a problem for the RWMH. Only one of five runs even mixes on the second larger mode. The reason for this is that the peak around $\gamma \approx 0$ is extremely sharp and the valley around it 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.

Insert Figure 12 about here.

To see how this translates into problems for inference, Figure 12 plots the share of the long-run variance in hours accounted for by anticipated shocks under both priors. Under the standard prior, both simulators generate roughly the same posteriors, although the SMC has a slightly longer left tail due to the small second mode. Once the prior for γ is changed to a $Beta(2, 1)$, the results from the simulators diverge. All but one of the RWMH runs attribute about 70% to 90% of the variance in hours to anticipated shocks. This is because the simulator gets stuck on the $\gamma \approx 0$ mode. On the other hand, the output of the SMC algorithm ascribes most of the movement in hours to unanticipated shocks.

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 the 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

An, S., and F. Schorfheide (2007): "Bayesian Analysis of DSGE Models," *Econometric Reviews*, 26(2-4), 113–172.

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. (2004): "A Sequential Particle Filter for Static Models," *Biometrika*, 89(3), 539–551.

Christiano, L. J., M. Eichenbaum, and C. L. Evans (2005): "Nominal Rigidities and the Dynamic Effects of a Shock to Monetary Policy," *Journal of Political Economy*, 113(1), 1–45.

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*.

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.

——— (2011): "Bayesian Macroeconometrics," in *Handbook of Bayesian Econometrics*, ed. by H. van Dijk, G. Koop, and J. Geweke, pp. 293–389. Oxford University Press.

——— (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.

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*.

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.

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.

Robert, C. P., and G. Casella (2004): *Monte Carlo Statistical Methods*. Springer.

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.

Tables

Table 1: SW Model with Standard Prior: Log MDD Estimates for Smets-Wouters Model, Standard Prior

Algorithm, Method	MDD Estimate	Standard Deviation
RWMH, Harmonic Mean	-901.97	0.04
SMC, Harmonic Mean	-901.86	0.06
SMC, Particle Estimate	-901.68	0.19

Notes: The RWMH algorithm uses 500,000 draws with the first 250,000 discarded. The average acceptance rate was 30%. The SMC algorithms use 12,000 particles, 500 stages, $\lambda = 2.1$, a mixture proposal and 3 blocks in each Metropolis-Hastings step.

Table 2: SW Model with Diffuse Prior: Posterior Comparison

	RWMH			SMC		
Parameter	Mean	$[0.05, 0.95]$	STD(Mean)	Mean	$[0.05, 0.95]$	STD(Mean)
ϕ	7.54	[4.31, 11.25]	0.30	7.27	[4.03, 11.02]	0.12
μ_w	0.74	[0.54, 0.90]	0.26	0.60	[0.08, 0.98]	0.04
l	-0.06	[-2.34, 2.26]	0.23	-0.07	[-2.33, 2.29]	0.05
ρ_w	0.79	[0.62, 0.92]	0.23	0.66	[0.19, 0.99]	0.04
r_π	2.81	[2.19, 3.55]	0.10	2.77	[2.13, 3.49]	0.03
ι_w	0.71	[0.41, 0.95]	0.09	0.74	[0.39, 0.97]	0.01
μ_p	0.73	[0.45, 0.91]	0.07	0.79	[0.50, 0.98]	0.01
ξ_w	0.91	[0.81, 0.97]	0.05	0.93	[0.80, 0.99]	0.01
σ_l	2.81	[1.35, 4.66]	0.05	2.84	[1.37, 4.68]	0.03
π	0.86	[0.43, 1.23]	0.05	0.86	[0.42, 1.23]	0.01
ξ_p	0.71	[0.60, 0.81]	0.03	0.72	[0.60, 0.82]	0.01

Notes: Means and standard deviations are over 5 runs for each algorithm. The RWMH algorithms use 500,000 draws with the first 250,000 discarded. The average acceptance rate was roughly 28%. The SMC algorithms use 12,000 particles, 500 stages, $\lambda = 2.1$, a mixture proposal and 3 blocks in each Metropolis-Hastings step.

Table 3: SW Model with Diffuse Prior: Two Modes

Parameter	Mode 1	Mode 2
ξ_w	0.834	0.968
ι_w	0.842	0.897
ρ_w	0.980	0.533
μ_w	0.958	0.482
σ_w	0.262	0.256
Log Posterior	-802.70	-802.39

Table 4: SW Model with Diffuse Prior: Log MDD Estimates

Algorithm, Method	MDD Estimate	Standard Deviation
RWMH, Harmonic Mean	-875.93	1.49
SMC, Harmonic Mean	-872.63	0.47
SMC, Particle Estimate	-872.39	0.18

Notes: Means and standard deviations are over 5 runs for each algorithm. The RWMH algorithm uses 500,000 draws with the first 250,000 discarded. The average acceptance rate was 30%. The SMC algorithms use 12,000 particles, 500 stages, $\lambda = 2.1$, a mixture proposal and 3 blocks in each Metropolis-Hastings step.

Table 5: Posterior Comparison for News Model

	RWMH			SMC		
Parameter	Mean	[0.05, 0.95]	STD(Mean)	Mean	[0.05, 0.95]	STD(Mean)
σ_{μ}^{4}	4.40	[1.04, 5.93]	0.46	4.27	[0.27, 5.91]	0.18
σ_{μ}^{8}	1.29	[0.04, 4.69]	0.46	1.37	[0.03, 5.17]	0.17
$\sigma_{z^i}^{8}$	5.78	[0.94, 10.60]	0.29	5.66	[0.80, 10.66]	0.11
$\sigma_{z^i}^{4}$	3.14	[0.22, 7.98]	0.26	3.15	[0.21, 8.01]	0.03
σ_{ζ}^{0}	3.87	[0.59, 6.75]	0.22	3.80	[0.48, 6.76]	0.06
σ_{ζ}^{8}	2.61	[0.18, 6.02]	0.20	2.67	[0.17, 6.23]	0.07
κ	9.32	[7.48, 11.43]	0.09	9.33	[7.47, 11.35]	0.08
$\sigma_{z^i}^{0}$	12.36	[9.06, 16.16]	0.09	12.29	[9.07, 15.90]	0.14
σ_{μ}^{0}	0.91	[0.06, 2.37]	0.05	1.02	[0.06, 2.69]	0.01
σ_{ζ}^{4}	2.44	[0.15, 6.01]	0.05	2.43	[0.15, 5.97]	0.11
θ	4.12	[3.20, 5.18]	0.04	4.13	[3.19, 5.19]	0.02
σ_{g}^{0}	0.61	[0.07, 1.07]	0.03	0.62	[0.06, 1.08]	0.01
σ_{g}^{8}	0.41	[0.03, 0.96]	0.03	0.40	[0.03, 0.98]	0.01
ρ_{x_g}	0.64	[0.37, 0.83]	0.01	0.64	[0.38, 0.83]	0.00
ρ_{z^i}	0.42	[0.21, 0.63]	0.01	0.43	[0.21, 0.63]	0.00
$\sigma_{\mu_a}^{0}$	0.21	[0.02, 0.35]	0.01	0.21	[0.02, 0.35]	0.01
$\sigma_{\mu_a}^{4}$	0.16	[0.01, 0.34]	0.01	0.16	[0.01, 0.34]	0.00
$\sigma_{\mu_a}^{8}$	0.15	[0.01, 0.33]	0.01	0.16	[0.01, 0.33]	0.01
$\sigma_{\mu_x}^{0}$	0.36	[0.18, 0.52]	0.01	0.36	[0.18, 0.52]	0.00
$\sigma_{\mu_x}^{4}$	0.10	[0.01, 0.26]	0.01	0.10	[0.01, 0.27]	0.00
$\sigma_{\mu_x}^{8}$	0.12	[0.01, 0.28]	0.01	0.12	[0.01, 0.29]	0.00
σ_{z}^{0}	0.66	[0.56, 0.74]	0.01	0.66	[0.55, 0.74]	0.00
σ_{z}^{4}	0.13	[0.01, 0.31]	0.01	0.13	[0.01, 0.32]	0.00
σ_{g}^{4}	0.57	[0.04, 1.06]	0.01	0.56	[0.04, 1.06]	0.02
γ	0.00	[0.00, 0.00]	0.00	0.01	[0.00, 0.01]	0.00
δ_2/δ_1	0.42	[0.31, 0.55]	0.00	0.42	[0.31, 0.56]	0.00
b	0.92	[0.89, 0.94]	0.00	0.92	[0.89, 0.94]	0.00
ρ_z	0.91	[0.85, 0.96]	0.00	0.91	[0.84, 0.96]	0.00
ρ_a	0.48	[0.39, 0.57]	0.00	0.48	[0.38, 0.57]	0.00
ρ_g	0.96	[0.92, 0.99]	0.00	0.96	[0.92, 0.99]	0.00
ρ_x	0.87	[0.68, 0.99]	0.00	0.87	[0.67, 0.99]	0.00
ρ_μ	0.98	[0.95, 1.00]	0.00	0.98	[0.95, 1.00]	0.00
ρ_ζ	0.18	[0.07, 0.31]	0.00	0.19	[0.08, 0.32]	0.00
σ_{z}^{8}	0.12	[0.01, 0.29]	0.00	0.12	[0.01, 0.29]	0.00
σ_{ygr}^{me}	0.30	[0.30, 0.30]	0.00	0.30	[0.30, 0.30]	0.00

Notes: Means and standard deviations are over 5 runs for each algorithm. The RWMH algorithms use 1,000,000 draws with the first 500,000 discarded. The average acceptance rate was roughly 27%. The SMC algorithms use 30,048 particles, 500 stages, $\lambda = 2.1$, a mixture proposal and 6 blocks in each Metropolis-Hastings step.

Table 6: Log MDD Estimates for New Models

Algorithm, Method	MDD Estimate	Standard Deviation
RWMH, Harmonic Mean	-1800.9	0.32
SMC, Harmonic Mean	-1798.0	1.54
SMC, Particle Estimate	-1802.3	0.23

Notes: Means and standard deviations are over 5 runs for each algorithm. The RWMH algorithms use 1,000,000 draws with the first 500,000 discarded. The average acceptance rate was roughly 27%. The SMC algorithms use 30,048 particles, 500 stages, $\lambda = 2.1$, a mixture proposal and 6 blocks in each Metropolis-Hastings step.

Figures

Figure 1: Stylized State-Space Model: Log Likelihood Function

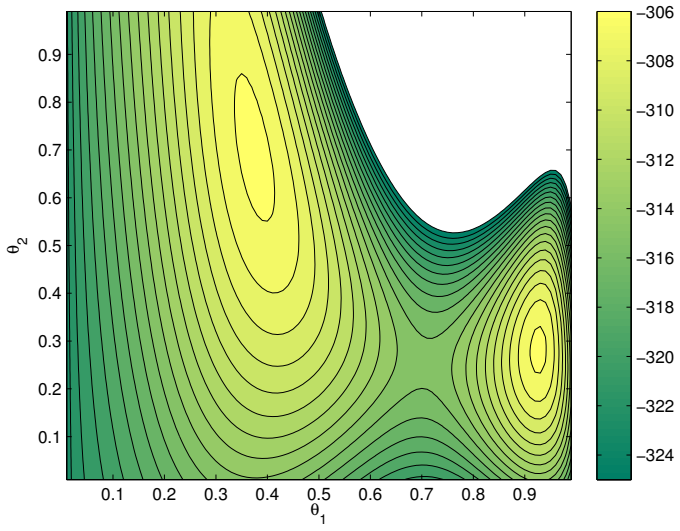


Figure 2: Stylized State-Space Model: Log Likelihood Function and Posterior Draws

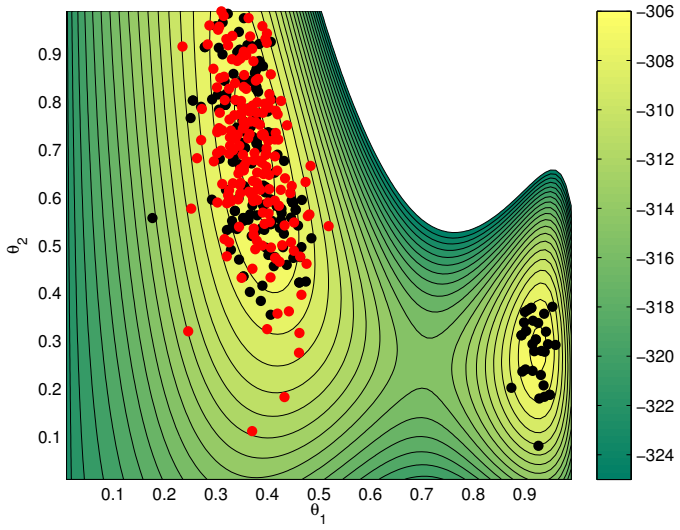


Figure 3: Stylized State-Space Model: Density Estimates for θ_1 as Function of n

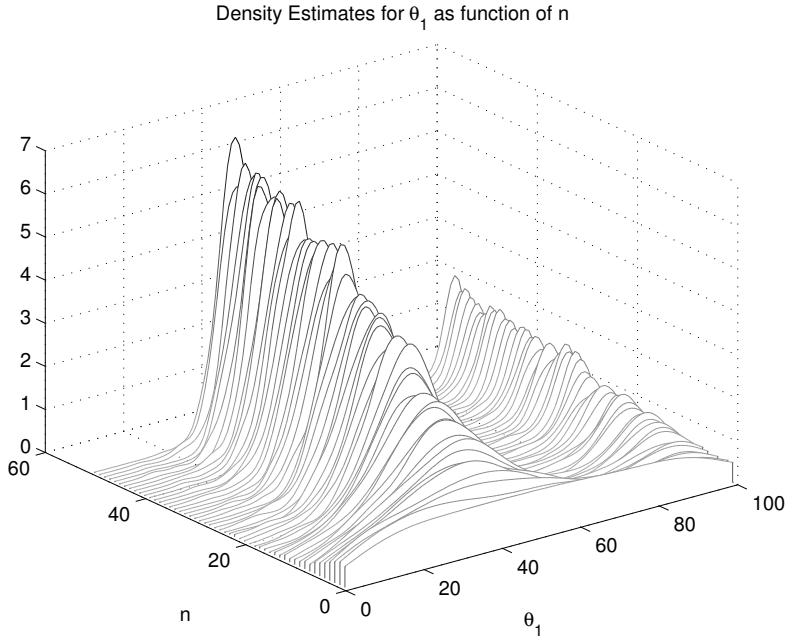


Figure 4: SW Model: Standard Priors

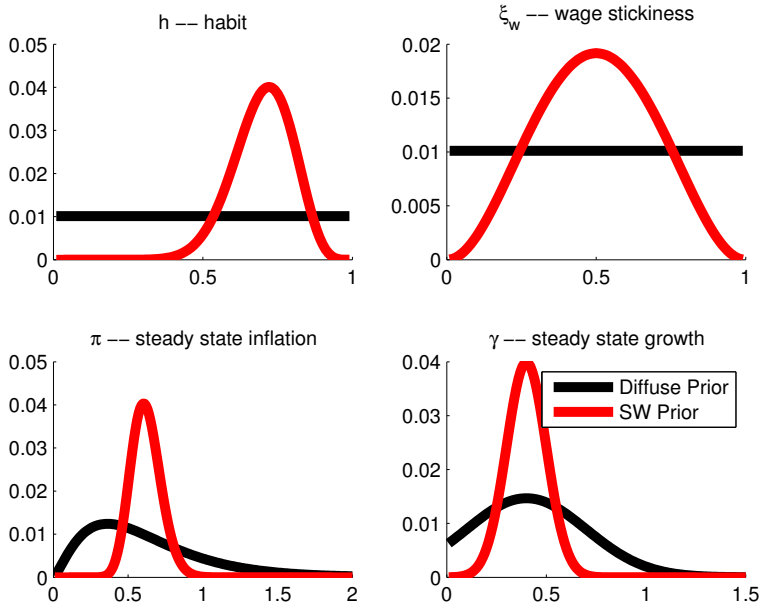
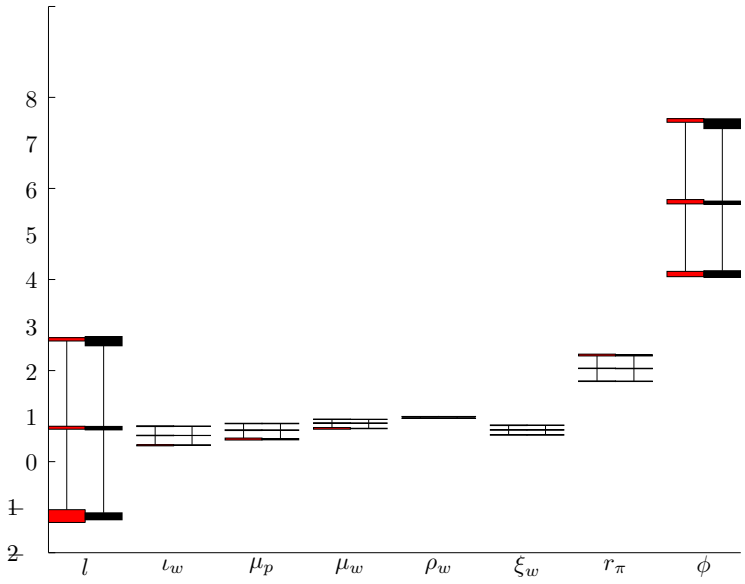
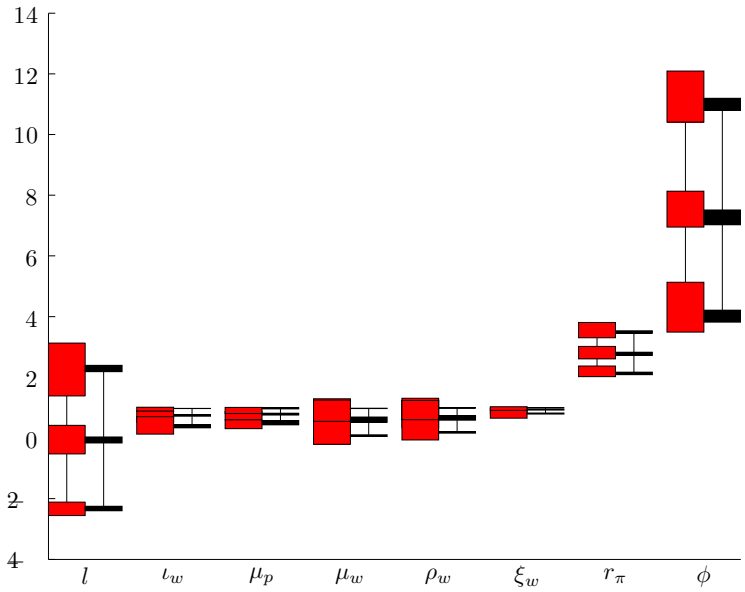


Figure 5: SW Model: Estimates of Mean, 5th and 95th Percentile for Selected Parameters

Standard Prior

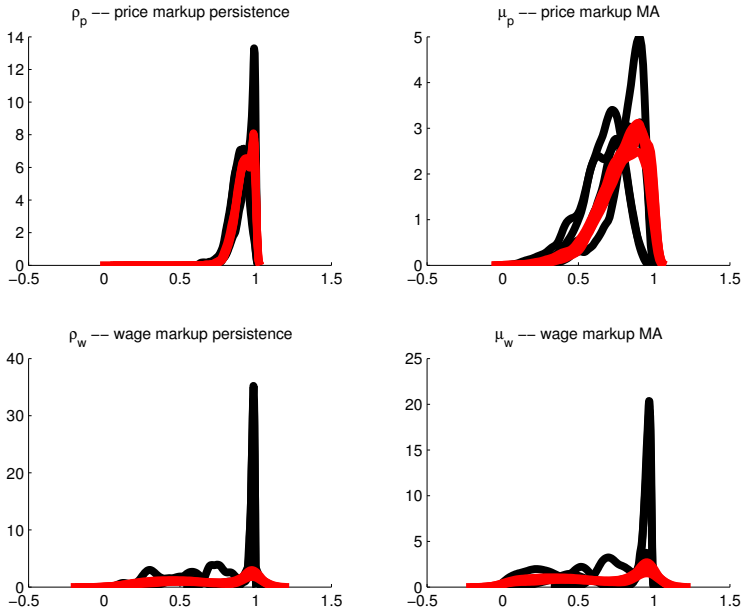


Diffuse Prior



Notes: RWMH (Red) and SMC(Black). Shaded region shows +/- two standard deviations around point estimate.

Figure 6: SW Model with Diffuse Prior: RWMH and SMC Posteriors for Markup Parameters



Notes: RWMH (Black) and SMC (Red). RWMH: 5 runs, 500,000 draws per run; SMC: 5 runs, 12,000 particles, 500 stages, 6 blocks.

Figure 7: SW Model with Diffuse Prior: Joint Posterior for Wage Markup Persistence and Wage Calvo Parameter

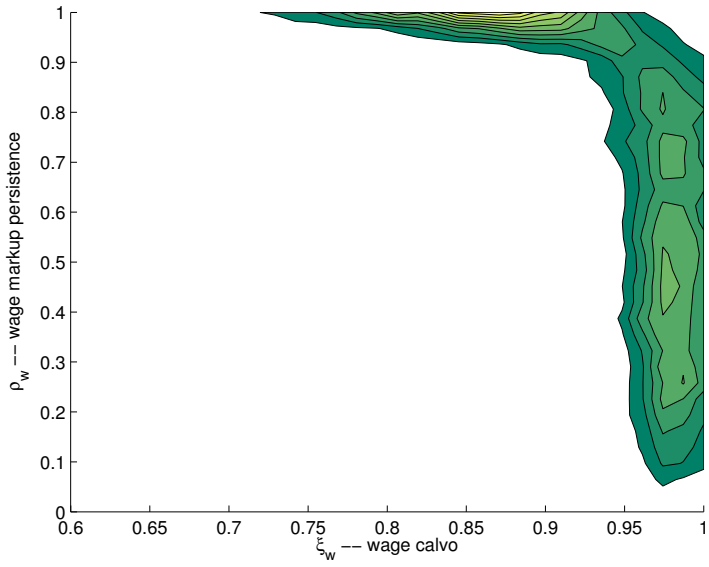
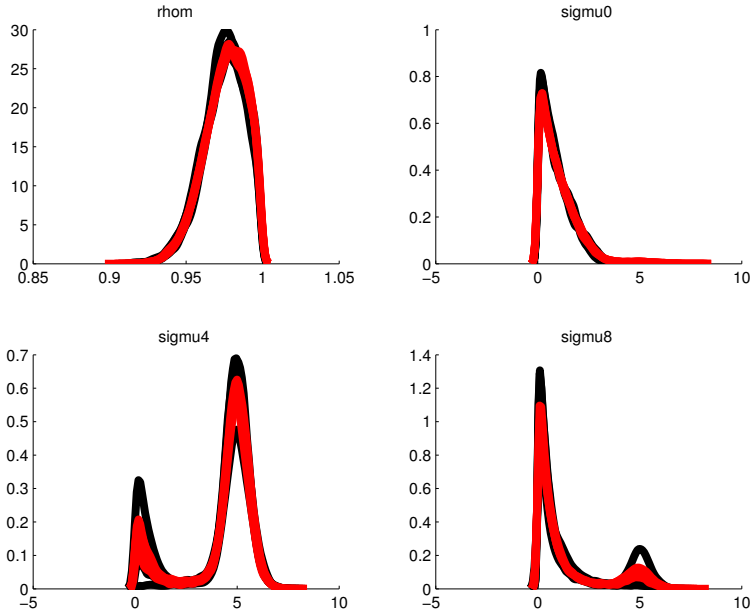


Figure 8: News Model: RWMH and SMC Posteriors of Wage Markup Process



Notes: RWMH (Black) and SMC (Red). RWMH: 5 runs, 1,000,000 draws per run; SMC: 5 runs, 30,028 particles, 500 stages, 6 blocks.

Figure 9: News Model: Joint Posterior for Variances of Anticipated Shocks to Wage Markup

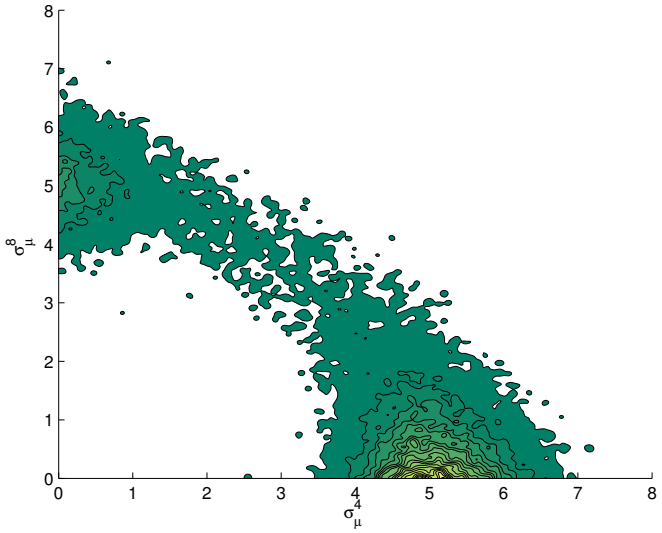
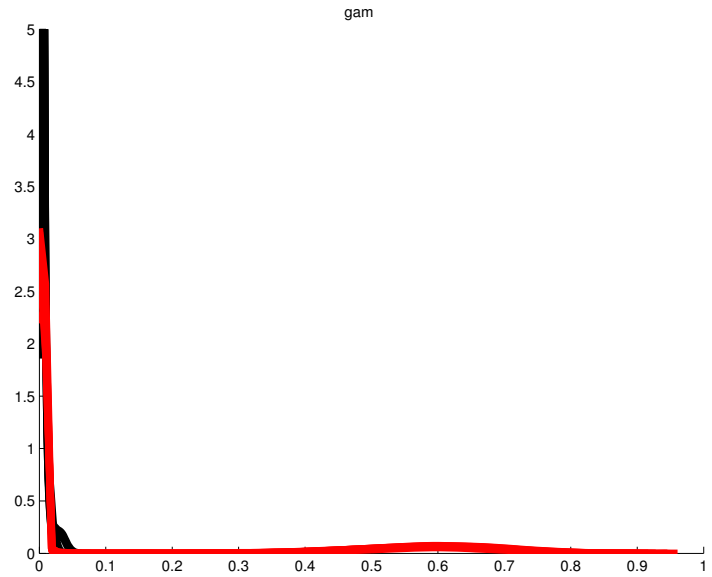
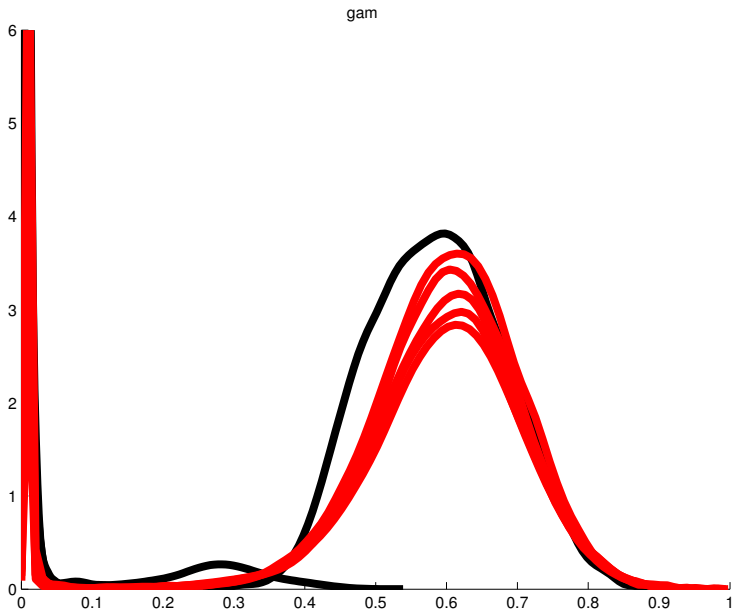


Figure 10: News Model: RWMH and SMC Posteriors of γ



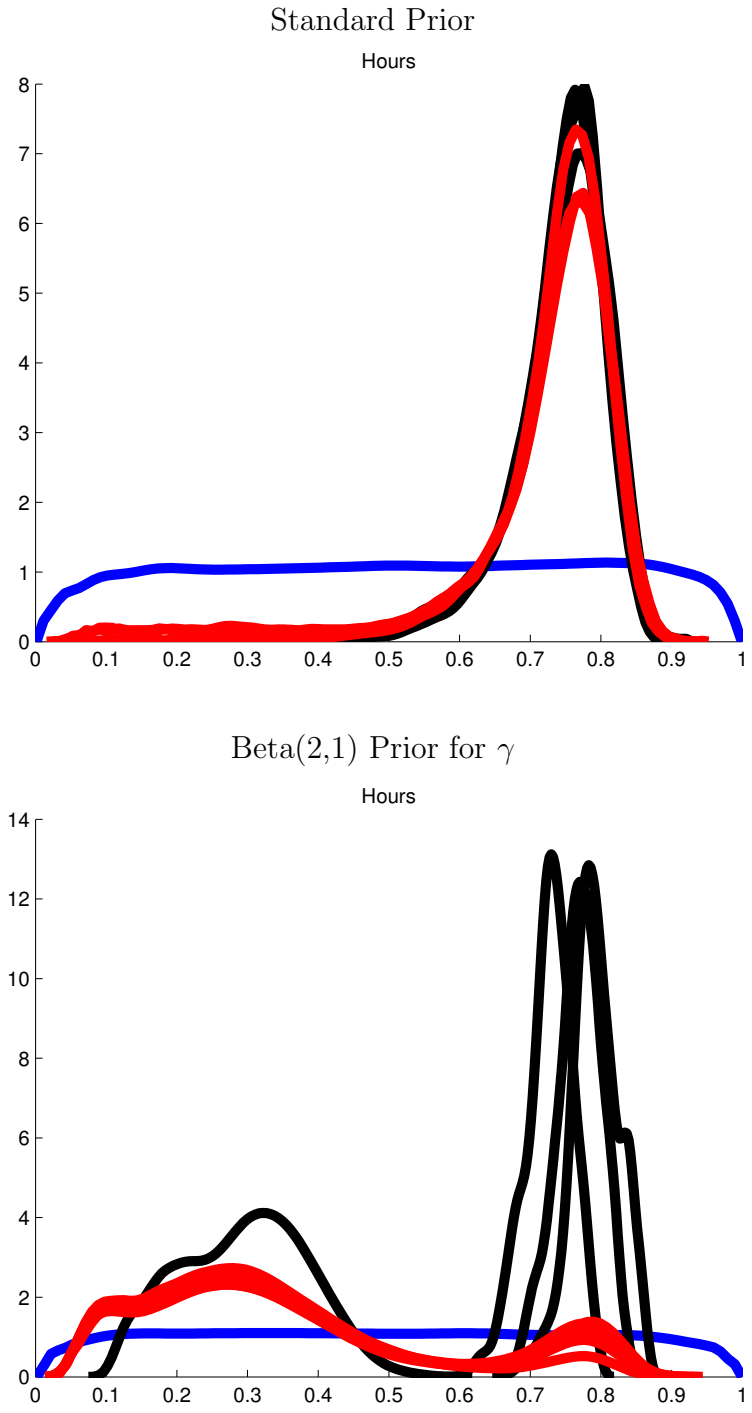
Notes: RWMH (Black) and SMC (Red). RWMH: 5 runs, 1,000,000 draws per run; SMC: 5 runs, 30,028 particles, 500 stages, 6 blocks.

Figure 11: News Model: RWMH and SMC Posteriors of γ



Notes: RWMH (Black) and SMC (Red). RWMH: 5 runs, 1,000,000 draws per run; SMC: 5 runs, 30,028 particles, 500 stages, 6 blocks

Figure 12: News Model: Anticipated Shocks' Variance Shares for Hours



Notes: Prior (Blue), RWMH (Black), and SMC (Red). RWMH: 5 runs, 1,000,000 draws per run; SMC: 5 runs, 30028 particles, 500 stages, 6 blocks.

Online Appendix for *Sequential Monte Carlo Sampling for DSGE Models*

Edward Herbst and Frank Schorfheide

A The Smets-Wouters Model

The equilibrium conditions of the Smets and Wouters (2007) model take the following form:

$$\hat{y}_t = c_y\hat{c}_t + i_y\hat{i}_t + z_y\hat{z}_t + \varepsilon_t^g \tag{A-1}$$

$$\hat{c}_t = \frac{h/\gamma}{1+h/\gamma}\hat{c}_{t-1} + \frac{1}{1+h/\gamma}E_t\hat{c}_{t+1} + \frac{wl_c(\sigma_c-1)}{\sigma_c(1+h/\gamma)}(\hat{l}_t - E_t\hat{l}_{t+1}) - \frac{1-h/\gamma}{(1+h/\gamma)\sigma_c}(\hat{r}_t - E_t\hat{\pi}_{t+1}) - \frac{1-h/\gamma}{(1+h/\gamma)\sigma_c}\varepsilon_t^b \tag{A-2}$$

$$\hat{i}_t = \frac{1}{1+\beta\gamma^{(1-\sigma_c)}}\hat{i}_{t-1} + \frac{\beta\gamma^{(1-\sigma_c)}}{1+\beta\gamma^{(1-\sigma_c)}}E_t\hat{i}_{t+1} + \frac{1}{\varphi\gamma^2(1+\beta\gamma^{(1-\sigma_c)})}\hat{q}_t + \varepsilon_t^i \tag{A-3}$$

$$\hat{q}_t = \beta(1-\delta)\gamma^{-\sigma_c}E_t\hat{q}_{t+1} - \hat{r}_t + E_t\hat{\pi}_{t+1} + (1-\beta(1-\delta)\gamma^{-\sigma_c})E_t\hat{r}_{t+1}^k - \varepsilon_t^b \tag{A-4}$$

$$\hat{y}_t = \Phi(\alpha\hat{k}_t^s + (1-\alpha)\hat{l}_t + \varepsilon_t^a) \tag{A-5}$$

$$\hat{k}_t^s = \hat{k}_{t-1} + \hat{z}_t \tag{A-6}$$

$$\hat{z}_t = \frac{1-\psi}{\psi}\hat{r}_t^k \tag{A-7}$$

$$\hat{k}_t = \frac{(1-\delta)}{\gamma}\hat{k}_{t-1} + (1-(1-\delta)/\gamma)\hat{i}_t + (1-(1-\delta)/\gamma)\varphi\gamma^2(1+\beta\gamma^{(1-\sigma_c)})\varepsilon_t^i \tag{A-8}$$

$$\hat{\mu}_t^p = \alpha(\hat{k}_t^s - \hat{l}_t) - \hat{w}_t + \varepsilon_t^a \tag{A-9}$$

$$\hat{\pi}_t = \frac{\beta\gamma^{(1-\sigma_c)}}{1+\iota_p\beta\gamma^{(1-\sigma_c)}}E_t\hat{\pi}_{t+1} + \frac{\iota_p}{1+\beta\gamma^{(1-\sigma_c)}}\hat{\pi}_{t-1} - \frac{(1-\beta\gamma^{(1-\sigma_c)}\xi_p)(1-\xi_p)}{(1+\iota_p\beta\gamma^{(1-\sigma_c)})(1+(\Phi-1)\varepsilon_p)\xi_p}\hat{\mu}_t^p + \varepsilon_t^p \tag{A-10}$$

$$\hat{r}_t^k = \hat{l}_t + \hat{w}_t - \hat{k}_t^s \tag{A-11}$$

$$\hat{\mu}_t^w = \hat{w}_t - \sigma_l\hat{l}_t - \frac{1}{1-h/\gamma}(\hat{c}_t - h/\gamma\hat{c}_{t-1}) \tag{A-12}$$

$$\begin{aligned}
\hat{w}_t &= \frac{\beta\gamma^{(1-\sigma_c)}}{1+\beta\gamma^{(1-\sigma_c)}}(E_t\hat{w}_{t+1}+E_t\hat{\pi}_{t+1})+\frac{1}{1+\beta\gamma^{(1-\sigma_c)}}(\hat{w}_{t-1}-\iota_w\hat{\pi}_{t-1}) \qquad \text{(A-13)}\\
&\quad -\frac{1+\beta\gamma^{(1-\sigma_c)}\iota_w}{1+\beta\gamma^{(1-\sigma_c)}}\hat{\pi}_t-\frac{(1-\beta\gamma^{(1-\sigma_c)}\xi_w)(1-\xi_w)}{(1+\beta\gamma^{(1-\sigma_c)})(1+(\lambda_w-1)\epsilon_w)\xi_w}\hat{\mu}_t^w+\varepsilon_t^w\\
\hat{r}_t &= \rho\hat{r}_{t-1}+(1-\rho)(r_\pi\hat{\pi}_t+r_y(\hat{y}_t-\hat{y}_t^*)) \qquad \text{(A-14)}\\
&\quad +r_{\Delta y}((\hat{y}_t-\hat{y}_t^*)-(\hat{y}_{t-1}-\hat{y}_{t-1}^*))+\varepsilon_t^r.
\end{aligned}$$

The exogenous shocks evolve according to

$$\begin{aligned}
\varepsilon_t^a &= \rho_a\varepsilon_{t-1}^a+\eta_t^a \qquad \text{(A-15)}\\
\varepsilon_t^b &= \rho_b\varepsilon_{t-1}^b+\eta_t^b \qquad \text{(A-16)}\\
\varepsilon_t^g &= \rho_g\varepsilon_{t-1}^a+\rho_{ga}\eta_t^a+\eta_t^g \qquad \text{(A-17)}\\
\varepsilon_t^i &= \rho_i\varepsilon_{t-1}^i+\eta_t^i \qquad \text{(A-18)}\\
\varepsilon_t^r &= \rho_r\varepsilon_{t-1}^r+\eta_t^r \qquad \text{(A-19)}\\
\varepsilon_t^p &= \rho_r\varepsilon_{t-1}^p+\eta_t^p-\mu_p\eta_{t-1}^p \qquad \text{(A-20)}\\
\varepsilon_t^w &= \rho_w\varepsilon_{t-1}^w+\eta_t^w-\mu_w\eta_{t-1}^w. \qquad \text{(A-21)}
\end{aligned}$$

The counterfactual no-rigidity prices and quantities evolve according to

$$\begin{aligned}
\hat{y}_t^* &= c_y\hat{c}_t^*+i_y\hat{i}_t^*+z_y\hat{z}_t^*+\varepsilon_t^g \qquad \text{(A-22)}\\
\hat{c}_t^* &= \frac{h/\gamma}{1+h/\gamma}\hat{c}_{t-1}^*+\frac{1}{1+h/\gamma}E_t\hat{c}_{t+1}^*+\frac{wl_c(\sigma_c-1)}{\sigma_c(1+h/\gamma)}(\hat{l}_t^*-E_t\hat{l}_{t+1}^*) \qquad \text{(A-23)}\\
&\quad -\frac{1-h/\gamma}{(1+h/\gamma)\sigma_c}r_t^*-\frac{1-h/\gamma}{(1+h/\gamma)\sigma_c}\varepsilon_t^b\\
\hat{i}_t^* &= \frac{1}{1+\beta\gamma^{(1-\sigma_c)}}\hat{i}_{t-1}^*+\frac{\beta\gamma^{(1-\sigma_c)}}{1+\beta\gamma^{(1-\sigma_c)}}E_t\hat{i}_{t+1}^*+\frac{1}{\varphi\gamma^2(1+\beta\gamma^{(1-\sigma_c)})}\hat{q}_t^*+\varepsilon_t^i \qquad \text{(A-24)}\\
\hat{q}_t^* &= \beta(1-\delta)\gamma^{-\sigma_c}E_t\hat{q}_{t+1}^*-r_t^*+(1-\beta(1-\delta)\gamma^{-\sigma_c})E_tr_{t+1}^{k*}-\varepsilon_t^b \qquad \text{(A-25)}\\
\hat{y}_t^* &= \Phi(\alpha k_t^{s*}+(1-\alpha)\hat{l}_t^*+\varepsilon_t^a) \qquad \text{(A-26)}\\
\hat{k}_t^{s*} &= k_{t-1}^*+z_t^* \qquad \text{(A-27)}
\end{aligned}$$

$$\hat{z}_t^* = \frac{1-\psi}{\psi}\hat{r}_t^{k*} \tag{A-28}$$

$$\hat{k}_t^* = \frac{(1-\delta)}{\gamma}\hat{k}_{t-1}^* + (1-(1-\delta)/\gamma)\hat{i}_t + (1-(1-\delta)/\gamma)\varphi\gamma^2(1+\beta\gamma^{(1-\sigma_c)})\varepsilon_t^i \tag{A-29}$$

$$\hat{w}_t^* = \alpha(\hat{k}_t^{s*} - \hat{l}_t^*) + \varepsilon_t^a \tag{A-30}$$

$$\hat{r}_t^{k*} = \hat{l}_t^* + \hat{w}_t^* - \hat{k}_t^* \tag{A-31}$$

$$\hat{w}_t^* = \sigma_l\hat{l}_t^* + \frac{1}{1-h/\gamma}(\hat{c}_t^* + h/\gamma\hat{c}_{t-1}^*). \tag{A-32}$$

The steady state (ratios) that appear in the measurement equation or the log-linearized equilibrium conditions are given by

$$\gamma = \bar{\gamma}/100 + 1 \tag{A-33}$$

$$\pi^* = \bar{\pi}/100 + 1 \tag{A-34}$$

$$\bar{r} = 100(\beta^{-1}\gamma^{\sigma_c}\pi^* - 1) \tag{A-35}$$

$$r_{ss}^k = \gamma^{\sigma_c}/\beta - (1-\delta) \tag{A-36}$$

$$w_{ss} = \left(\frac{\alpha^\alpha(1-\alpha)^{(1-\alpha)}}{\Phi r_{ss}^{k\ \alpha}}\right)^{\frac{1}{1-\alpha}} \tag{A-37}$$

$$i_k = (1-(1-\delta)/\gamma)\gamma \tag{A-38}$$

$$l_k = \frac{1-\alpha}{\alpha}\frac{r_{ss}^k}{w_{ss}} \tag{A-39}$$

$$k_y = \Phi l_k^{(\alpha-1)} \tag{A-40}$$

$$i_y = (\gamma - 1 + \delta)k_y \tag{A-41}$$

$$c_y = 1 - g_y - i_y \tag{A-42}$$

$$z_y = r_{ss}^k k_y \tag{A-43}$$

$$wl_c = \frac{1}{\lambda_w}\frac{1-\alpha}{\alpha}\frac{r_{ss}^k k_y}{c_y}. \tag{A-44}$$

The prior distribution for the SW model is summarized in Table A-1.

Table A-1: SW Model: Standard Prior

Parameter	Type	Para (1)	Para (2)	Parameter	Type	Para (1)	Para (2)
ϕ	Normal	4.00	1.50	α	Normal	0.30	0.05
σ_c	Normal	1.50	0.37	ρ_a	Beta	0.50	0.20
h	Beta	0.70	0.10	ρ_b	Beta	0.50	0.20
ξ_w	Beta	0.50	0.10	ρ_g	Beta	0.50	0.20
σ_l	Normal	2.00	0.75	ρ_i	Beta	0.50	0.20
ξ_p	Beta	0.50	0.10	ρ_r	Beta	0.50	0.20
ι_p	Beta	0.50	0.15	ρ_p	Beta	0.50	0.20
ι_w	Beta	0.50	0.15	ρ_w	Beta	0.50	0.20
ψ	Beta	0.50	0.15	μ_p	Beta	0.50	0.20
Φ	Normal	1.25	0.12	μ_w	Beta	0.50	0.20
r_π	Normal	1.50	0.25	ρ_{ga}	Beta	0.50	0.20
ρ	Beta	0.75	0.10	σ_a	Inv. Gamma	0.10	2.00
r_y	Normal	0.12	0.05	σ_b	Inv. Gamma	0.10	2.00
$r_{\Delta y}$	Normal	0.12	0.05	σ_g	Inv. Gamma	0.10	2.00
π	Gamma	0.62	0.10	σ_i	Inv. Gamma	0.10	2.00
β^{-1}	Gamma	0.25	0.10	σ_r	Inv. Gamma	0.10	2.00
l	Normal	0.00	2.00	σ_p	Inv. Gamma	0.10	2.00
γ	Normal	0.40	0.10	σ_w	Inv. Gamma	0.10	2.00

Notes: Para (1) and Para (2) correspond to the mean and standard deviation of the Beta, Gamma, and Normal distributions and to the upper and lower bounds of the support for the Uniform distribution. For the Inv. Gamma distribution, Para (1) and Para (2) refer to s and ν, where $p(\sigma|\nu, s) \propto \sigma^{-\nu-1}e^{-\nu s^2/2\sigma^2}$.

Table A-2 shows the posterior means as well as 90% equal-tail-probability credible intervals. We also report the standard deviation of posterior mean across the five repetitions of the posterior simulation.

Table A-3 shows the diffuse prior for the SW model.

Table A-2: SW Model with Standard Prior: Posterior Comparison

	RWMH			SMC		
Parameter	Mean	[0.05, 0.95]	STD(Mean)	Mean	[0.05, 0.95]	STD(Mean)
ϕ	5.71	[4.12, 7.49]	0.02	5.69	[4.12, 7.43]	0.02
σ_c	1.32	[1.13, 1.54]	0.01	1.32	[1.13, 1.54]	0.00
h	0.72	[0.65, 0.79]	0.00	0.72	[0.65, 0.79]	0.00
λ_w	1.50	[1.50, 1.50]	0.00	1.50	[1.50, 1.50]	0.00
ξ_w	0.70	[0.59, 0.80]	0.00	0.70	[0.59, 0.80]	0.00
σ_l	1.88	[1.01, 2.86]	0.02	1.87	[1.01, 2.85]	0.01
ξ_p	0.64	[0.55, 0.73]	0.00	0.64	[0.54, 0.73]	0.00
ι_p	0.58	[0.36, 0.78]	0.00	0.57	[0.36, 0.78]	0.00
ι_w	0.25	[0.12, 0.41]	0.00	0.25	[0.12, 0.41]	0.00
ψ	0.55	[0.37, 0.74]	0.00	0.55	[0.37, 0.74]	0.00
Φ	1.58	[1.46, 1.71]	0.00	1.58	[1.46, 1.71]	0.00
r_π	2.05	[1.77, 2.34]	0.00	2.04	[1.77, 2.34]	0.00
ρ	0.80	[0.76, 0.84]	0.00	0.80	[0.76, 0.84]	0.00
r_y	0.09	[0.05, 0.13]	0.00	0.09	[0.05, 0.12]	0.00
$r_{\Delta y}$	0.22	[0.18, 0.27]	0.00	0.22	[0.18, 0.27]	0.00
π	0.69	[0.52, 0.87]	0.00	0.69	[0.52, 0.87]	0.00
β^{-1}	0.17	[0.08, 0.27]	0.00	0.17	[0.08, 0.27]	0.00
l	0.74	[-1.20, 2.69]	0.02	0.74	[-1.20, 2.65]	0.02
γ	0.42	[0.39, 0.45]	0.00	0.42	[0.39, 0.45]	0.00
α	0.19	[0.16, 0.22]	0.00	0.19	[0.16, 0.22]	0.00
ρ_a	0.96	[0.94, 0.98]	0.00	0.96	[0.94, 0.97]	0.00
ρ_b	0.21	[0.08, 0.37]	0.00	0.21	[0.08, 0.37]	0.00
ρ_g	0.98	[0.96, 0.99]	0.00	0.98	[0.96, 0.99]	0.00
ρ_i	0.73	[0.63, 0.82]	0.00	0.73	[0.63, 0.82]	0.00
ρ_r	0.15	[0.06, 0.26]	0.00	0.15	[0.06, 0.27]	0.00
ρ_p	0.90	[0.80, 0.97]	0.00	0.90	[0.80, 0.96]	0.00
ρ_w	0.97	[0.95, 0.99]	0.00	0.97	[0.95, 0.99]	0.00
μ_p	0.69	[0.49, 0.84]	0.00	0.69	[0.49, 0.84]	0.00
μ_w	0.85	[0.73, 0.93]	0.00	0.84	[0.73, 0.93]	0.00
ρ_{ga}	0.50	[0.35, 0.65]	0.00	0.50	[0.35, 0.65]	0.00
σ_a	0.47	[0.42, 0.52]	0.00	0.47	[0.42, 0.52]	0.00
σ_b	0.24	[0.20, 0.28]	0.00	0.24	[0.20, 0.28]	0.00
σ_g	0.54	[0.49, 0.59]	0.00	0.54	[0.49, 0.59]	0.00
σ_i	0.45	[0.38, 0.54]	0.00	0.45	[0.38, 0.54]	0.00
σ_r	0.25	[0.22, 0.28]	0.00	0.25	[0.22, 0.28]	0.00
σ_p	0.14	[0.11, 0.17]	0.00	0.14	[0.11, 0.17]	0.00
σ_w	0.25	[0.21, 0.29]	0.00	0.25	[0.21, 0.29]	0.00

Notes: Means and standard deviations are over 5 runs for each algorithm. The RWMH algorithms use 500,000 draws with the first 250,000 discarded. The average acceptance rate was roughly 30%. The SMC algorithms use 12,000 particles, 500 stages, $\lambda = 2.1$, a mixture proposal and 3 blocks in each Metropolis-Hastings step.

Table A-3: SW Model: Diffuse Prior

Parameter	Type	Para (1)	Para (2)	Parameter	Type	Para (1)	Para (2)
ϕ	Normal	4.00	3.50	α	Normal	0.30	0.12
σ_c	Normal	1.50	0.90	ρ_a	Uniform	0.00	1.00
h	Uniform	0.00	1.00	ρ_b	Uniform	0.00	1.00
ξ_w	Uniform	0.00	1.00	ρ_g	Uniform	0.00	1.00
σ_l	Normal	2.00	1.75	ρ_i	Uniform	0.00	1.00
ξ_p	Uniform	0.00	1.00	ρ_r	Uniform	0.00	1.00
ι_p	Uniform	0.00	1.00	ρ_p	Uniform	0.00	1.00
ι_w	Uniform	0.00	1.00	ρ_w	Uniform	0.00	1.00
ψ	Uniform	0.00	1.00	μ_p	Uniform	0.00	1.00
Φ	Normal	1.25	0.36	μ_w	Uniform	0.00	1.00
r_π	Normal	1.50	0.75	ρ_{ga}	Uniform	0.00	1.00
ρ	Uniform	0.00	1.00	σ_a	Inv. Gamma	0.10	2.00
r_y	Normal	0.12	0.15	σ_b	Inv. Gamma	0.10	2.00
$r_{\Delta y}$	Normal	0.12	0.15	σ_g	Inv. Gamma	0.10	2.00
π	Gamma	0.62	0.40	σ_i	Inv. Gamma	0.10	2.00
β^1	Gamma	0.25	0.30	σ_r	Inv. Gamma	0.10	2.00
l	Normal	0.00	2.00	σ_p	Inv. Gamma	0.10	2.00
γ	Normal	0.40	0.30	σ_w	Inv. Gamma	0.10	2.00

Notes: Para (1) and Para (2) correspond to the mean and standard deviation of the Beta, Gamma, and Normal distributions and to the upper and lower bounds of the support for the Uniform distribution. For the Inv. Gamma distribution, Para (1) and Para (2) refer to s and ν, where $p(\sigma|\nu, s) \propto \sigma^{-\nu-1}e^{-\nu s^2/2\sigma^2}$.

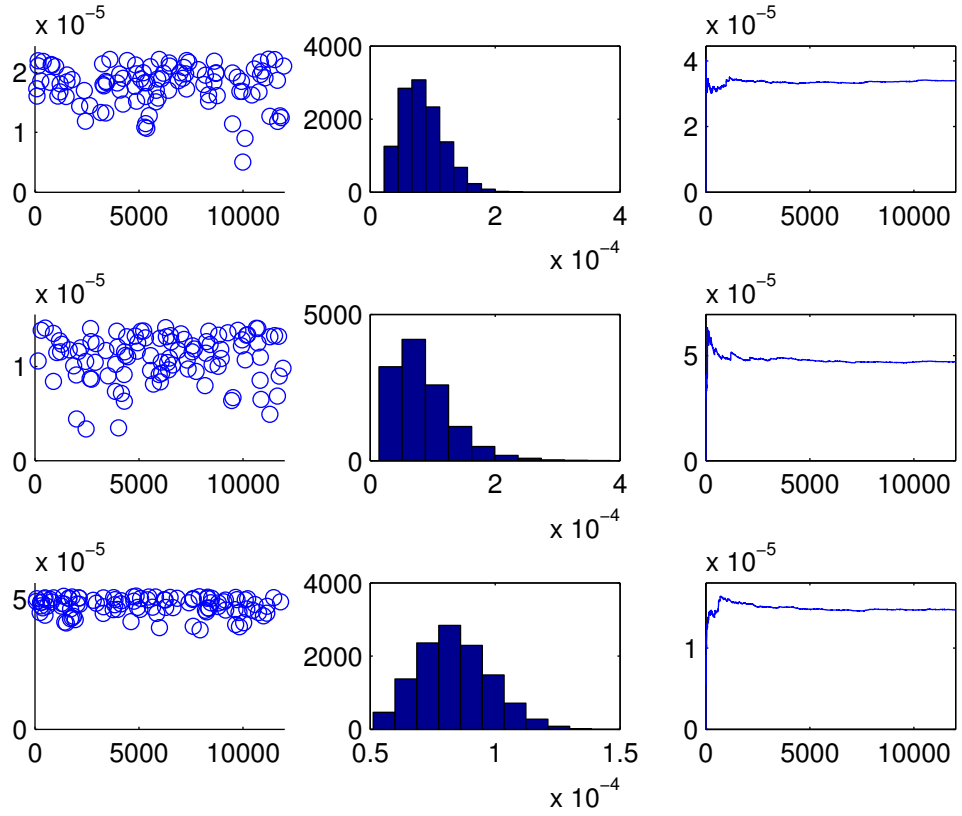
Table A-4: SW Model with Diffuse Prior: Posterior Comparison

	RWMH			SMC		
Parameter	Mean	[0.05, 0.95]	STD(Mean)	Mean	[0.05, 0.95]	STD(Mean)
ϕ	7.54	[4.31, 11.25]	0.30	7.27	[4.03, 11.02]	0.12
μ_w	0.74	[0.54, 0.90]	0.26	0.60	[0.08, 0.98]	0.04
l	-0.06	[-2.34, 2.26]	0.23	-0.07	[-2.33, 2.29]	0.05
ρ_w	0.79	[0.62, 0.92]	0.23	0.66	[0.19, 0.99]	0.04
r_π	2.81	[2.19, 3.55]	0.10	2.77	[2.13, 3.49]	0.03
ι_w	0.71	[0.41, 0.95]	0.09	0.74	[0.39, 0.97]	0.01
μ_p	0.73	[0.45, 0.91]	0.07	0.79	[0.50, 0.98]	0.01
ξ_w	0.91	[0.81, 0.97]	0.05	0.93	[0.80, 0.99]	0.01
σ_l	2.81	[1.35, 4.66]	0.05	2.84	[1.37, 4.68]	0.03
π	0.86	[0.43, 1.23]	0.05	0.86	[0.42, 1.23]	0.01
ξ_p	0.71	[0.60, 0.81]	0.03	0.72	[0.60, 0.82]	0.01
σ_c	1.65	[1.34, 2.06]	0.02	1.66	[1.34, 2.02]	0.01
Φ	1.72	[1.53, 1.93]	0.02	1.73	[1.51, 1.95]	0.01
ρ_p	0.91	[0.81, 0.99]	0.02	0.92	[0.82, 1.00]	0.00
ι_p	0.10	[0.01, 0.27]	0.01	0.11	[0.01, 0.28]	0.00
ψ	0.75	[0.51, 0.96]	0.01	0.76	[0.51, 0.96]	0.01
r_y	0.15	[0.09, 0.24]	0.01	0.15	[0.08, 0.23]	0.00
γ	0.41	[0.37, 0.44]	0.01	0.41	[0.37, 0.44]	0.00
ρ_b	0.19	[0.03, 0.43]	0.01	0.20	[0.03, 0.46]	0.01
ρ_i	0.72	[0.62, 0.83]	0.01	0.74	[0.63, 0.84]	0.00
ρ_{ga}	0.44	[0.26, 0.61]	0.01	0.44	[0.26, 0.61]	0.00
h	0.69	[0.59, 0.78]	0.00	0.69	[0.58, 0.77]	0.00
λ_w	1.50	[1.50, 1.50]	0.00	1.50	[1.50, 1.50]	0.00
ρ	0.88	[0.84, 0.91]	0.00	0.88	[0.84, 0.92]	0.00
$r_{\Delta y}$	0.28	[0.22, 0.35]	0.00	0.28	[0.22, 0.35]	0.00
β^{-1}	0.06	[0.00, 0.18]	0.00	0.06	[0.00, 0.19]	0.00
α	0.17	[0.14, 0.21]	0.00	0.17	[0.14, 0.21]	0.00
ρ_a	0.97	[0.96, 0.98]	0.00	0.97	[0.96, 0.98]	0.00
ρ_g	0.98	[0.97, 1.00]	0.00	0.98	[0.97, 1.00]	0.00
ρ_r	0.06	[0.00, 0.15]	0.00	0.05	[0.00, 0.14]	0.00
σ_a	0.46	[0.41, 0.51]	0.00	0.46	[0.41, 0.51]	0.00
σ_b	0.24	[0.19, 0.28]	0.00	0.24	[0.18, 0.28]	0.00
σ_g	0.55	[0.50, 0.60]	0.00	0.55	[0.50, 0.60]	0.00
σ_i	0.47	[0.39, 0.56]	0.00	0.46	[0.39, 0.55]	0.00
σ_r	0.24	[0.22, 0.27]	0.00	0.24	[0.22, 0.27]	0.00
σ_p	0.13	[0.09, 0.16]	0.00	0.14	[0.09, 0.22]	0.00
σ_w	0.26	[0.22, 0.29]	0.00	0.26	[0.22, 0.30]	0.00

Notes: Means and standard deviations are over 5 runs for each algorithm. The RWMH algorithms use 500,000 draws with the first 250,000 discarded. The average acceptance rate was roughly 28%. The SMC algorithms use 12,000 particles, 500 stages, $\lambda = 2.1$, a mixture proposal and 3 blocks in each Metropolis-Hastings step.

Figure A-1 shows scatter plots of the top-100 particle weights (Column 1); histograms for the remaining weights (Column 2); and recursive estimates of the variance of the weights. The rows correspond to $n = 100$, $n = 200$, and $n = 500$, respectively.

Figure A-1: Visual Diagnostics for SW Model at $n = 100, 200,$ and 500



B Schmitt-Grohé and Uribe (2010) Model

B.1 Steady State

$$\mu_{y_{ss}} = \mu_{a,ss}^{\frac{\alpha_k}{\alpha_k - 1}} \mu_{x,ss} \tag{A-45}$$

$$\mu_{k_{ss}} = \mu_{x,ss} \, \mu_{a,ss}^{\frac{1}{\alpha_k - 1}} \tag{A-46}$$

$$x_{g_{ss}} = \left(\frac{1}{\mu_{y_{ss}}} \right)^{\frac{1}{1-\rho_{xg}}} \tag{A-47}$$

$$\left(\frac{g}{y} \right)_{ss} = \frac{0.20}{x_{g_{ss}}} \tag{A-48}$$

$$\left(\frac{y}{k} \right)_{ss} = \frac{\frac{1}{\mu_{a,ss} \, \beta \, \mu_{y_{ss}}^{(-\sigma)}} - (1 - \delta_0)}{\alpha_k \, \mu_{k_{ss}}} \tag{A-49}$$

$$\left(\frac{i}{k} \right)_{ss} = 1 - \frac{1 - \delta_0}{\mu_{k_{ss}}} \tag{A-50}$$

$$\left(\frac{i}{y} \right)_{ss} = \left(\frac{i}{k} \right)_{ss} / \left(\frac{y}{k} \right)_{ss} \tag{A-51}$$

$$\left(\frac{c}{y} \right)_{ss} = 1 - x_{g_{ss}} \left(\frac{g}{y} \right)_{ss} - \left(\frac{i}{y} \right)_{ss} \tag{A-52}$$

$$\psi = \frac{\frac{\left(1 - \mu_{y_{ss}}^{(-\sigma)} \, \beta \, b\right) \alpha_h}{1 + \mu_{ss}} \left(\frac{y}{c} \right)_{ss}}{h_{ss}^{\theta} \left(\frac{1}{\mu_{y_{ss}}} \right)^{\frac{1-\gamma}{\gamma}} \left(\theta \left(1 - \frac{b}{\mu_{y_{ss}}} \right) + \left(1 - \mu_{y_{ss}}^{(-\sigma)} \, \beta \, b \right) \left(\frac{y}{c} \right)_{ss} \frac{\alpha_h \frac{\gamma}{1 - \beta \, (1-\gamma) \, \mu_{y_{ss}}^{1-\sigma}}}{1 + \mu_{ss}} \right)} \tag{A-53}$$

$$kss = \left(\frac{\frac{y\_k_t}{l^{1-\alpha_k - \alpha_h} \, h_{ss}^{\alpha_h}}}{\mu_{kss}^{(-\alpha_k)}} \right)^{\frac{1}{\alpha_k - 1}} \tag{A-54}$$

$$\delta_1 = \mu_{k_{ss}} \, \alpha_k \, y\_k_t \tag{A-55}$$

B.2 Detrended Equilibrium

B.2.1 Optimality and Market Clearing Conditions

Investment Equation:

$$k_t = \left(1 - \left(\delta_0 + \delta_1 \left(u_t - 1\right) + \frac{\delta_2}{2} \left(u_t - 1\right)^2\right)\right) \frac{k_{t-1}}{\mu_{k_t}} + z_t^i \, i_t \left(1 - \frac{\kappa}{2} \left(\frac{i_t \mu_{k_t}}{i_{t-1}} - \mu_{k_{ss}}\right)^2\right) \tag{A-56}$$

Resource Constraint:

$$y_t = g_t \, x_{g_t} + i_t + c_t \tag{A-57}$$

Production Function:

$$y_t = z_t \left(u_t k_{t-1} / \mu_{k_t}\right)^{\alpha_k} h_t^{\alpha_h} l^{1-\alpha_h-\alpha_k} \tag{A-58}$$

Value of Consumption Bundle:

$$v_t = c_t - b \frac{c_{t-1}}{\mu_{y_t}} - \psi h_t^{\theta} \, s_t \tag{A-59}$$

Geometric Average of past habit-adjusted consumption:

$$s_t = \left(c_t - b \frac{c_{t-1}}{\mu_{y_t}}\right)^{\gamma} \left(\frac{s_{t-1}}{\mu_{y_t}}\right)^{1-\gamma} \tag{A-60}$$

Consumption Decision:

$$\lambda_t = \zeta_t \, v_t^{-\sigma} - \frac{\gamma s_t p_t}{c_t - b \, \frac{c_{t-1}}{\mu_{y_t}}} - \beta \, b \mu_{y_{t+1}}^{-\sigma} \left(\zeta_{t+1} \, v_{t+1}^{-\sigma} - \frac{\gamma s_{t+1} p_{t+1}}{c_{t+1} - b \, \frac{c_t}{\mu_{y_{t+1}}}}\right) \tag{A-61}$$

Hours Decision:

$$\theta \, \psi s_t \, v_t^{-\sigma} \, \zeta_t \, h_t^{\theta-1} = \lambda_t \frac{\alpha_h y_t / h_t}{1 + \mu_t} \tag{A-62}$$

Dynamics for the shadow price of past consumption:

$$p_t = \psi \zeta_t v_t^{-\sigma} h_t^{\theta} + \beta \, (1 - \gamma) \, \mu_{y_{t+1}}^{1-\sigma} \, p_{t+1} \frac{s_{t+1}}{s_t} \tag{A-63}$$

Euler Equation:

$$\lambda_t\, q_t = \beta\, \lambda_{t+1}\, \mu_{a\,t+1} \mu_{y_{t+1}}^{-\sigma} \left(\alpha_k\, u_{t+1} \frac{y_{t+1}}{k_t\, u_{t+1}/\mu_{k\,t+1}} + \left(1 - \left(\delta_0 + \delta_1\, (u_{t+1} - 1) + \frac{\delta_2}{2}\, (u_{t+1} - 1)^2 \right) \right) q_{t+1} \right) \tag{A-64}$$

Capacity Utilization:

$$q_t\, (\delta_1 + \delta_2\, (u_t - 1)) = \alpha_k \frac{y_t}{u_t\, k_{t-1}/\mu_{k\,t}} \tag{A-65}$$

Dynamics of q_t:

$$\begin{aligned} \lambda_t &= q_t \lambda_t\, z_t^i \left(1 - \frac{\kappa}{2} \left(\frac{i_t \mu_{k\,t}}{i_{t-1}} - \mu_{k\,ss} \right)^2 - \kappa\, \frac{i_t \mu_{k\,t}}{i_{t-1}} \left(\frac{i_t \mu_{k\,t}}{i_{t-1}} - \mu_{k\,ss} \right) \right) \\ &+ \beta \mu_{a\,t+1} \mu_{y_{t+1}}^{-\sigma}\, q_{t+1} \lambda_{t+1} z_{t+1}^i \left(\frac{i_{t+1}\, \mu_{k\,t+1}}{i_t} \right)^2 \kappa \left(\frac{i_{t+1}\, \mu_{k\,t+1}}{i_t} - \mu_{k\,ss} \right) \end{aligned} \tag{A-66}$$

B.2.2 Exogenous Processes and Trends

Stochastic trend in output:

$$\mu_{y_t} = \mu_{x\,t}\, \mu_{a\,t}^{\frac{\alpha_k}{\alpha_k - 1}} \tag{A-67}$$

Stochastic trend in capital and investment:

$$\mu_{k\,t} = \mu_{x\,t}\, \mu_{a\,t}^{\frac{1}{\alpha_k - 1}} \tag{A-68}$$

Government Spending Trend:

$$x_{g_t} = \frac{x_{g\,t-1}^{\rho_{xg}}}{\mu_{y_t}} \tag{A-69}$$

Capital-specific technology trend shock:

$$\log \left(\frac{\mu_{a\,t}}{\mu_{a,ss}} \right) = \rho_a\, \log \left(\frac{\mu_{a\,t-1}}{\mu_{a,ss}} \right) + \epsilon_{a,t}^0 + \epsilon_{a,t-4}^4 + \epsilon_{a,t-8}^8 \tag{A-70}$$

Neutral technology trend shock:

$$\log \left(\frac{\mu_{x\,t}}{\mu_{x,ss}} \right) = (\rho_x - 0.5)\, \log \left(\frac{\mu_{x\,t-1}}{\mu_{x,ss}} \right) + \epsilon_{x,t}^0 + \epsilon_{x,t-4}^4 + \epsilon_{x,t-8}^8 \tag{A-71}$$

Neutral technology shock:

$$\log(z_t) = \rho_z \log(z_{t-1}) + \epsilon^0_{z,t} + \epsilon^4_{z,t-4} + \epsilon^8_{z,t-8} \tag{A-72}$$

Investment-specific technology shock:

$$\log\left(z^i_t\right) = \rho_{z^i} \log\left(z^i_{t-1}\right) + \epsilon^0_{z^i,t} + \epsilon^4_{z^i,t-4} + \epsilon^8_{z^i,t-8} \tag{A-73}$$

Government Spending Shock

$$\log\left(\frac{g_t}{gss_t}\right) = \rho_g \log\left(\frac{g_{t-1}}{gss_t}\right) + \epsilon^0_{g,t} + \epsilon^4_{g,t-4} + \epsilon^4_{g,t-8} \tag{A-74}$$

Preference shock:

$$\log(\zeta_t) = \rho_\zeta \log(\zeta_{t-1}) + \epsilon^0_{\zeta,t} + \epsilon^4_{\zeta,t-4} + \epsilon^8_{\zeta,t-8} \tag{A-75}$$

Wage markup:

$$\log\left(\frac{\mu_t}{\mu_{ss}}\right) = \rho_\mu \log\left(\frac{\mu_{t-1}}{\mu_{ss}}\right) + \epsilon^0_{\mu,t} + \epsilon^4_{\mu,t-4} + \epsilon^8_{\mu,t-8} \tag{A-76}$$

B.2.3 Observation Equations

$$ygr_t = 100 \log\left(\frac{y_t \mu_{y_t}}{y_{t-1}}\right) + \epsilon^{me}_{y,t} \tag{A-77}$$

$$cgr_t = 100 \log\left(\frac{c_t \mu_{y_t}}{c_{t-1}}\right) \tag{A-78}$$

$$igr_t = 100 \log\left(\frac{i_t \mu_{a_t} \mu_{k_t}}{i_{t-1}}\right) \tag{A-79}$$

$$hgr_t = 100 \log\left(\frac{h_t}{h_{t-1}}\right) \tag{A-80}$$

$$ggr_t = 100 \log\left(\frac{g_t \, x_{g_t} \mu_{y_t}}{g_{t-1} \, x_{g_{t-1}}}\right) \tag{A-81}$$

$$zgr_t = 100 \log\left(\frac{z_t \mu_{x_t}^{1-\alpha_k}}{z_{t-1}}\right) \tag{A-82}$$

$$agr_t = 100 \log(\mu_{a_t}) \tag{A-83}$$

B.3 Replicating the Results in Schmitt-Grohé and Uribe (2012)

The description of the model in the previous subsections corresponds to the model presented in the text of Schmitt-Grohé and Uribe (2012). Their implementation is slightly different.

Wage Markup: The process for the wage markup shock actually operates on the *gross* markup,

$$1 + \mu.$$

Prior Normalizations: As noted in the text and Table 2 of SGU, the prior for θ is actually on $\theta - 1$. That is,

$$\theta \sim 1 + Gamma(4.00, 1.00).$$

Similarly, support for ρ_x is $[-0.5, 0.5]$, or

$$\rho_x \sim Beta(0.7, 0.2) - 0.5.$$

Finally, as noted in early drafts of SGU, the parameters $[b, \rho_{xg}, \rho_z, \rho_a, \rho_g, \rho_\mu, \rho_\zeta, \rho_{z^i}]$ are rescaled by 0.99, presumably to help identify the news innovations.

Table A-5: News Model: Prior Distribution

Parameter	Type	Para (1)	Para (2)	Parameter	Type	Para (1)	Para (2)
$\theta - 1$	Gamma	4.00	1.00	$\sigma_{z^i}^0$	Gamma	17.15	17.15
γ	Beta	0.50	0.29	$\sigma_{z^i}^4$	Gamma	7.00	7.00
κ	Gamma	4.00	1.00	$\sigma_{z^i}^8$	Gamma	7.00	7.00
δ_2/δ_1	Inv. Gamma	0.68	2.59	σ_z^0	Gamma	1.50	1.50
b	Beta	0.50	0.20	σ_z^4	Gamma	0.61	0.61
ρ_{x_g}	Beta	0.70	0.20	σ_z^8	Gamma	0.61	0.61
$\rho_z/0.99$	Beta	0.70	0.20	σ_μ^0	Gamma	1.19	1.19
$\rho_a/0.99$	Beta	0.50	0.20	σ_μ^4	Gamma	0.49	0.49
$\rho_g/0.99$	Beta	0.70	0.20	σ_μ^8	Gamma	0.49	0.49
$\rho_x + 0.5$	Beta	0.70	0.20	σ_g^0	Gamma	1.05	1.05
$\rho_\mu/0.99$	Beta	0.70	0.20	σ_g^4	Gamma	0.43	0.43
$\rho_\zeta/0.99$	Beta	0.50	0.20	σ_g^8	Gamma	0.43	0.43
$\rho_{z^i}/0.99$	Beta	0.50	0.20	σ_ζ^0	Gamma	6.30	6.30
$\sigma_{\mu_a}^0$	Gamma	0.31	0.31	σ_ζ^4	Gamma	2.57	2.57
$\sigma_{\mu_a}^4$	Gamma	0.13	0.13	σ_ζ^8	Gamma	2.57	2.57
$\sigma_{\mu_a}^8$	Gamma	0.13	0.13	σ_{ygr}^{me}	Uniform	0.00	0.30
$\sigma_{\mu_x}^0$	Gamma	0.45	0.45				
$\sigma_{\mu_x}^4$	Gamma	0.19	0.19				
$\sigma_{\mu_x}^8$	Gamma	0.19	0.19				

Notes: Para (1) and Para (2) correspond to the mean and standard deviation of the Beta, Gamma, and Normal distributions and to the upper and lower bounds of the support for the Uniform distribution. For the Inv. Gamma distribution, Para (1) and Para (2) refer to s and ν, where $p(\sigma|\nu, s) \propto \sigma^{-\nu-1}e^{-\nu s^2/2\sigma^2}$.

Figure A-2 plots the evolution of the effective sample size as a function of n (where $N_\phi = 500$). As is apparent from the figure, there are about 15 resampling steps out of a total 500 stages, or about 3% of the time. We view this as a reasonable performance for the algorithm.

Figure A-2: The Evolution of the Effective Sample Size for the News Model