Tailored Randomized-block MCMC Methods for Analysis of DSGE Models*

Siddhartha Chib        Srikanth Ramamurthy†

September 2008; January 2009; March 2009

Abstract

In this paper we develop new Markov chain Monte Carlo schemes for Bayesian estimation of DSGE models. The motivation for our work arises from some of the shortcomings of the single block random walk Metropolis Hastings (M-H) algorithm (RW-MH), the sampling method that has been used to date in this context. In our basic tailored randomized block (TaRB-MH) algorithm, the parameters of the model are randomly clustered at every iteration into an arbitrary number of blocks. Then each block is sequentially updated through an M-H step. Furthermore, the proposal density for each block is tailored to the location and curvature of the target density based on the output of simulated annealing, following Chib and Greenberg (1994, 1995) and Chib and Ergashev (2008). We also provide an extension of the TaRB-MH algorithm for sampling multi-modal distributions. In this version, which we refer to as the TaRBMJ-MH algorithm, at a pre-specified mode jumping iteration (say every 100th), a single-block proposal is generated from one of the modal regions using a mixture proposal density, and this proposal is then accepted according to an M-H probability of move. At the non-mode jumping iterations, the draws are obtained by applying the TaRB-MH algorithm. The methodological developments are completed by showing how the approach in Chib (1995) and Chib and Jeliazkov (2001) can be adapted to these sampling schemes for estimating the model marginal likelihood. We illustrate our methods with the aid of stylized problems and three DSGE models that have appeared in the literature. The first is the model in Ireland (2004) where we show that the TaRB-MH algorithm is more reliable and efficient than the RW-MH algorithm. Our second example is the model in An and Schorfheide (2007). The posterior distribution in this model is more challenging to simulate on account of multiple modes. As shown by these authors, the RW-MH algorithm is unable to jump from the low modal region to the high modal region, and vice-versa. The TaRBMJ-MH

*We gratefully acknowledge the support from Center for Research in Economics and Strategy (CRES), in the Olin Business School, Washington University in St. Louis.
†Chib: Olin Business School, Washington University in St. Louis, Campus Box 1133, 1 Bookings Drive, St. Louis, MO 63130 (e-mail: chib@wustl.edu); Ramamurthy: Department of Economics, Washington University in St. Louis, Campus Box 1208, 1 Bookings Drive, St. Louis, MO 63130 (e-mail: sramamur@artsci.wustl.edu).
method, on the other hand, does not suffer from this problem and moves between the two modal regions, exploring the posterior distribution globally in an efficient manner. The final example is the model in Smets and Wouters (2007) that involves a 36-dimensional posterior distribution. We show that the performance of our TaRB-MH algorithm is unaffected by the dimension of the posterior distribution. The sample autocorrelation functions decay to zero within 30-40 lags for most parameters. In contrast, the sampled draws from the RW-MH chain are highly persistent, exhibiting significant autocorrelation even at lags 2500 and beyond. In addition, the RW-MH does not explore the same high density regions of the posterior distribution as the TaRB-MH algorithm.

Key words: Dynamic Stochastic General Equilibrium models; Markov chain Monte Carlo; Metropolis-Hastings algorithm; marginal likelihood; randomized blocks; tailored proposal densities; multi modal densities; simulated annealing.

1 Introduction

Dynamic stochastic general equilibrium (DSGE) models are arguably the dominant framework for modeling and analyzing macroeconomic phenomena. Over the past few years there has been substantial interest in the fitting of these models from a Bayesian perspective implemented by Markov chain Monte Carlo (MCMC) methods (for example, Fernandez-Villaverde and Rubio-Ramirez (2004), Lubik and Schorfheide (2004), Smets and Wouters (2003, 2007), An and Schorfheide (2007), Canova (2007), DeJong and Dave (2007)). One reason for the interest in the Bayesian approach is that it provides an avenue (namely through the prior distribution) for incorporating substantive information about the parameters. This proves important because the parameters in these models are often ill-determined (or unreasonable) when fit by maximum likelihood on account of the stylized nature of the models and the relatively small data sets that can be utilized for the fitting. The attractiveness of Bayesian methods has also grown due to the availability of powerful MCMC simulation methods that provide the technology for sampling the posterior distribution of the parameters (Chib and Greenberg (1995), Chib (2001)).

One feature of the current generation of Bayesian DSGE fitting methods is that they sample the posterior distribution by what is formally known as a single block random-walk Metropolis-Hastings (M-H) algorithm (RW-MH). In the RW-MH algorithm the parameters are sampled in a single block by drawing a proposal from a random walk process. This proposal value is then accepted as the next draw according to the
corresponding M-H probability of move (which in this case is essentially the ratio of the posterior density at the proposed value and the posterior density at the current value); if the proposed value is rejected, the current value is retained as the new value of the Markov chain.

The main appeal of this sampling method is its speed and relative ease of implementation. It has been noticed, however, that in the realm of DSGE models, where the posterior distributions can be irregular, the tuning of the RW-MH algorithm is not straightforward. Substantial pre-run tuning effort is often necessary, which can add considerably to the overall time of implementation. For one, it turns out that the effect of the starting value of the Markov chain can wear off slowly, thus necessitating large simulation sample sizes of the order of many hundreds of thousands. In some situations, for instance in a multi-modal problem, the effect of the starting value may not wear off in realistic sampling time. This problem was demonstrated by An and Schorfheide (2007) who noted that the algorithm was unable to escape the region surrounding one of the modes. We revisit this example later in the paper. Another problem is that the variance of the increment in the random walk proposal can be difficult to set, especially in higher-dimensional problems, and the sampler performance can be severely comprised by a poor choice of it. With too small a variance the search process can be extremely slow, whereas with a large variance there can be many rejections and the same value can be repeated many times in the chain. As a result, the sampled sequence tends to exhibit high serial correlations and slow convergence to the posterior distribution.

Our goal in this paper is to suggest new MCMC schemes for fitting DSGE models that are more reliable than the existing approach. We achieve this by combining the efficiency of tailored proposals (Chib and Greenberg (1994)) with a flexible blocking strategy that virtually eliminates pre-run tuning. In our basic approach, which we refer to as the Tailored Randomized Block M-H or TaRB-MH algorithm, the parameters of the model are randomly clustered at every iteration into a random number of blocks. Then each block is sequentially updated through an M-H step in which the proposal density is tailored to closely mimic the target density of that block. Since the posterior surface can be irregular in DSGE models, we construct our tailored proposal densities from the output of simulated annealing, following Chib and Ergashev (2008). In effect, the
TaRB-MH algorithm has the features of a general black box approach that requires little user involvement. We also provide an extension of this algorithm for sampling multi-modal distributions. In this version, which we refer to as the TaRBMJ-MH algorithm, at a pre-specified move jumping iteration (say every 100th), a single-block proposal is generated from one of the modal regions using a mixture proposal density, and this proposal is then accepted according to an M-H probability of move. At the non-mode jumping iterations, the draws are obtained by applying the TaRB-MH algorithm. It should be noted that while these methods are developed with the DSGE applications in mind, the methods are general and can be applied to any Bayesian problem.

We complete our methodological developments by presenting an approach for calculating the model marginal likelihood, which is needed for the comparison of alternative DSGE models. The method we present is based on the framework of Chib (1995), and its M-H version in Chib and Jeliazkov (2001), with suitable modifications to accommodate the randomized block approach proposed in this paper.

We illustrate the performance of our methods with the aid of three DSGE models. The first is the model in Ireland (2004). In this example we provide a comparative analysis of the RW-MH and TaRB-MH algorithms. Specifically, we pick three different starting values - the prior mean, a local mode and the dominant mode. We also specify different variance-covariance matrices for the RW proposal. We then run three independent RW chains initiated at each of the starting values for 250,000 iterations following a burn-in of 50,000 iterations. Our results indicate that the chains initialized at the first two values do not fully navigate the posterior distribution. In some instances, this problem is not overcome even when the simulation sample size is increased to one million. In contrast, the sampled draws from the TaRB-MH algorithm quickly move to the same regions of the parameter space, regardless of the starting value. The TaRB-MH output is also efficient with the sample autocorrelation functions (ACF) of the draws decaying to zero within 15-20 lags for most parameters. As a result, a simulation sample size of, say, 10,000 with a burn-in of a 1000 iterations, is sufficient to adequately explore the posterior distribution in this problem.

Our second example is the model in An and Schorfheide (2007). The posterior distribution in this model is more challenging to simulate on account of possibly two
modes. As shown by the authors, the RW sampler was simply unable to escape the attraction of the individual modes even when the sampler was run for a million iterations following a burn-in of 100,000 iterations. What is especially interesting is that, despite a difference of approximately 8 on the log scale between the unnormalized posterior ordinates at the two modes, the RW chain did not cross over from the low modal region to the high modal region. The TaRB-MJ-MH method, on the other hand, jumps to the high mode almost instantaneously and explores the posterior distribution globally in an efficient manner.

Finally, we apply the TaRB-MH algorithm to the Smets and Wouters (2007) model to illustrate its performance in the context of a large-scale (36 parameter) DSGE model. Given the dimension of the parameter space and the complexity of the posterior surface, it is challenging to find the mode of the posterior distribution. This in turn makes it difficult to tune and implement the RW-MH algorithm. Even the best performing RW-MH algorithm produces output with autocorrelations at lags 2500 and beyond. In comparison, the performance of the TaRB-MH algorithm is unaffected by the dimension of the problem and its complexity. The autocorrelation functions of the sampled output decay to zero within 30-40 lags for most parameters and the sampler explores high density regions of the posterior distribution that are missed by the RW-MH algorithm.

The rest of the paper is organized as follows. To motivate the discussion of our methods, as well as for completeness of the paper, we begin with a brief discussion of the general setup of linearized DSGE models in the following section. In Section 3 we present the details of our approach for sampling the posterior distribution and discuss a suitable way of calculating the marginal likelihood. Section 4 deals with the application of our methods to the two DSGE models mentioned above. Concluding remarks are provided in Section 5.

2 The setup

In this paper our focal application is the class of macroeconomic models commonly referred to as DSGE models. These are structural microfounded models set in a general equilibrium framework. Thus, the outcome of the economic variables is determined by the collective actions of the agents in the economy being modeled. Further, in the face
of uncertainty, the agents take into account the future expected values of the variables when making their decisions. In the ensuing discussion, as well as in the applications in Section 3, this decision making process is assumed to be based on rational expectations (expectations based on the full information set available up to the current period). For both theoretical and inferential purposes, one is usually interested in analyzing the behavior of the economy around its deterministic steady state. Since these models are typically highly nonlinear, one often works with first order approximations. This reduces the model to a system of stochastic expectational difference equations, popularly known as a linear rational expectations model. For the rest of this paper, this is our structural model of interest.¹

As discussed in An-Schorfheide (2007), Sargent (1989), Smets-Wouters (2003) and others, the process of transforming a structural model to its reduced form involves two steps. The first step is to solve the model. In the solution step, the expectational terms are integrated out to obtain a stochastic law of motion for the endogenous variables of the model. Here, one generally restricts attention to the subset of parameter space, \( \mathcal{S}_D \), that generates a unique stable solution to the model (the determinate solution). Several solution methods have been proposed in the literature to solve linear rational expectations models (for a concise comparative discussion of these methods see Anderson (2008)). We adopt the method in Sims (2002) to obtain the model solution. Following this approach the variables \( s_t \) in the model are written as a function of their past values (the vector autoregressive component), the innovations \( \varepsilon_t \) (assumed to be a zero mean Gaussian random vector) and an endogenous rational expectations error \( \eta_t \)

\[
G_0(\theta)s_t = G_1(\theta)s_{t-1} + G_2(\theta)\varepsilon_t + G_3(\theta)\eta_t.
\] (1)

Here, \( G_j^{3}_{j=0} \) are matrices of appropriate dimensions involving the parameters \( \theta \) of the model. Henceforth, we let \( \mathcal{S}_L \) denote the linear restrictions on the parameters imposed by the underlying theoretical model.

The details of the solution algorithm are beyond the scope of this paper. We simply note that the solution proceeds by applying the generalized Schur decomposition to the pair of matrices \((G_0, G_1)\) from which the set of generalized eigenvalues of the preceding

¹These stylized assumptions are inconsequential from the point of view of this paper as the MCMC methods are applicable to a wide variety of problems.
matrix pair can be computed. The general idea behind this exercise is to isolate the stable and unstable blocks of the system. Loosely speaking, for a unique solution to exist, we require that the number of unstable roots of the system equal the number of non-predetermined variables in the model. The solution is then constructed by solving ‘forward’ (‘backward’) the unstable (stable) block of the system. From an empirical standpoint, the convenience of a linear framework is that the output of the solution is a vector Markov process for the endogenous variables of the model

\[ s_t = D(\theta)s_{t-1} + F(\theta)\varepsilon_t \tag{2} \]

where the matrices \( D(\theta) \) and \( F(\theta) \) are awkward implicit functions of the model parameters, obtained from the solution.

For purposes of conducting inference on the parameters, the preceding vector Markov process can be viewed as the state transition equation in a linear Gaussian state space model (SSM). Accordingly, the SSM formulation is completed by defining a measurement equation that relates the states \( s_t \) to a vector of observable data \( y_t \).

\[ y_t = a(\theta) + B(\theta)s_t \tag{3} \]

Depending on the model and representation of the data, the vector \( a \) and the matrix \( B \) may be constants or involve the parameters of the model.

Let \( \Omega(\theta) \) be the variance-covariance matrix of \( \varepsilon \). It is assumed that the parameters in \( \Omega \) satisfy the usual positivity and positive definiteness constraints, \( S_\Omega \). Given a sample of data \( y_{n\times T} = \{y_t\}, t = 1, \ldots, T \), the joint density of the data (the likelihood function) given the parameters can be calculated as

\[ f(y|\theta) = \prod_{t=1}^{T} \frac{1}{(2\pi)^{n/2}|\Sigma_{t|t-1}|^{1/2}} \exp \left\{ -\frac{1}{2} \eta_{t|t-1}' \Sigma_{t|t-1}^{-1} \eta_{t|t-1} \right\} I_{S_L}(\theta) I_{S_n}(\theta) I_{S_D}(\theta) \tag{4} \]

where \( \eta_{t|t-1} = y_t - a - B\hat{s}_{t|t-1} \) is the one-step ahead prediction error, \( \hat{s}_{t|t-1} \) is the one-step ahead estimate of the state and \( \Sigma_{t|t-1} \) is the conditional variance of the prediction error, each given information up to time \( (t - 1) \). These quantities are obtained from the usual Kalman filter recursions (see for instance, Harvey, 1990).

It is important to recognize that in most practical models of interest, the solve procedure can only be performed numerically for a given value of the parameters. This means
that the model has to be solved numerically for every value of the parameters at which
the likelihood function is to be evaluated. Also notice that the likelihood function is re-
stricted to the parameter space $\theta \in S_L \cap S_R \cap S_D$. Of these, the determinacy constraint
is the most tedious to verify as it requires the model to be solved for the particular
value of the parameters. Therefore, to verify if a particular value of the parameters
satisfies the restrictions, it is efficient to check the constraints in the order listed above.
Alternatively, when possible, one could also work with transformed parameters that are
unrestricted, requiring only the determinacy check.

In the Bayesian context, the empirical state space model is completed with a prior
distribution $\pi(\theta)$ on the parameters. The posterior distribution of the parameters $\pi(\theta|y)$
is calculated (up to the normalizing constant) as the product of the likelihood function
and the prior distribution

$$
\pi(\theta|y) \propto f(y|\theta) \times \pi(\theta)
$$

(5)

From the Bayesian point of view, a summary of the posterior (for instance, the mean
or the median) serves as an estimate of the parameters. However, as the posterior
distribution is typically intractable, such summaries are not obtained easily. Instead,
one relies on MCMC methods to draw sample variates from the posterior distribution.
These sampled draws are then used to find the posterior mean, posterior quantiles and
other summaries of the distribution. In the following section we discuss how this general
approach can effectively operationalized.

3 Posterior sampling and marginal likelihood esti-

mation

3.1 Basic sampling approach

In MCMC methods, one samples the posterior density by simulating a Markov chain
whose invariant density is the specified target density. The sampled draws, beyond a
suitable burn-in phase, are then taken as a (correlated) sample from the posterior density.
While the general idea is clear, designing a sampler that mixes well (has low serial
correlations in the sampled draws), and converges quickly to the invariant distribution,
is challenging, especially in the context of DSGE models.
The sampling schemes that we focus on belong to the class of tailored multiple block MH algorithms as discussed in Chib and Greenberg (1994, 1995). In a multiple-block MH algorithm, the parameters are grouped into several distinct blocks, and then each block of parameters is updated in sequence by a MH step, conditioned on the most current value of the parameters in the remaining blocks. It is noteworthy that a multiple-block M-H algorithm has not been implemented before in the setting of DSGE models. It is our aim to show, however, that versions of the multiple-block M-H algorithm are capable of effectively sampling the complex posterior distributions that arise from DSGE models.

A key question in the multiple block sampling method is the number and composition of the blocks. One principle is to form groups so that parameters in different blocks are not strongly correlated whereas those within a block are. Unlike usual statistical models, however, where location and scale parameters (for instance) may be approximately orthogonal and hence placed in separate groups, in a DSGE model such a priori knowledge is rarely available. In fact, because of the solve process, which converts the structural model into a state-space model as discussed above, the parameters become scrambled in an unspecified way, thus creating correlations between parameters that cannot be analytically isolated.

Another complication is that the DSGE model posteriors tend to be irregular so that the “grouping by correlation” criterion is not necessarily ideal. To see why, suppose that a bivariate posterior distribution has positively sloped contours for some portion of the support but has vertical contours for another portion of the support. In that case, it is meaningful in the positively sloped regions to sample the parameters in a group, but to sample the parameters one at a time to escape into the other portion of the support. In other words, in irregular problems, a permanent grouping of parameters in one block is not necessarily optimal.

Our novel idea to blocking, which works well and has never been tried before in the MCMC literature, is to randomize the formation of blocks in every iteration. This means that both the number of blocks and its components are randomized. A related but different randomization approach is utilized to great effect in Pitt and Shephard (1997) and Elerian, Chib and Shephard (2001) where, in the context of state space and diffusion models, respectively, the block sizes for contiguous time ordered variables are
randomized. In that blocking strategy, therefore, the time ordered variables are never scrambled, just the length of the sequence of variables that form a block changes. In contrast, in our scheme, the blocks are formed from a scrambled set of parameters. One clear virtue of our randomized blocking scheme is that it avoids the pitfalls from a poor choice of a priori blocks. Another is that it allows for the groupings to change, which is important in irregular problems (as in the preceding illustration). We note that (if necessary) such a scheme could also be organized to allow for some parameters in $\theta_l$ to form a fixed block, and the remaining parameters to form random blocks. In this sense, this randomized blocking strategy is flexible enough to accommodate different model specific blocking needs.

The second component of our approach is that the proposal density in the M-H step is constructed to adapt to both the location and curvature of the posterior density of that block. While this notion of local tailoring dates back to Chib and Greenberg (1994, 1995), we further enhance this approach by basing the tailored densities on the output of the simulated annealing optimization method (following Chib and Ergashev (2008)). This enhancement is particularly helpful in dealing with irregular distributions, and also for localizing the proposal density far enough away from the current region (depending on where the global mode for that block of parameters is located) to permit large moves.

Our multiple block tailored M-H algorithm can be summarized as follows.

**Algorithm:** TaRB-MH algorithm

**Step 1** Initialize $\theta^{(0)} \in S_L \cap S_{\Omega} \cap S_D$ and fix $n_0$ (the burn-in) and $M$ (the MCMC sample size)

**Step 2** In each iteration $j$, $j = 1, \ldots, n_0 + M$, randomly generate blocks $(\theta_{j,1}, \theta_{j,2}, \ldots, \theta_{j,p_j})$

**Step 3** Within each iteration, sample each block $\theta_{j,l}$, $l = 1, \ldots, p_j$, by an M-H step with a tailored proposal density as described below.

**Step 4** Repeat steps 2-3 $n_0 + M$ times, discard the draws from the first $n_0$ iterations and save the subsequent $M$ draws $\theta^{(n_0+1)}, \ldots, \theta^{(n_0+M)}$

Generating random blocks in Step 2 above is, of course, straightforward and does not require comment. We therefore focus on an explanation of Step 3. Suppose that at the
end of the \((j-1)\)st iteration of the MCMC iteration, there are \(p_j\) randomly constructed blocks \((\theta_{j,1}, \ldots, \theta_{j,p_j})\). Let \(\theta_{j,-l}\) denote the most current value of all the blocks except the \(l\)th. Then to construct the tailored proposal density for \(\theta_{j,l}\) we find

\[
\hat{\theta}_{j,l} = \arg \max_{\theta_{j,l}} \log \{ f(y|\theta_{j,l}, \theta_{j,-l}) \times \pi(\theta) \}
\]

from a suitably formulated version of simulated annealing.

In simulated annealing one proposes a random modification to the current guess of the maximum which is then accepted or rejected probabilistically. Moves that lower the function value can sometimes be accepted. The probability of accepting such downhill moves declines over iterations according to a “cooling schedule,” thus allowing the method to converge. Specifically, starting from an initial temperature \(t_0\) and an initial guess for the maximum \(\theta^{(0)}_{j,l}\), such that \((\theta_{j,1}, \ldots, \theta^{(0)}_{j,l}, \ldots, \theta_{j,p_j}) \in S_L \cap S_{\Omega} \cap S_D\), repeated proposals are generated for a randomly chosen element \(\theta_m\) of \(\theta_{j,l}\) using a random walk process

\[
\theta'_m = \theta_m + s \mathcal{N}(0,1)
\]

where \(s > 0\) is a suitable scale factor. As in the M-H step we restrict the proposals to the constrained set. Perturbations resulting in a higher function value are always accepted, whereas those resulting in a lower function evaluation are accepted with probability

\[
p = \exp\{\Delta[\log(f(y|\theta_{j,l}, \theta_{j,-l}) \times \pi(\theta))] / t \} < 1
\]

where \(\Delta[\log(f(y|\theta_{j,l}, \theta_{j,-l}) \times \pi(\theta))]\) is the change in the log of the objective function, computed as the log of the objective function at the perturbed value of the parameters minus the log of the objective function at the existing value of the parameters. As one can verify, this probability is a decreasing function of the temperature, with convergence achieved as \(t \to 0\).

Our implementation of this algorithm is as follows. We first divide the search process into various stages, denoted by \(k\), \(k = 1, 2, \ldots, K\), with the length of each stage \(l_k\) given by \(b + l_{k-1}\), where \(b \in \mathbb{N}\) is the stage expansion factor. We then specify the initial temperature \(t_0\) which is held constant in each stage but reduced across stages according to the linear cooling schedule \(t_k = at_{k-1}\), where \(0 < a < 1\) is the cooling constant. Within each stage, the algorithm searches for the maximum, proposing values from a random walk process as mentioned above.
Once $\hat{\theta}_{j,l}$ has been so found, we calculate the curvature of the target posterior distribution of that block as the negative inverse of the Hessian:\footnote{The covariance matrix computed as the inverse of the negative Hessian is not always guaranteed to be positive definite, especially in the context of DSGE models where the target can be highly irregular. When the negative Hessian is not positive definite, we compute the modified cholesky $R$ of the negative Hessian. Subsequently, we specify $(RR')^{-1}$ as the variance of the proposal density. While there are several versions of the modified cholesky, our implementation is based on algorithm 3.4 in Nocedal and Wright (2004).}

$$V_{j,l} = \left( - \frac{\partial^2 \log \{ f(y|\theta_{j,l}, \theta_{j,-l}) \times \pi(\theta) \} }{ \partial \theta_j \theta'_j } \right)_{\theta_{j,l} = \hat{\theta}_{j,l}}^{-1}$$

Our proposal density $q_l(\theta_{j,l}|\theta_{j,-l}, y)$ of $\theta_{j,l}$ is then given by

$$q_l(\theta_{j,l}|\theta_{j,-l}, y) = t(\theta_{j,l}|\hat{\theta}_{j,l}, V_{j,l}, \nu),$$

a multivariate student-$t$ density with $\nu > 2$ degrees of freedom. Note that this proposal density depends on the current values of the remaining parameters (blocks) and changes from one iteration to the next as the constituents of the blocks change.

We now draw a proposal value $\theta^\dagger_{j,l}$ from the latter proposal density. If the proposal value violates any of the constraints it is rejected immediately. Otherwise, the proposed value is taken as the new value of the block with the M-H probability of move given by

$$\alpha_l(\theta_{j,l}, \theta^\dagger_{j,l}|\theta_{j,-l}, y) = \min \left\{ \frac{f(y|\theta_{j,l}, \theta^\dagger_{j,l}) \pi(\theta^\dagger_{j,l})}{f(y|\theta_{j,l}, \theta_{j,-l}) \pi(\theta_{j,l})} \frac{t(\theta_{j,l}|\hat{\theta}_{j,l}, V_{j,l}, \nu)}{t(\theta^\dagger_{j,l}|\hat{\theta}_{j,l}, V_{j,l}, \nu)}, 1 \right\}$$

If the proposed value is rejected, the current value of the block is retained as the new value of that block. Step 3 of the algorithm is completed by repeating this process for each block.

As an instructive example, consider the following SSM to which no particular economic interpretation is attached.

$$y_t = a + B s_t + u_t$$

$$s_t = G s_{t-1} + \varepsilon_t$$

where $y_t$ is a $10 \times 1$ vector of observables at time $t$, $s_t$ is a $5 \times 1$ vector of time-$t$ unobserved (latent) states, $a$, $B$ and $G$ are matrices of appropriate dimensions, $u_t \sim \mathcal{N}_{10}(0, \Sigma)$ and $\varepsilon_t \sim \mathcal{N}_{5}(0, \Omega)$. We impose the following identification restrictions: $G$ is diagonal,
\( \mathbf{B}_{i,i} = 1 \) for \( i = 1, \ldots, 5 \), \( \mathbf{B}_{i,j} = 0 \) for \( i, j = 1, \ldots, 5 \), \( j > i \), \( \Sigma = \text{diag}\{\sigma^2_i\}_{i=1}^{10} \) and \( \Omega = \mathbf{I}_5 \).

This leads to 60 unknown parameters that we collect in the vector \( \bm{\theta} \). To ensure that the variance of the measurement errors are positive, we reparameterize \( \Sigma \) as

\[
\Sigma^* = \text{diag}\{\sigma^*_{2i}\}_{i=1}^{10}, \quad \sigma^*_{2i} \in \mathcal{R}
\]

where \( \sigma^2_i = \exp(\sigma^*_{2i}) \). Finally, we restrict the parameter space \( \Theta \) to satisfy stationarity of the vector Markov process \( \mathbf{s}_t \): \( \Theta_S = \{\bm{\theta} : \text{abs}(\text{eig}(\mathbf{G})) < 1\} \), which in this case is trivially satisfied if \( G_{i,i} < 1 \), \( i = 1 \ldots, 5 \).

We generate 200 observations from the data generating process \( \bm{\theta}_{DGP} \), summarized in table 1. The likelihood function \( f(\mathbf{y}|\bm{\theta}) \) can be calculated through the Kalman filter as outlined in Section 2. The model is completed by specifying a prior distribution \( \pi(\bm{\theta}) \) of the parameters. For notational convenience, we denote the parameters in the principal diagonal of \( \mathbf{G} \) as \( \theta_1 = \text{vecr}([\mathbf{G}_{i,i}]) \), \( i = 1 \ldots, 5 \), those in \( \mathbf{a} \) as \( \theta_2 \), the free parameters in the lower triangular portion of \( \mathbf{B} \) as \( \theta_3 = \text{vecr}([\mathbf{B}_{i,j}]) \), \( i, j = 2, \ldots, 5 \), \( j < i \), the remaining parameters in \( \mathbf{B} \) as \( \theta_4 = \text{vecr}([\mathbf{B}_{i,j}]) \), \( i = 6, \ldots, 10 \), \( j = 1, \ldots, 5 \) and the diagonal elements of \( \Sigma^* \) as \( \theta_5 = \{\sigma^*_{2i}\}_{i=1}^{10} \), where vecr stands for row vectorization. Our specified joint distribution of the parameters takes the form

\[
\pi(\bm{\theta}) = \mathcal{N}(\theta_1|\mathbf{g}_0, \mathbf{V}_g)\mathcal{N}(\theta_2|\mathbf{a}_0, \mathbf{V}_a)\mathcal{N}(\theta_3|\theta_{30}, \mathbf{V}_{\theta_3})\mathcal{N}(\theta_4|\theta_{40}, \mathbf{V}_{\theta_4})\mathcal{N}(\theta_5|\sigma^*_{0}, \mathbf{V}_{\sigma^*})\mathbf{I}_{\Theta_S}.
\]

For the hyperparameters, we let

\[
\begin{align*}
g_{0k} &= 0.5; \quad k = 1, \ldots, 5 \\
\mathbf{V}_g &= 5 \times \mathbf{I}_5 \\
a_{0k} &= 0.5; \quad k = 1, \ldots, 10 \\
\mathbf{V}_a &= 5 \times \mathbf{I}_{10} \\
\theta_{30k} &= 0; \quad k = 1, \ldots, 10 \\
\mathbf{V}_{\theta_3} &= 5 \times \mathbf{I}_{10} \\
\theta_{40k} &= 0; \quad k = 1, \ldots, 25 \\
\mathbf{V}_{\theta_4} &= 5 \times \mathbf{I}_{25} \\
\sigma^*_{0k} &= -1; \quad k = 1, \ldots, 10 \\
\mathbf{V}_{\sigma^*} &= \mathbf{I}_{10}
\end{align*}
\]
These choices reflect the belief that apriori the parameters are independent, as well as that relatively little is known about them. Further, the $N(-1, 1)$ prior for $\sigma^*_i$ implies that $\sigma_i$ is lognormally distributed with an approximate mean of 0.61 and variance of 0.63. The posterior distribution (up to the normalizing constant) can now be written as

$$\pi(\theta | y) \propto f(y | \theta) \times \pi(\theta).$$

This is the target function that we would like to explore.

Following step 1 in the TaRB-MH algorithm, we initialize the sampler at $\theta = (g_0, a_0, \theta_{30}, \theta_{40}, \sigma^*_0)$. Then steps 2 and 3 are iterated 11,000 times as follows. In every iteration the parameters are first randomly grouped into various blocks. In our scheme this is accomplished by randomly permuting the index of the parameters at the beginning of an iteration. Then block 1 is initialized at the first element of this shuffled index. A new block is started at every subsequent element with probability 0.15 (or in other words, with probability 0.85 that element is included in the current block). Once the blocks have been so formed in a given iteration, we proceed to step 3, setting the parameters of the simulated annealing algorithm at $t_0 = 5, a = 0.4, K = 8, b = 8$ and $s = 0.02$. For each block, we now find the mode and curvature of the current target density to construct a student-$t$ proposal density with 15 degrees of freedom. We then apply the M-H step, revise the value of that block according to the probability given in (7), and repeat this process for the next block. The first 1000 draws are discarded as burn-ins and the last 10,000 are used to summarize the posterior distribution.

Figure 1 shows the kernel smoothed histogram of the marginal posterior density for 8 select parameters along with the corresponding prior distribution. Notice that the marginal posterior densities for all the parameters except those in $a$ are roughly centered around the data generating process. Also, the parameters in $a$ rightly capture the true mean of the data. With 200 observations, the role of the prior distribution is diminished because the likelihood function carries substantial information about the parameters. The figure also includes plots of the autocorrelation function. As seen in these plots, the autocorrelation among the sampled draws drop to zero within very few iterations.

To provide a comparative analysis of these results, we sample the posterior distribution with two additional MCMC schemes. The first is a single block RW-MH sampler. For this sampler we first conduct a thorough search for the posterior mode to obtain a
Table 1–DGP and prior moments of parameters in SSM example

<table>
<thead>
<tr>
<th>Param.</th>
<th>DGP</th>
<th>Prior</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_{1,1}, \ldots, G_{5,5}$</td>
<td>0.80</td>
<td>0.50</td>
</tr>
<tr>
<td>$G_{6,1}, \ldots, G_{10,5}$</td>
<td>0.20</td>
<td>0.50</td>
</tr>
<tr>
<td>$a_1, \ldots, a_5$</td>
<td>0.10</td>
<td>0.50</td>
</tr>
<tr>
<td>$a_6, \ldots, a_{10}$</td>
<td>1.00</td>
<td>0.50</td>
</tr>
<tr>
<td>$B$</td>
<td>1.00</td>
<td>—</td>
</tr>
<tr>
<td>$\sigma_1^2, \ldots, \sigma_5^2$</td>
<td>0.30</td>
<td>0.61</td>
</tr>
<tr>
<td>$\sigma_6^2, \ldots, \sigma_{10}^2$</td>
<td>0.50</td>
<td>0.61</td>
</tr>
</tbody>
</table>

Note: Prior variance in paranthesis.

suitable variance-covariance matrix for the increment of the RW chain. As one might anticipate, optimization in a 50 dimensional parameter space can be significantly challenging. Indeed, despite converging repeatedly to the same parameter values with an unnormalized posterior ordinate of approximately $-2122.64$, the covariance matrix could not approximated due to numerical difficulties involving the inversion of the Hessian matrix. Consequently, we approximate the covariance by the modified cholesky approach.
Figure 1: Sampling results for the SSM example using the TaRB-MH algorithm: Marginal prior-posterior plots and autocorrelation functions for select parameters. Dotted lines indicate the prior distribution, whereas the bold lines indicate the posterior.

mentioned earlier. We also scale down the covariance matrix by a factor of $10^{-6}$ to achieve an acceptance rate of 30 percent. This sampler is then run for 1 million iterations following a burn-in of 250,000 iterations. Our second competing MCMC scheme is
the fixed block version of the TaRB-MH algorithm, henceforth referred to as FBTaRB-MH. Because there is no model solution involved in this problem, it is viable to construct fixed blocks without much experimentation. In fact, a natural blocking strategy here, and more generally in the context of dynamic factor models that share the same structure, is to permanently group the parameters in $G$ in one block, $a$ in a second block, $B$ in a third block, and $\Sigma$ in a fourth block. However, because of the relatively large dimensionality of $B$, we further partition those parameters into the free ones in the upper and lower $5 \times 5$ matrix blocks. In other words, utilizing the same notations introduced above, $\theta = (\theta_1^\prime, \theta_2^\prime, \theta_3^\prime, \theta_4^\prime, \theta_5^\prime)$. With the blocks so constructed, we run the sampler for 11,000 iterations and discard the first 1000 draws. We also set the tailoring parameters in simulated annealing to be identical to our TaRB-MH sampler above.

Not surprisingly, the results from the RW chain are less than satisfactory with sample autocorrelation functions (ACF) in the order of 0.9 for lags as high as 5000 and above for several parameters. In comparison, the FBTaRB-MH algorithm performs very well with the serial correlation among the draws decaying to zero within 250 lags for most parameters. However, the TaRB-MH algorithm aces both the preceding MCMC schemes. The sample ACFs vanish within 70 lags for all but the parameters in $a$.

The serial correlations among the sampled draws can be summarized in terms of the so-called inefficiency factors. The inefficiency factors approximate the ratio of the numerical variance of the posterior mean from the MCMC chain relative to that from hypothetical iid draws. For a given sequence of draws the inefficiency factor is computed as

$$1 + 2 \sum_{l=1}^{L} \rho_k(l)$$

where $\rho_k(l)$ is the autocorrelation at lag $l$ for the $k$th sequence, and $L$ is the value at which the autocorrelation function tapers off (the higher order autocorrelations are also downweighted by a windowing procedure, but we ignore this aspect for simplicity). The inefficiency factors from all three sampling schemes, together with a summary of the posterior distribution are given in Tables 2a and 2b. As can be seen in this table, the TaRB-MH algorithm produces virtually i.i.d. draws for the parameters in $B$, $G$ and $\Sigma$. For the remaining 8 parameters, the highest inefficiency factor in the case of the TaRB-MH algorithm is still less than 50. In comparison, the inefficiency factors range
Table 2a–Posterior summary of parameters in $G$, $a$ and $\Sigma$

<table>
<thead>
<tr>
<th>Param</th>
<th>TaRB-MH</th>
<th>FBTaRB-MH</th>
<th>RW-MH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>St. dev.</td>
<td>Ineff</td>
</tr>
<tr>
<td>$G_{1,1}$</td>
<td>0.77</td>
<td>0.05</td>
<td>4.89</td>
</tr>
<tr>
<td>$G_{2,2}$</td>
<td>0.18</td>
<td>0.11</td>
<td>2.44</td>
</tr>
<tr>
<td>$G_{3,3}$</td>
<td>0.75</td>
<td>0.06</td>
<td>5.16</td>
</tr>
<tr>
<td>$G_{4,4}$</td>
<td>0.55</td>
<td>0.07</td>
<td>3.83</td>
</tr>
<tr>
<td>$G_{5,5}$</td>
<td>-0.09</td>
<td>0.12</td>
<td>2.52</td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.02</td>
<td>0.34</td>
<td>29.56</td>
</tr>
<tr>
<td>$a_2$</td>
<td>1.25</td>
<td>0.22</td>
<td>28.78</td>
</tr>
<tr>
<td>$a_3$</td>
<td>1.47</td>
<td>0.39</td>
<td>38.06</td>
</tr>
<tr>
<td>$a_4$</td>
<td>0.19</td>
<td>0.17</td>
<td>12.66</td>
</tr>
<tr>
<td>$a_5$</td>
<td>1.05</td>
<td>0.20</td>
<td>32.72</td>
</tr>
<tr>
<td>$a_6$</td>
<td>1.06</td>
<td>0.18</td>
<td>23.94</td>
</tr>
<tr>
<td>$a_7$</td>
<td>1.98</td>
<td>0.14</td>
<td>21.18</td>
</tr>
<tr>
<td>$a_8$</td>
<td>0.10</td>
<td>0.22</td>
<td>27.53</td>
</tr>
<tr>
<td>$a_9$</td>
<td>2.06</td>
<td>0.13</td>
<td>42.61</td>
</tr>
<tr>
<td>$a_{10}$</td>
<td>1.57</td>
<td>0.09</td>
<td>39.70</td>
</tr>
<tr>
<td>$\sigma_{1,1}^2$</td>
<td>1.33</td>
<td>0.14</td>
<td>2.81</td>
</tr>
<tr>
<td>$\sigma_{2,2}^2$</td>
<td>0.30</td>
<td>0.42</td>
<td>6.99</td>
</tr>
<tr>
<td>$\sigma_{3,3}^2$</td>
<td>1.00</td>
<td>0.19</td>
<td>4.97</td>
</tr>
<tr>
<td>$\sigma_{4,4}^2$</td>
<td>0.16</td>
<td>0.45</td>
<td>4.05</td>
</tr>
<tr>
<td>$\sigma_{5,5}^2$</td>
<td>0.63</td>
<td>0.30</td>
<td>5.31</td>
</tr>
<tr>
<td>$\sigma_{6,6}^2$</td>
<td>0.51</td>
<td>0.13</td>
<td>2.34</td>
</tr>
<tr>
<td>$\sigma_{7,7}^2$</td>
<td>0.92</td>
<td>0.13</td>
<td>2.84</td>
</tr>
<tr>
<td>$\sigma_{8,8}^2$</td>
<td>1.03</td>
<td>0.14</td>
<td>5.12</td>
</tr>
<tr>
<td>$\sigma_{9,9}^2$</td>
<td>0.86</td>
<td>0.12</td>
<td>2.56</td>
</tr>
<tr>
<td>$\sigma_{10,10}^2$</td>
<td>0.49</td>
<td>0.14</td>
<td>3.91</td>
</tr>
</tbody>
</table>

Note: Inefficiency factors (ineff) are based on 5000 lags. The acceptance rates in the three schemes are as follows: 47% in the TaRB-MH, 20%, 45% and 60%, respectively, for the three blocks in the FBTaRB-MH and 35% in the RW-MH algorithms.

between 3 and 168 in the FBTaRB-MH algorithm. In particular, the sampling of the parameters in $B$ are markedly more efficient with the TaRB-MH algorithm compared to the FBTaRB-MH. While these differences can be reduced slightly by further partitioning $B$ into sub-blocks, the virtues of the TaRB-MH algorithm are quite clear, especially when no such clear blocking strategy is available as in the context of DSGE models. As we show below, the TaRB-MH algorithm performs spiritedly in all 3 DSGE examples. Finally, note that in the RW-MH case, the inefficiency factors are all above 600, with
Table 2b–Posterior summary of parameters in B

<table>
<thead>
<tr>
<th>Param</th>
<th>TaRB-MH</th>
<th></th>
<th></th>
<th>FBTaRB-MH</th>
<th></th>
<th></th>
<th>RW-MH</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>St. dev.</td>
<td>Ineff</td>
<td>Mean</td>
<td>St. dev.</td>
<td>Ineff</td>
<td>Mean</td>
<td>St. dev.</td>
<td>Ineff</td>
</tr>
<tr>
<td>$B_{2,1}$</td>
<td>0.61</td>
<td>0.08</td>
<td>6.61</td>
<td>0.61</td>
<td>0.08</td>
<td>28.40</td>
<td>0.62</td>
<td>0.08</td>
<td>1661.23</td>
</tr>
<tr>
<td>$B_{3,1}$</td>
<td>0.55</td>
<td>0.14</td>
<td>10.71</td>
<td>0.55</td>
<td>0.14</td>
<td>102.08</td>
<td>0.56</td>
<td>0.13</td>
<td>2226.26</td>
</tr>
<tr>
<td>$B_{3,2}$</td>
<td>-0.33</td>
<td>0.15</td>
<td>4.25</td>
<td>-0.33</td>
<td>0.14</td>
<td>23.70</td>
<td>-0.34</td>
<td>0.14</td>
<td>873.90</td>
</tr>
<tr>
<td>$B_{4,1}$</td>
<td>0.07</td>
<td>0.10</td>
<td>11.17</td>
<td>0.06</td>
<td>0.11</td>
<td>57.06</td>
<td>0.08</td>
<td>0.11</td>
<td>2316.39</td>
</tr>
<tr>
<td>$B_{4,2}$</td>
<td>0.34</td>
<td>0.11</td>
<td>6.20</td>
<td>0.35</td>
<td>0.11</td>
<td>31.54</td>
<td>0.33</td>
<td>0.11</td>
<td>1084.67</td>
</tr>
<tr>
<td>$B_{4,3}$</td>
<td>-0.01</td>
<td>0.11</td>
<td>16.85</td>
<td>-0.01</td>
<td>0.10</td>
<td>51.07</td>
<td>-0.04</td>
<td>0.11</td>
<td>3999.89</td>
</tr>
<tr>
<td>$B_{5,1}$</td>
<td>-0.11</td>
<td>0.09</td>
<td>8.44</td>
<td>-0.11</td>
<td>0.09</td>
<td>83.88</td>
<td>-0.12</td>
<td>0.09</td>
<td>2272.21</td>
</tr>
<tr>
<td>$B_{5,2}$</td>
<td>0.14</td>
<td>0.13</td>
<td>4.39</td>
<td>0.14</td>
<td>0.13</td>
<td>24.28</td>
<td>0.14</td>
<td>0.13</td>
<td>1115.01</td>
</tr>
<tr>
<td>$B_{5,3}$</td>
<td>-0.51</td>
<td>0.08</td>
<td>4.12</td>
<td>-0.51</td>
<td>0.08</td>
<td>30.40</td>
<td>-0.51</td>
<td>0.08</td>
<td>1056.84</td>
</tr>
<tr>
<td>$B_{5,4}$</td>
<td>0.10</td>
<td>0.09</td>
<td>5.31</td>
<td>0.10</td>
<td>0.09</td>
<td>24.05</td>
<td>0.07</td>
<td>0.09</td>
<td>1507.12</td>
</tr>
<tr>
<td>$B_{6,1}$</td>
<td>0.02</td>
<td>0.08</td>
<td>11.25</td>
<td>0.02</td>
<td>0.09</td>
<td>168.78</td>
<td>0.02</td>
<td>0.08</td>
<td>3585.38</td>
</tr>
<tr>
<td>$B_{6,2}$</td>
<td>0.09</td>
<td>0.10</td>
<td>6.11</td>
<td>0.08</td>
<td>0.10</td>
<td>76.07</td>
<td>0.09</td>
<td>0.10</td>
<td>1112.23</td>
</tr>
<tr>
<td>$B_{6,3}$</td>
<td>-0.42</td>
<td>0.08</td>
<td>10.29</td>
<td>-0.41</td>
<td>0.07</td>
<td>59.29</td>
<td>-0.39</td>
<td>0.08</td>
<td>2769.35</td>
</tr>
<tr>
<td>$B_{6,4}$</td>
<td>-0.54</td>
<td>0.06</td>
<td>4.02</td>
<td>-0.54</td>
<td>0.06</td>
<td>45.81</td>
<td>-0.54</td>
<td>0.06</td>
<td>1020.02</td>
</tr>
<tr>
<td>$B_{6,5}$</td>
<td>0.12</td>
<td>0.09</td>
<td>4.26</td>
<td>0.12</td>
<td>0.09</td>
<td>29.85</td>
<td>0.12</td>
<td>0.09</td>
<td>1505.24</td>
</tr>
<tr>
<td>$B_{7,1}$</td>
<td>0.35</td>
<td>0.06</td>
<td>5.61</td>
<td>0.35</td>
<td>0.07</td>
<td>48.23</td>
<td>0.37</td>
<td>0.06</td>
<td>1073.97</td>
</tr>
<tr>
<td>$B_{7,2}$</td>
<td>0.38</td>
<td>0.11</td>
<td>4.55</td>
<td>0.37</td>
<td>0.11</td>
<td>43.72</td>
<td>0.36</td>
<td>0.10</td>
<td>914.44</td>
</tr>
<tr>
<td>$B_{7,3}$</td>
<td>-0.03</td>
<td>0.06</td>
<td>3.87</td>
<td>-0.03</td>
<td>0.06</td>
<td>29.72</td>
<td>-0.03</td>
<td>0.06</td>
<td>1728.41</td>
</tr>
<tr>
<td>$B_{7,4}$</td>
<td>0.00</td>
<td>0.07</td>
<td>4.13</td>
<td>0.00</td>
<td>0.08</td>
<td>50.99</td>
<td>0.03</td>
<td>0.06</td>
<td>4794.97</td>
</tr>
<tr>
<td>$B_{7,5}$</td>
<td>-0.39</td>
<td>0.11</td>
<td>3.62</td>
<td>-0.39</td>
<td>0.11</td>
<td>48.57</td>
<td>-0.39</td>
<td>0.11</td>
<td>1138.31</td>
</tr>
<tr>
<td>$B_{8,1}$</td>
<td>-0.58</td>
<td>0.09</td>
<td>7.99</td>
<td>-0.59</td>
<td>0.09</td>
<td>100.25</td>
<td>-0.57</td>
<td>0.09</td>
<td>1599.24</td>
</tr>
<tr>
<td>$B_{8,2}$</td>
<td>0.11</td>
<td>0.12</td>
<td>5.50</td>
<td>0.12</td>
<td>0.12</td>
<td>69.59</td>
<td>0.11</td>
<td>0.12</td>
<td>1928.35</td>
</tr>
<tr>
<td>$B_{8,3}$</td>
<td>0.01</td>
<td>0.09</td>
<td>10.63</td>
<td>0.00</td>
<td>0.09</td>
<td>85.08</td>
<td>-0.02</td>
<td>0.09</td>
<td>4632.40</td>
</tr>
<tr>
<td>$B_{8,4}$</td>
<td>0.55</td>
<td>0.08</td>
<td>4.20</td>
<td>0.55</td>
<td>0.08</td>
<td>47.34</td>
<td>0.56</td>
<td>0.08</td>
<td>1188.44</td>
</tr>
<tr>
<td>$B_{8,5}$</td>
<td>-0.16</td>
<td>0.11</td>
<td>2.38</td>
<td>-0.17</td>
<td>0.11</td>
<td>38.39</td>
<td>-0.16</td>
<td>0.11</td>
<td>2986.94</td>
</tr>
<tr>
<td>$B_{9,1}$</td>
<td>-0.10</td>
<td>0.07</td>
<td>8.47</td>
<td>-0.10</td>
<td>0.07</td>
<td>83.03</td>
<td>-0.12</td>
<td>0.07</td>
<td>3921.19</td>
</tr>
<tr>
<td>$B_{9,2}$</td>
<td>-0.59</td>
<td>0.11</td>
<td>6.31</td>
<td>-0.58</td>
<td>0.11</td>
<td>53.51</td>
<td>-0.57</td>
<td>0.11</td>
<td>1250.48</td>
</tr>
<tr>
<td>$B_{9,3}$</td>
<td>0.27</td>
<td>0.06</td>
<td>3.62</td>
<td>0.27</td>
<td>0.06</td>
<td>79.24</td>
<td>0.28</td>
<td>0.05</td>
<td>917.19</td>
</tr>
<tr>
<td>$B_{9,4}$</td>
<td>-0.11</td>
<td>0.07</td>
<td>4.36</td>
<td>-0.11</td>
<td>0.07</td>
<td>66.67</td>
<td>-0.11</td>
<td>0.07</td>
<td>909.41</td>
</tr>
<tr>
<td>$B_{9,5}$</td>
<td>0.00</td>
<td>0.10</td>
<td>4.52</td>
<td>0.01</td>
<td>0.08</td>
<td>44.00</td>
<td>0.02</td>
<td>0.09</td>
<td>2835.37</td>
</tr>
<tr>
<td>$B_{10,1}$</td>
<td>0.00</td>
<td>0.05</td>
<td>9.33</td>
<td>0.00</td>
<td>0.05</td>
<td>106.10</td>
<td>0.01</td>
<td>0.05</td>
<td>4926.01</td>
</tr>
<tr>
<td>$B_{10,2}$</td>
<td>-0.05</td>
<td>0.08</td>
<td>5.18</td>
<td>-0.06</td>
<td>0.08</td>
<td>47.93</td>
<td>-0.06</td>
<td>0.08</td>
<td>2366.47</td>
</tr>
<tr>
<td>$B_{10,3}$</td>
<td>0.17</td>
<td>0.05</td>
<td>3.64</td>
<td>0.17</td>
<td>0.05</td>
<td>61.71</td>
<td>0.17</td>
<td>0.05</td>
<td>1003.02</td>
</tr>
<tr>
<td>$B_{10,4}$</td>
<td>-0.02</td>
<td>0.05</td>
<td>4.35</td>
<td>-0.03</td>
<td>0.06</td>
<td>53.08</td>
<td>0.01</td>
<td>0.05</td>
<td>4923.15</td>
</tr>
<tr>
<td>$B_{10,5}$</td>
<td>-0.43</td>
<td>0.09</td>
<td>5.44</td>
<td>-0.45</td>
<td>0.09</td>
<td>34.40</td>
<td>-0.44</td>
<td>0.09</td>
<td>2611.21</td>
</tr>
</tbody>
</table>

most of them well above 1000, but more importantly, the exploration is highly dependent on the starting value of the chain. We illustrate this in detail with the Ireland (2004)
model in section 4. On the other hand, the effect of the starting value has no material
effect on the TaRB-MH sampler as it wears off within 100 to 200 iterations.

3.1.1 Computational burden

All three MCMC schemes were coded in Gauss 8.0 and executed on a Windows Vista
64-bit machine with a 2.66 GHz Intel Core2 architecture CPU. The TaRB-MH algorithm
took roughly 30 hours to generate 11,000 draws and the FBTaRB-MH algorithm about
24 hours of CPU time. In comparison, the RW-MH algorithm needed around 3 hours to
complete 1.25 million iterations but the time to find the posterior mode before the run
could start was close to 24 hours. The latter was the time that it took for one of several
threads of the optimizer to locate an approximate mode and an invertible Hessian for the
RW increment. Considering the significant efficiency gains, which are obtained without
the same tuning costs, investment in the TaRB-MH algorithm is well justified.

3.2 Extension to multi-modal problems

In this section we propose a simple extension to our algorithm, still in the same class of
tailored MH methods, that can be used to efficiently sample multi-modal distributions.
Before that, consider once again the RW-MH algorithm. As mentioned earlier, one
strategy for the RW-MH algorithm is to thoroughly scan the parameter space for the
various modes and then start the RW chain at the highest one. The problems with this
approach are twofold. One is that the mass in the various modal regions is not known
apriori. The other, more pronounced issue, concerns truly multi-modal distributions. In
this regard, a suitable version of the RW-MH algorithm that is often used is an intuitive
modification of the regular RW-MH algorithm. The idea is to propose occasional large
jumps that break the attraction to the current mode. This method, however, has limited
applicability and often works well only in small dimensional problems.

To describe our modification of the TaRB-MH algorithm, consider for simplicity the
case where there are only two well defined modes. We assume that the modal values
have been found by initial optimization. Let the location of the two modes be $\mu_1$ and
$\mu_2$. Also, let $V_1$ and $V_2$ denote the inverse of the negative Hessian matrices at the two
modes. In addition to the proposal densities that are calculated within the TaRB-MH
algorithm, define the following mixture proposal

\[ q(\theta|y) = p \, t(\theta|\mu_1, V_1, \nu_1) + (1 - p) \, t(\theta|\mu_2, V_2, \nu_2), \]

where \( p \) is the probability of drawing a value from a \( t \) density with \( \nu_1 \) degrees of freedom that is centered at \( \mu_1 \) and variance given by \( V_1 \). We now incorporate this mixture proposal in our regular TaRB-MH algorithm in the following way. Once every few (say a 100) iterations we update all the parameters in one block using this mixture proposal. That is, we first draw a \( \theta^\dagger \) from the \( t \) density centered at the first mode with probability \( p \) (from the \( t \) density centered at the first mode with probability \( 1 - p \)). This draw is subsequently accepted as the next value of the chain with the probability of move \( \alpha_{MJ}(\theta, \theta^\dagger|y) \) given by

\[
\alpha_{MJ}(\theta, \theta^\dagger|y) = \min \left\{ \frac{f(y|\theta^\dagger) \pi(\theta^\dagger)}{f(y|\theta) \pi(\theta)} \frac{p \, t(\theta|\mu_1, V_1, \nu_1) + (1 - p) \, t(\theta|\mu_2, V_2, \nu_2)}{p \, t(\theta^\dagger|\mu_1, V_1, \nu_1) + (1 - p) \, t(\theta^\dagger|\mu_2, V_2, \nu_2)}, 1 \right\}. \tag{9}
\]

The next (say) 100 draws of the chain are obtained from the TaRB-MH algorithm. At that point one returns to the mode-jumping step and the whole process is repeated.

It is interesting to note that the probability of move could also be defined in terms of the component of the proposal density from which \( \theta^\dagger \) is drawn, as in the Hybrid MCMC algorithm described in Geweke (2005, Section 4.6). The latter scheme is implemented in the rejoinder section of An and Schorfheide (2007) but there is one crucial difference between the two approaches. In our approach, the mode jumping step is embedded within the multiple block TaRB-MH framework whereas the scheme implemented in the rejoinder section of An and Schorfheide (2007) operates on the entire set of parameters in a single block. This difference is important because sampling the parameters in one block is inefficient for making moves within a give mode, as we have argued in this paper. As a result, our mode-jumping algorithm is more effective as we show in Section 4.2 in the context of their example.

In summary, the TaBMJ M-H algorithm has the following form.

**Algorithm: TaRB-MJ-MH algorithm**

**Step 1** Initialize \( \theta^{(0)} \in S_L \cap S_\Omega \cap S_D \) and fix \( n_0 \) (the burn-in) and \( M \) (the MCMC sample size).
Step 2a If not a mode-jumping iteration, then

**Step i** Randomly generate blocks \( (\theta_{j,1}, \theta_{j,2}, \ldots, \theta_{j,p_j}) \)

**Step ii** For \( l = 1, \ldots, p_j \), sample \( \theta_{j,l}^{\dagger} \) from the tailored proposal density \( t(\theta_{j,l}|\hat{\theta}_{j,l}, V_{j,l}, \nu) \) and accept this proposal with the probability of move \( \alpha_l(\theta_{j,l}, \theta_{j,l}^{\dagger}|\theta_{j,-l}, y) \)

Step 2b If a mode-jumping iteration, then

with probability \( p \) draw \( \theta^{\dagger} \) from the proposal density \( t(\theta|\mu_1, V_1, \nu_1) \) and with probability \( (1-p) \) draw \( \theta^{\dagger} \) from the proposal density \( t(\theta|\mu_2, V_2, \nu_2) \) and accept proposal with the probability of move \( \alpha_{MJ}(\theta, \theta^{\dagger}|y) \)

**Step 3** Repeat Steps 2a-2b \( n_0 + M \) times, discard the draws from the first \( n_0 \) iterations and save the subsequent \( M \) draws \( \theta^{(n_0+1)}, \ldots, \theta^{(n_0+M)} \)

The generalization of this method to more than two modes is straightforward. If there are \( M \) distinct modes, denoted by \( \mu_k, k = 1, 2, \ldots, M \), then one could let

\[
q(\theta|y) = \sum_{k=1}^{M} p_k t(\theta|\mu_k, V_k, \nu_k); \quad \sum_k p_k = 1
\]

and embed this proposal into the TaRB-MH algorithm as Step 2b in the preceding algorithm.

To illustrate our approach, consider a target function given by the bivariate mixture normal distribution

\[
\pi(\theta|y) = cN(\mu_1, \Sigma_1) + (1-c)N(\mu_2, \Sigma_2)
\]

where \( \mu_1 = [1, -1]' \), \( \mu_2 = 6\mu_1 \), \( \Sigma_1 = \text{diag}(1.3, 1.3) \), \( \Sigma_2 = \text{diag}(0.05, 0.05) \) and \( c = 0.99 \). The motivation for the specific choice of \( c \) comes from the posterior distribution in the An and Schorfheide (2007) model. The significance of \( c \) is that the difference in the ordinates at the two modes is roughly 1.34 in the log scale. Although this is a simple problem in two dimensions, the challenge arises because of the tiny second mode in the tails of an otherwise unimodal distribution. For this problem, we initialize the TaRBMJ-MH algorithm at the low mode. The mode jumping step is proposed every 100 iterations with \( p = 0.5 \). The simulation results from 20000 draws are shown in Figure 2.
Figure 2: Sampling from a bivariate contaminated normal using the TaRBMJ-MH algorithm: The top panel plots the actual draws (without any burn-ins) against time for the two variables. The bottom left panel shows a bivariate scatter plot of the draws. The actual target function is the 3-dimensional surface plot in the bottom right panel.

The proposal densities used for this experiment are both student t-distributions, with 5 degrees of freedom for the regular TaRB-MH algorithm and 30 degrees of freedom for the mode jumping step. For the sake of illustration we plotted all 20000 draws without any burn-ins to show when the initial jump from the low to the high mode occurred. The top row plots the time series of the draws for the two random variables. Note that the sampler rightly revisits the low mode roughly 1% of the time. The bottom row includes a bivariate scatter plot of the draws (left) along with a 3 dimensional surface plot of the actual density.

For a second example we construct a two component mixture of six dimensional normals. We pick the values for the first mode $\mu_1$ to equal the low modal values of six of the parameters that differ widely across the two modes in the An-Schorfheide (2007) model (discussed in Section 4). To make the problem interesting we center the second
mode \( \mu_2 \) at 15 times \( \mu_1 \). Thus, the two modal regions are widely separated.

\[
\mu_1 : (1.41, 0.81, 0.49, 0.80, 1.07, 0.30) \\
\mu_2 : (21.15, 12.15, 7.35, 12.00, 16.05, 4.50)
\]

We also match the variance-covariance matrices \( \Sigma_1 \) and \( \Sigma_2 \) to the reduced variance-covariance matrices at the two modes in the aforementioned model for the relevant parameters. The specific values are given by

\[
\Sigma_1 = 
\begin{bmatrix}
0.0885 & -0.0023 & 0.0077 & 0.0041 & -0.0229 & -0.0025 \\
-0.0023 & 0.0055 & 0.0015 & 0.0028 & 0.0013 & 0.0001 \\
0.0077 & 0.0015 & 0.0031 & 0.0018 & -0.0011 & -0.0002 \\
0.0041 & 0.0028 & 0.0018 & 0.0029 & 0.0004 & -0.0012 \\
-0.0229 & 0.0013 & -0.0011 & 0.0004 & 0.0169 & 0.0004 \\
-0.0025 & 0.0001 & -0.0002 & -0.0012 & 0.0004 & 0.0024
\end{bmatrix}
\]

\[
\Sigma_2 = 
\begin{bmatrix}
0.1365 & -0.0009 & 0.0063 & 0.0075 & -0.0119 & -0.0101 \\
-0.0009 & 0.0082 & 0.0021 & 0.0048 & 0.0045 & -0.0049 \\
0.0063 & 0.0021 & 0.0020 & 0.0018 & 0.0011 & -0.0018 \\
0.0075 & 0.0048 & 0.0018 & 0.0137 & 0.0042 & -0.0078 \\
-0.0119 & 0.0045 & 0.0011 & 0.0042 & 0.0153 & -0.0051 \\
-0.0101 & -0.0049 & -0.0018 & -0.0078 & -0.0051 & 0.0096
\end{bmatrix}
\]

Finally, we assign 20\% probability to the first mode. Our target distribution takes the form

\[
\pi(\theta|y) = 0.2N(\mu_1, \Sigma_1) + 0.8N(\mu_2, \Sigma_2)
\]

The results from 25000 iterations of the TaRB-MH algorithm are summarized in Figure 3. Rows 1 and 3 plot the time series of the actual draws for the first three parameters (the results are identical for all six parameters). Notice the fluid movement of the sampler across the two modes. We also plot the kernel smoothed histogram of the draws in rows 2 and 4. These plots suggest an accurate replication of the target density.

### 3.3 Marginal likelihood computation

One of the advantages of the Bayesian framework is that it is possible to compare competing models in a formal way through marginal likelihoods and Bayes factors. In this section we show how the framework of Chib (1995), and its M-H version in Chib and Jeliazkov (2001), can be modified to accommodate the randomized block approach proposed in this paper.
Figure 3: Sampling from a two component mixture of 6-variate normals using the TaRBMJ-MH algorithm: The top row shows the sampled draws against time (again without any burn-ins). The simulated marginal density (kernel smoothed histogram) plots for the variables are shown in the bottom row.

In the Chib framework, one starts by noting that the marginal likelihood (which is integral of the sampling density with respect to the prior density) can be expressed in the form

\[ m(y) = \frac{f(y|\theta)\pi(\theta)}{\pi(\theta|y)}, \]

(11)

by a rearrangement of Bayes theorem. Importantly, this is an identity in \( \theta \). Thus, the marginal likelihood can be obtained by evaluating this identity at a value \( \theta^* \) (say), which is usually taken to be a high density point in the support of the posterior. Clearly, the first two components in the latter expression are readily computed. As for the third component, we follow the approach developed in Chib (1995) and decompose the posterior ordinate into marginal and conditional components each of which is then estimated in turn.
In order to apply this approach we fix the number of blocks in the posterior ordinate estimation step to the average number of blocks (say $B$) that are realized in the TaRB-MH run. Then, we construct the blocks $\theta_1, \ldots, \theta_B$ with each block consisting of randomly chosen components from $\theta$ and write

$$\pi(\theta^*|y) = \pi(\theta^*_1|y)\pi(\theta^*_2|y, \theta^*_1) \cdots \pi(\theta^*_B|y, \theta^*_1, \ldots, \theta^*_{B-1}).$$

(12)

In this decomposition, the typical ordinate is $\pi(\theta^*_l|y, \theta^*_1, \ldots, \theta^*_{l-1})$, $l = 1, \ldots, B$, which can now be estimated in a manner akin to that in Chib (1995) by utilizing the output from appropriate reduced runs in which the parameters in $\theta_l$ form a fixed block, and the remaining free parameters form random blocks.

Specifically, following Chib and Jeliazkov (2001), the simulation-consistent estimate of the $l$th ordinate is obtained as

$$\hat{\pi}(\theta^*_l|y, \theta^*_1, \ldots, \theta^*_{l-1}) = \frac{n_1^{-1} \sum n_1 \alpha(\theta^*_l|\theta^*_1, \Psi^*_{l-1}, \Psi^{l+1}(g)) q_l(\theta^*_l|\Psi^*_{l-1}, \Psi^{l+1}(g), y)}{n_1^{-1} \sum n_1 \alpha(\theta^*_l, \theta^*_l|\Psi^*_{l-1}, \Psi^{l+1}(g))}$$

(13)

where $\Psi^*_{l-1} = (\theta_1, \ldots, \theta_{l-1}), \Psi^{l+1} = (\theta_{l+1}, \ldots, \theta_B)$, and $\alpha(\theta^*_l, \theta^*_l|\Psi^*_{l-1}, \Psi^{l+1}(g)), q_l(\theta^*_l|\Psi^*_{l-1}, \Psi^{l+1}(g), y)$ and $\alpha(\theta^*_l, \theta^*_l|\Psi^*_{l-1}, \Psi^{l+1}(g))$ are obtained from (6)-(7) with $\theta_{-l} \equiv (\Psi^*_{l-1}, \Psi^{l+1})$.

In the preceding expression, the average in the numerator is with respect to draws from the conditional distribution $\pi(\theta^*_l, \Psi^{l+1}|\Psi^*_{l-1})$. Accordingly, the draws $\{\theta^*_l, \Psi^{l+1}(g)\}$ in the numerator are obtained from the $l$th stage reduced MCMC run, in which the parameters in the preceding blocks, collected in $\Psi^*_{l-1}$, are held fixed at $\Psi^*_{l-1}$. Furthermore, the sampling in each of these reduced runs is implemented by the TaRB-MH algorithm, with the parameters in $\theta_l$ forming a fixed block, and the parameters in $\Psi^{l+1}$ forming random blocks. As in the main run, this randomization serves to increase the efficiency of the sampling in the reduced runs.

On the other hand, the average in the denominator is taken with respect to the conditional product measure $\pi(\Psi^{l+1}|\Psi^*_{l-1}) q_l(\theta_l|\Psi^*_{l-1}, \Psi^{l+1})$. Conveniently, draws from $\pi(\Psi^{l+1}|\Psi^*_{l-1})$ are available from the calculation of the $(l+1)$st stage numerator which are supplemented with a drawing of $\theta_l$ from $q_l(\theta_l|\Psi^*_{l-1}, \Psi^{l+1})$. In the final stage, we draw $n_1$ values of $\theta_B$ from $q_B(\theta_B|\Psi^*_{B-1}, \Psi^{l+1})$ and compute the denominator average based on these draws.
Having estimated $\pi(\theta^*|y)$, the log of the marginal likelihood is then available as

$$\log \hat{m}(y) = \log f(y|\theta^*) + \log \pi(\theta^*) - \sum_{l=1}^{B} \log \hat{\pi}(\theta^*_l|y, \theta^*_1, \ldots, \theta^*_{l-1}).$$  \hspace{0.5cm} (14)

We summarize this readily implementable calculation in algorithmic form as follows.

**Algorithm: Marginal likelihood**

**Step 1** Partition $\theta$ into $B$ blocks, each block consisting of randomly chosen elements from $\theta$.

**Step 2** In the $l$th stage, $l = 1, \ldots, B$, repeat $n_1$ times:

(a) Generate $\theta_l^{(g)}, \Psi^{l+1,(g)}$ from $\pi(\theta_l, \Psi^{l+1}|y, \Psi^*_l)$ by the TaRB-MH algorithm, randomizing only over the parameters in $\Psi^{l+1}$, and calculate the $l$th stage numerator summand in (13).

(b) Supplement the preceding draw with a draw $\theta^{(g)}_{l-1}$ from $q_{l-1}(\theta_l-1|y, \Psi^*_l-2, \Psi^{l,(g)})$ and calculate the $(l-1)$st stage denominator summand in (13).

(c) Store these values.

**Step 3** For the final stage, draw $n_1$ values $\{\theta_B\}$ from $q_B(\theta_B|y, \Psi^*_B-1)$ and compute the $l$th stage denominator in (13).

**Step 4** Calculate the log of the marginal likelihood as in (14).

We complete the calculation of the marginal likelihood by finding the numerical standard error of the estimate which we get from the approach given in Chib (1995). As we show in our applications, the numerical standard error of the marginal likelihood estimate is typically low which indicates that, in conjunction with our sampling scheme, this method of estimating the marginal likelihood is reliable and efficient.

**4 Applications**

In this section we apply the methods discussed above to three DSGE models. In the first example (Ireland, 2004), we provide a comparative analysis of the TaRB-MH and
RW-MH algorithms. Subsequently, we illustrate the TaRBMJ-MH algorithm in the context of the model in An and Schorfheide (2007) where the posterior distribution is multi-modal. Finally, we round off this section by applying the TaRB-MH algorithm to the large scale DSGE model in Smets and Wouters (2007).

### 4.1 Ireland (2004) model

The DSGE model in Ireland (2004) is summarized by the following system of equations

\[
\begin{align*}
\hat{x}_t &= \alpha_x \hat{x}_{t-1} + (1 - \alpha_x) E_t \hat{x}_{t+1} - (\hat{r}_t - E_t \hat{r}_{t+1}) + (1 - \omega)(1 - \rho_a) \hat{a}_t \\
\hat{\pi}_t &= \beta \alpha_\pi \hat{\pi}_{t-1} + \beta (1 - \alpha_\pi) E_t \hat{\pi}_{t+1} + \psi \hat{x}_t - \hat{e}_t \\
\hat{g}_t &= \hat{y}_t - \hat{y}_{t-1} + \hat{z}_t \\
\hat{r}_t &= \rho_r \hat{r}_{t-1} + \rho_\pi \hat{\pi}_t + \rho_g \hat{g}_t + \rho_x \hat{x}_t + \epsilon_{r,t}
\end{align*}
\]  

where \( x_t, g_t, \pi_t, r_t \) and \( y_t \) denote output gap, output growth, inflation, nominal interest rate and stochastically detrended output, respectively, and the hats denote log-deviation of the variables from their steady state or average values; \( E_t \) denotes the expectation of the relevant variables by the agents in this economy, formed under rational expectations; finally, \( \hat{a}_t, \hat{e}_t \) and \( \hat{z}_t \) capture exogenous shifts in preferences, costs of production and technology, respectively. These equations, in ascending order, represent a forward looking IS curve, a new Keynesian Phillips curve, growth rate of output, growth rate of output gap and the modified Taylor rule (1993). Further, it is assumed that the exogenous driving processes \( \hat{a}_t, \hat{e}_t \) and \( \hat{z}_t \) evolve independently of one another as

\[
\begin{align*}
\hat{a}_t &= \rho_a \hat{a}_{t-1} + \epsilon_{a,t} \\
\hat{e}_t &= \rho_e \hat{e}_{t-1} + \epsilon_{e,t} \\
\hat{z}_t &= \epsilon_{z,t}
\end{align*}
\]  

The innovations in the model \( \epsilon_t = [\epsilon_{a,t}, \epsilon_{e,t}, \epsilon_{z,t}, \epsilon_{r,t}]' \) are assumed to be distributed as multivariate normal

\[
\epsilon_t \sim N_4(0, \Omega)
\]

where \( \Omega = \text{diag}(\sigma^2_a, \sigma^2_e, \sigma^2_z, \sigma^2_R) \). These equations are derived by log-linearizing the original microfounded model around its deterministic steady state, details of which are
available in Ireland (2004).

There are two other parameters \( z \) and \( \pi \) in the non-linear model that determine the steady-state values of output growth and inflation, respectively. In addition, \( \beta \) determines the steady-state value of the short-term nominal interest rate through the relation \( \bar{r} = z\pi / \beta \). Following Ireland (2004), the values of \( z \), \( \pi \) and \( \beta \) are set to the average levels of output growth, inflation and interest rates in the data. Also, \( \psi \) and \( \rho_r \) are fixed at 0.10 and 1.00, respectively. We collect the remaining 12 parameters of interest in the vector \( \theta \)

\[
\theta = (\omega, \alpha_x, \alpha_\pi, \rho_\pi, \rho_g, \rho_\delta, \rho_\epsilon, \sigma_\alpha, \sigma_\pi, \sigma_z, \sigma_r).
\]

The parameters \((\omega, \alpha_x, \alpha_\pi, \rho_\pi, \rho_g, \rho_\delta)\), where \( \omega, \alpha_x, \alpha_\pi \) are each assumed to be between 0 and 1, and each of \( \rho_\pi, \rho_g \), and \( \rho_\delta \) are greater than 0, may be called the structural parameters. It is also assumed that \((\rho_\delta, \rho_\epsilon)\) each lie between 0 and 1. Recalling the notations introduced in Section 2, we let \( S_L \) denote the subset of \( \mathcal{R}^{12} \) satisfying these linear constraints. Finally, the variance parameters \( \sigma^2_i \) lie in the region \( S_\Omega \) that satisfy the usual positivity and positive definiteness constraints. To represent this model in the canonical form (1) for Sims’ solution method, define

\[
\mathbf{s}_t = \left[ \hat{g}_t, \hat{r}_t, \hat{\pi}_t, \hat{x}_t, \hat{\delta}_t, \hat{\epsilon}_t, \mathbb{E}_t \hat{\pi}_{t+1}, \mathbb{E}_t \hat{x}_{t+1} \right]'
\]

\[
\mathbf{\eta}_t = \left[ \hat{\pi}_t - \mathbb{E}_{t-1} \hat{\pi}_t, \hat{x}_t - \mathbb{E}_{t-1} \hat{x}_t \right]'
\]

and \( \varepsilon_t \) as above.

The data for the fitting are the series of demeaned log-deviations of output growth \( \hat{g}_t \), inflation \( \hat{\pi}_t \), and the short-term nominal interest rate \( \hat{r}_t \) from their steady-states or average values for the period 1980:I to 2003:I, with all three observables measured in decimal units. The resulting measurement equation has the straightforward form

\[
\begin{bmatrix}
\hat{g}_t \\
\hat{\pi}_t \\
\hat{r}_t \\
\end{bmatrix}
= 
\begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
\mathbf{s}_t \\
\end{bmatrix}
\]

(17)

In the following section we turn to the specification of the prior distribution \( \pi(\theta) \) for the parameters in this model.
4.1.1 Prior distribution

As mentioned in Section 1, an informative prior distribution can play an important role in the estimation of DSGE models by adding mass to a priori important regions of the parameter space that are otherwise not captured by the likelihood function, thus smoothing the likelihood function to a certain extent. It is therefore important to construct a prior that is meaningful. Our preferred method for constructing such a prior is to reason in terms of the implied distribution of the data (Chib and Ergashev (2008), Del Negro and Schorfheide (2008)).

Continuing our discussion of the Ireland model, we start by assuming that the parameters are apriori independent. Thus, the joint prior distribution is simply the product of the marginal distribution of each parameter. We now sample parameters from this prior 10,000 times, ensuring that the draws satisfy all the constraints mentioned above. For each draw of the parameters, we solve the model and generate the time series of the outcomes, namely $\hat{g}_t$, $\hat{\pi}_t$ and $\hat{r}_t$. We now check whether this simulated data is reasonable. Specifically, we inspect the median, 5% and 95% quantiles of the implied data. If the low and high quantiles look unreasonably wide or too narrow for any time period, we repeat the process after adjusting some or all of the marginal components of the prior. Note that this procedure of prior elicitation does not involve the real data in any way.

We now summarize a sample prior constructed in this fashion for our illustrative model.

- We assume a beta distribution for the parameters constrained to the $(0, 1)$ interval. Thus, $\omega$, $\alpha_x$, $\alpha_{\pi}$, $\rho_a$ and $\rho_c$ are each assumed to be distributed as

  $$\text{Beta}(\alpha_i, \beta_i)$$

  where the $(\alpha_i, \beta_i)$ are equal to $(3, 12)$, $(3.5, 31.5)$, $(3.5, 31.5)$, $(10, 1.8)$ and $(10, 1.8)$ in the order of the parameters listed above. These values roughly correspond to a mean and standard deviation of $(0.20, 0.10)$, $(0.10, 0.05)$, $(0.10, 0.05)$, $(0.85, 0.10)$ and $(0.85, 0.10)$, respectively.

- Parameters $\rho_x$, $\rho_g$ and $\rho_x$ are each assumed to follow a Gamma distribution

  $$\rho_i \sim G(\alpha_i, \beta_i)$$
with \((\alpha_i, \beta_i)\) being \((9, 30)\), \((9, 30)\) and \((16, 64)\) for \(i = \pi, g\) and \(x\), respectively. This translates to means and standard deviations of \((0.30, 0.10)\), \((0.30, 0.10)\) and \((0.25, 0.0625)\).

- Finally, we suppose a standard Inverse Gamma prior on 10,000 times the variance parameters:

\[
10000 \times \sigma_i^2 \sim IG(\alpha_i, \beta_i)
\]

\((\alpha_i, \beta_i) = (3, 60), (2, 0.08), (2.1, 5.5), (2.06, 0.53)\). This transformation is helpful in specifying an IG prior given the extremely small values of these parameters (relative to the others).

The implied data for each of the variables are shown in Figure 4. As one can see from the figure, this prior implies a maximum (minimum) quarterly deviation of roughly 4.5% \((-4.5\%)\) for output and the rate of interest, and around 5% \((-5\%)\) in the case of inflation. This is understandably a substantial change in the variables when measured on a quarterly basis. From our perspective, though, this indicates a reasonably well dispersed prior distribution on the parameters.

![Figure 4: Implied \(\hat{g}\), \(\hat{\pi}\) and \(\hat{r}\) from the prior distribution of the parameters in Ireland (2004). See Table 4 for a summary of the prior distribution.](image-url)

### 4.1.2 Posterior and MCMC sampling

Before presenting the results from the TaRB-MH algorithm, we focus on the implementation of the RW-MH algorithm for this problem. To illustrate the importance of the starting value of the chain and the variance of the random walk proposal we start three
different RW chains, one at the prior mean, one at a local mode and the third at the dominant mode\(^3\). For the first chain we specify the variance of the random walk proposal to be \(k \times I_{12}\), where \(I_{12}\) is the twelve dimensional identity matrix, whereas for the other two chains we let it be \(k\) times the variance at the respective modes (calculated as the negative inverse of the Hessian). For the second and third chains it was reasonably easy to find a suitable \(k\) (that generated roughly 30% acceptance rates) within a few tries. For the first chain, though, this task proved extremely tedious and time consuming. We started with \(k = 0.5\) and finally stopped at \(k = 0.0001\). Even with such a small variance the acceptance rate was only a disappointing 8.5%, which points to the complex correlation structure between the parameters. A summary of the results from the three chains, each run for 250,000 iterations following a burn-in of 50,000 iterations, are reported in Table 3 along with their respective starting values.

Table 3–Summary of results from three RW-MH chains for the parameters in Ireland (2004) model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Chain I</th>
<th></th>
<th>Chain II</th>
<th></th>
<th>Chain III</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\omega)</td>
<td>0.2000</td>
<td>0.0701</td>
<td>0.0589</td>
<td>0.1498</td>
<td>0.1058</td>
</tr>
<tr>
<td>(\alpha_x)</td>
<td>0.1000</td>
<td>0.0651</td>
<td>0.0612</td>
<td>0.0823</td>
<td>0.0629</td>
</tr>
<tr>
<td>(\alpha_{\pi})</td>
<td>0.1000</td>
<td>0.0825</td>
<td>0.0443</td>
<td>0.0809</td>
<td>0.0605</td>
</tr>
<tr>
<td>(\rho_{\pi})</td>
<td>0.3000</td>
<td>0.6079</td>
<td>0.2934</td>
<td>0.5886</td>
<td>0.5515</td>
</tr>
<tr>
<td>(\rho_g)</td>
<td>0.3000</td>
<td>0.4022</td>
<td>0.3201</td>
<td>0.3722</td>
<td>0.3593</td>
</tr>
<tr>
<td>(\rho_x)</td>
<td>0.2500</td>
<td>0.1825</td>
<td>0.2742</td>
<td>0.1979</td>
<td>0.1760</td>
</tr>
<tr>
<td>(\rho_a)</td>
<td>0.8500</td>
<td>0.9583</td>
<td>0.5179</td>
<td>0.8694</td>
<td>0.9334</td>
</tr>
<tr>
<td>(\rho_e)</td>
<td>0.8500</td>
<td>0.8843</td>
<td>0.8858</td>
<td>0.8838</td>
<td>0.8874</td>
</tr>
<tr>
<td>(10000\sigma_a^2)</td>
<td>30.0000</td>
<td>29.4853</td>
<td>0.3627</td>
<td>4.8860</td>
<td>13.6777</td>
</tr>
<tr>
<td>(10000\sigma_e^2)</td>
<td>0.0800</td>
<td>0.0077</td>
<td>0.0037</td>
<td>0.0066</td>
<td>0.0060</td>
</tr>
<tr>
<td>(10000\sigma_r^2)</td>
<td>5.0000</td>
<td>3.6314</td>
<td>0.4287</td>
<td>0.7947</td>
<td>0.6977</td>
</tr>
<tr>
<td>(10000\sigma_f^2)</td>
<td>0.5000</td>
<td>0.1041</td>
<td>0.1088</td>
<td>0.0967</td>
<td>0.0857</td>
</tr>
</tbody>
</table>

Note: The columns \(\text{start}\) and \(\text{post mean}\) report the starting value and the mean of the sampled values for each of the three chains. Chain I is started at the prior mean with a variance of \(0.0001 \times I_{12}\). Chain II is initiated at a local mode at which the ordinate of the (unnormalised) log posterior is roughly 992.68. The starting value for Chain III is the dominant mode, reaching a height of 1197.30 in the log scale.

A quick glance at the table reveals that sufficient exploration of the target distribution

\(^3\)These modes are found by a standard deterministic optimizer based on the method of steepest decent and through experimentation with the starting values.
is not achieved with 250,000 iterations of chains I and II. In fact, further sampling shows that a million iterations also may not be enough. On the other hand, chain III, that was initiated at the dominant mode, performed quite well. The performance of these chains is better assessed in Figure 5 where we plot the sampled draws over time along with the corresponding sample ACF for a couple of parameters from each of the three chains. As the top row shows, the first and second chains essentially did not converge, especially for $\sigma_a$. Not surprisingly, the sampled draws from these chains are highly persistent. This is in fact true even for chain III, where the ACFs retain significant mass for certain parameters even at lags of 250 and higher.

Figure 5: Time series and autocorrelation function plots for select parameters in the Ireland model from each of the three RW chains. Top left: Chain I, top right: chain II and bottom: chain III.

As might be evident from this exercise, unless one is able to exploit in some way the correlation among the various parameters, designing a sampler that makes large moves across the support of the distribution and quickly navigates to the meaningful regions of the posterior is not straightforward. The process can require substantial pre-run tuning and even so one is not assured of getting reliable results.

The principal goal of our proposed method, as in any MCMC scheme, is to explore
the posterior distribution globally. We would also like the sampler to be efficient, making large moves across the support of the target, while at the same time generating reasonable acceptance rates. On both dimensions the TaRB-MH performed well as can be seen in Table 4. To elaborate on the specifics of these results, we initialized the TaRB-MH chain at the prior mean\(^4\). Notice that unlike the RW-MH algorithm this does not require any sort of pre-run tuning. We also set the initial temperature \(t_0\) in simulated annealing to 2, the rate of temperature decline \(a\) to 0.4, the total number of stages \(K\) to 6, the stage length increment \(b\) to 10. We ran the chain for 10,000 iterations beyond a burn-in of 1000 iterations. We also specified a student-t distribution with \(\nu = 15\) degrees of freedom for the proposal density.

Table 4–Posterior sampling results using the TaRB-MH algorithm for the Ireland (2004) model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Prior Mean</th>
<th>Prior Standard deviation</th>
<th>Posterior Mean</th>
<th>Posterior Numerical S.E.</th>
<th>Posterior 90 percent interval</th>
<th>Inefficiency factors</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\omega)</td>
<td>0.20</td>
<td>0.10</td>
<td>0.1089</td>
<td>0.0010</td>
<td>[0.0381,0.2036]</td>
<td>5.2791</td>
</tr>
<tr>
<td>(\alpha_x)</td>
<td>0.10</td>
<td>0.05</td>
<td>0.0778</td>
<td>0.0006</td>
<td>[0.0186,0.1669]</td>
<td>2.7625</td>
</tr>
<tr>
<td>(\alpha_\pi)</td>
<td>0.10</td>
<td>0.05</td>
<td>0.0807</td>
<td>0.0009</td>
<td>[0.0184,0.1819]</td>
<td>4.9731</td>
</tr>
<tr>
<td>(\rho_\pi)</td>
<td>0.30</td>
<td>0.10</td>
<td>0.5522</td>
<td>0.0023</td>
<td>[0.3341,0.7767]</td>
<td>4.1913</td>
</tr>
<tr>
<td>(\rho_g)</td>
<td>0.30</td>
<td>0.10</td>
<td>0.3747</td>
<td>0.0011</td>
<td>[0.2751,0.4867]</td>
<td>3.9146</td>
</tr>
<tr>
<td>(\rho_x)</td>
<td>0.25</td>
<td>0.0625</td>
<td>0.2001</td>
<td>0.0016</td>
<td>[0.1108,0.3134]</td>
<td>9.2058</td>
</tr>
<tr>
<td>(\rho_a)</td>
<td>0.85</td>
<td>0.10</td>
<td>0.9310</td>
<td>0.0008</td>
<td>[0.8814,0.9662]</td>
<td>15.013</td>
</tr>
<tr>
<td>(\rho_e)</td>
<td>0.85</td>
<td>0.10</td>
<td>0.8674</td>
<td>0.0016</td>
<td>[0.7582,0.9555]</td>
<td>9.7198</td>
</tr>
<tr>
<td>(10000\sigma_2^2)</td>
<td>30.00</td>
<td>30.00</td>
<td>15.7994</td>
<td>0.3784</td>
<td>[6.0171,38.228]</td>
<td>15.814</td>
</tr>
<tr>
<td>(10000\sigma_e^2)</td>
<td>0.08</td>
<td>1.00</td>
<td>0.0068</td>
<td>0.0000</td>
<td>[0.0041,0.0107]</td>
<td>6.2913</td>
</tr>
<tr>
<td>(10000\sigma_\gamma^2)</td>
<td>5.00</td>
<td>15.00</td>
<td>0.7633</td>
<td>0.0030</td>
<td>[0.4785,1.1145]</td>
<td>3.1988</td>
</tr>
<tr>
<td>(10000\sigma_\eta^2)</td>
<td>0.50</td>
<td>2.00</td>
<td>0.0969</td>
<td>0.0005</td>
<td>[0.0635,0.1443]</td>
<td>6.3380</td>
</tr>
</tbody>
</table>

Remark: The results reported in this table are based on prior mean as the starting value. However, the results are insensitive to this choice.

The first point to note is that, unlike in the RW-MH algorithm, the sampling results in the case of the TaRB-MH algorithm are virtually identical regardless of where the chain is initialized. Second, the sampler navigates to the same region of the parameter space within a few iterations. Specifically, a TaRB-MH chain initialized at the prior mean climbs to the main regions of the posterior distribution within 30 iterations, whereas one initialized at the local mode corresponding to chain II in the RW-MH algorithm

\(^4\)The results reported here are virtually identical regardless of where the chain in initialized
Figure 6: Sampling results for the Ireland (2004) model using the TaRB-MH algorithm: Marginal prior-posterior plots and autocorrelation functions for the structural and autoregressive parameters.

... takes less than 10 iterations with the same settings for the SA parameters. This is not surprising given that information on both the location of the mode for a given block, as well as the correlation among the parameters in that block, is utilized in the
Figure 6: Sampling results for the Ireland (2004) model using the TaRB-MH algorithm (cont’d): Marginal prior-posterior plots and autocorrelation functions for the variances of the shock parameters.

sampling. Consequently, the autocorrelation among the draws decays quickly to zero. This translates into an efficient sampling scheme, one in which the Markov chain can be expected to converge quickly to its stationary distribution. In terms of the inefficiency factors discussed in section 3, whereas in the best case scenario of the RW-MH algorithm (refer to chain III in Table 3) the inefficiency factors range from 60 and 230, those for the TaRB-MH algorithm are between 2.5 and 15.\(^5\) Thus, in some cases the TaRB-MH algorithm performs close to an independence sampler.

The extent (or lack thereof) of the serial correlation can also be observed through a plot of the autocorrelation functions of the sampled draws. We plot these functions in Figure 6. These plots show that the serial correlations among the draws decay quickly to zero for all the parameters. In contrast, the serial correlations from the best performing RW-MH chain III, are considerably more persistent. This indicates that the mixing properties of our sampler are markedly better than those of the RW-MH algorithm. This figure also includes the kernel smoothed histogram of the marginal posterior

\(^5\)The inefficiency factors are computed for a maximum lag of \(L = 250\).
superimposed on the prior distribution of the parameters. As can be observed, for most parameters, the likelihood function carries information beyond that contained in the prior.

4.1.3 Marginal likelihood

In Section 3.2 we showed how the marginal likelihood can be estimated within the Chib (1995) framework when sampling is done with our tailored randomized block method. In this section, we apply that method to the Ireland model. Recall that the marginal likelihood computation requires decomposing the posterior ordinate into marginal-conditional densities. Unlike the fixed block case discussed in Chib and Jeliazkov (2001), there is some freedom here in selecting the number of blocks (stages), as well as the components of these blocks. Therefore, it is worthwhile to consider the marginal likelihood estimate, and the numerical standard errors that result, as the number of stages (and the sample size $n_1$ in the reduced MCMC runs) are varied. Accordingly, we consider both two and three stage schemes for calculating the posterior ordinate. We consider these many stages because the average number of blocks in the full MCMC run for this model is around 3. We also compare our results to the estimate of the marginal likelihood when sampling is done with the help of the RW-MH algorithm.

The results are summarized in Table 5. For the two stage and three stage computations under the TaRB-MH algorithm, we calculate the marginal likelihood from 5,000, 10,000 and 15,000 draws in the reduced runs. Under the single block RW-MH scheme, the computations are carried out in 1-stage, with the draws in the numerator of equation (13) being those from the posterior sampling and those in the denominator being those from $N(\theta^*, c\Sigma)$, where $\Sigma$ is the variance matrix computed at the posterior mode. In this calculation we use the output from our most efficient implementation of the RW-MH algorithm (chain III in the preceding section) and illustrate the results for samples of sizes 75,000, 150,000 and 250,000.

It is noteworthy that the numerical standard error of the marginal likelihood estimate (which are reported in parentheses in the table) are lower under the more efficient sampling scheme. This is in keeping with the discussion in Chib and Jeliazkov (2001). As one can see from the table, the numerical standard errors under the less efficient
Table 5—Log marginal likelihood estimates (with numerical standard errors) for the Ireland (2004) model based on the outputs from the TaRB-MH and RW-MH algorithms

<table>
<thead>
<tr>
<th></th>
<th>TaRB-MH</th>
<th>RW-MH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2 stage</td>
<td>3 stage</td>
</tr>
<tr>
<td>$n_1$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5,000</td>
<td>1170.08 (0.0324)</td>
<td>1170.26 (0.0400)</td>
</tr>
<tr>
<td>10,000</td>
<td>1170.18 (0.0268)</td>
<td>1170.29 (0.0302)</td>
</tr>
<tr>
<td>15,000</td>
<td>1170.15 (0.0216)</td>
<td>1170.33 (0.0250)</td>
</tr>
</tbody>
</table>

RW-MH algorithm are between 15 and 20 times higher than those under the TaRB-MH algorithm, for any sample length. Furthermore, in this case, the estimates (for different reduced MCMC sample sizes) from the the 2-stage run are marginally more efficient than those from the 3-stage run.

4.2 An and Schorfheide (2007) model

While in many problems the posterior is relatively well defined in that there is a unique dominant mode, there can be instances where the posterior is (at least seemingly) multi-modal. In the realm of DSGE models, one such interesting example is demonstrated by An and Schorfheide (2007) in a 13-parameter model. In this model, the two distinct modes that show up during optimization appear to be separated by a deep valley, posing a severe impediment for the RW-MH algorithm. In fact, the authors show that the RW sampler was simply unable to escape the attraction of the individual modes even when the sampler was run for a million iterations following a burn-in of 100,000 iterations. What is especially interesting is that, despite a difference of approximately 8 in the log scale between the unnormalized posterior ordinates at the two modes, the RW chain does not cross over from the low modal region to the high modal region. Since the probability content in each of these separated regions is not known apriori, it is not possible to approximate the posterior distribution globally by running multiple chains of the RW-MH algorithm.

Before presenting the model in An and Schorfheide (2007), it is worth mentioning that the TaRB-MH algorithm proposed in the previous section is far less susceptible to
such problems. The first point to note is that a purely randomized block version of the TaRB-MH sampler quickly navigates to the high modal region when initialized at the prior mean with occasional visits to the low modal region. In contrast, a RW-MH chain initiated at the same value converges to the low modal region. More importantly, even when the TaRB-MH algorithm is started at the low mode, it does not get trapped in the region. The simple strategy employed here exploits the fact that six of the parameters differ significantly across the two modes. Thus, by grouping these six parameters in one block, and the remaining seven parameters in another, one is able to move across the two modes quite easily. Nonetheless, the TaRBMJ-MH method is more efficient and easier to implement in the case of complicated multi-modal distributions. For the rest of this section we briefly present the model in An and Schorfheide (2007), henceforth referred to as the AS07 model.

The linearized DSGE model in AS07 is given as follows.

\[ \hat{g}_t = \mathbb{E}_t \hat{g}_{t+1} + \hat{g}_t - \mathbb{E}_t \hat{g}_{t+1} - \frac{1}{\tau}(\hat{r}_t - \mathbb{E}_t \hat{r}_{t+1} - \mathbb{E}_t \hat{z}_{t+1}) \]  
\[ \hat{r}_t = \beta \mathbb{E}_t \hat{r}_{t+1} + \kappa(\hat{g}_t - \hat{y}_t) \]  
\[ \hat{c}_t = \hat{y}_t - \hat{g}_t \]  
\[ \hat{r}_t = \rho_r \hat{r}_{t-1} + (1 - \rho_r) \psi_1 \hat{r}_t + (1 - \rho_r) \psi_2(\Delta \hat{y}_t + \hat{z}_t) + \varepsilon_{r,t} \]

where \( y_t, \pi_t, r_t \) and \( c_t \) denote output, inflation, nominal interest rate and consumption, respectively, and the hats denote log-deviation of the variables from their steady state or average values. In addition to the interest rate shock \( \varepsilon_{r,t} \) in the Taylor rule above, the system is driven by two other shocks, namely, a consumption shock and a technology shock. The evolution of these exogenous processes is governed by the independent AR(1) processes

\[ \hat{g}_t = \rho_g \hat{a}_{t-1} + \varepsilon_{g,t} \]  
\[ \hat{z}_t = \rho_z \hat{z}_{t-1} + \varepsilon_{z,t} \]

where \( \varepsilon_{i,t} \) is normally distributed with mean 0 and standard deviation \( \sigma_i \). As before, we collect the shocks in the vector \( \varepsilon_t = [\varepsilon_{r,t}, \varepsilon_{g,t}, \varepsilon_{z,t}]' \). Then \( \varepsilon_t \sim \mathcal{N}_3(\mathbf{0}, \Omega) \), where \( \Omega = \text{diag}[\sigma_r^2, \sigma_g^2, \sigma_z^2] \).
As in the previous section, define the terms

\[ s_t = [\hat{\pi}_t, \hat{y}_t, \hat{r}_t, \hat{g}_t, \hat{z}_t, E_t\hat{\pi}_{t+1}, E_t\hat{y}_{t+1}]' \]

and

\[ \eta_t = [\hat{\pi}_t - E_{t-1}\hat{\pi}_t, \hat{y}_t - E_{t-1}\hat{y}_t]' \]

to express the model compactly as in (1). The model is solved, subject to the determinacy constraint, to obtain a stochastic law of motion for the 7-dimensional state vector \( s_t \).

The simulated data used in the fitting are assumed to be the series of quarterly observations on per capita GDP growth rates \( \hat{Y}_t = \gamma Q + 100(\hat{y}_t - \hat{y}_{t-1} + \hat{\pi}_t) \), annualized inflation rates \( \pi_t = \pi^A + 400\hat{\pi}_t \), and annualized nominal interest rate \( r_t = \pi^A + r^A + 4\gamma Q + 400\hat{r}_t \), where \( \gamma Q, r^A, \) and \( \pi^A \) are related to the steady states of the relevant variables. Table 6 contains the parameters of the data generating process, along with the prior distribution, both of which are taken from AS07. It is assumed that the sample size is 80. To account for the lagged output in the data, the state vector is augmented with \( \hat{y}_{t-1} \). The redefined state vector takes the form

\[ s_t = [\hat{\pi}_t, \hat{y}_t, \hat{r}_t, \hat{g}_t, \hat{z}_t, E_t\hat{\pi}_{t+1}, E_t\hat{y}_{t+1}, \hat{y}_{t-1}]' \]

and the resulting SSM is then written as

\[
\begin{bmatrix}
\hat{Y}_t \\
\hat{\pi}_t \\
r_t \\
\hat{y}_t
\end{bmatrix} =
\begin{bmatrix}
\gamma(Q) \\
\pi^A \\
\pi^A + r^A
\end{bmatrix}
\begin{bmatrix}
0 & 1 & 0 & 0 & 1 & 0 & 0 & -1 \\
4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 4 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
s_t \\
\theta
\end{bmatrix}
\]

(24)

\[ s_t = D(\theta)s_{t-1} + F(\theta)\varepsilon_t. \]  

(25)

4.2.1 Sampling results from the TaRBMJ-MH algorithm

From the point of view of this paper, an interesting aspect of this model is the occurrence of possibly multiple modes. Specifically, by maximization of the posterior distribution with respect to the parameters one can find two distinct optima with the optima differing primarily in terms of the components \( (\tau, \psi_2, \rho_r, \rho_z, \sigma_g, \sigma_z) \). The modal values are presented in Table 7. While it is not a simple task to ascertain the importance of the two modal regions in the posterior distribution, one point to note is that the difference
Table 6–DGP and prior distribution for the model parameters in An and Schorfheide (2007)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>DGP</th>
<th>Density</th>
<th>Mean</th>
<th>Standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau$</td>
<td>2.00</td>
<td>Gamma</td>
<td>2.00</td>
<td>0.50</td>
</tr>
<tr>
<td>$\kappa$</td>
<td>0.15</td>
<td>Gamma</td>
<td>0.20</td>
<td>0.10</td>
</tr>
<tr>
<td>$\psi_1$</td>
<td>1.50</td>
<td>Gamma</td>
<td>1.50</td>
<td>0.25</td>
</tr>
<tr>
<td>$\psi_2$</td>
<td>1.00</td>
<td>Gamma</td>
<td>0.50</td>
<td>0.25</td>
</tr>
<tr>
<td>$\rho_r$</td>
<td>0.60</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td>$\rho_g$</td>
<td>0.95</td>
<td>Beta</td>
<td>0.80</td>
<td>0.10</td>
</tr>
<tr>
<td>$\rho_z$</td>
<td>0.65</td>
<td>Beta</td>
<td>0.66</td>
<td>0.15</td>
</tr>
<tr>
<td>$r^A$</td>
<td>0.40</td>
<td>Gamma</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>$\pi^A$</td>
<td>4.00</td>
<td>Gamma</td>
<td>7.00</td>
<td>2.00</td>
</tr>
<tr>
<td>$\gamma^Q$</td>
<td>0.50</td>
<td>Normal</td>
<td>0.40</td>
<td>0.20</td>
</tr>
<tr>
<td>$\sigma_r$</td>
<td>0.20</td>
<td>Inverse Gamma</td>
<td>0.50</td>
<td>0.26</td>
</tr>
<tr>
<td>$\sigma_g$</td>
<td>0.80</td>
<td>Inverse Gamma</td>
<td>1.25</td>
<td>0.65</td>
</tr>
<tr>
<td>$\sigma_z$</td>
<td>0.45</td>
<td>Inverse Gamma</td>
<td>0.63</td>
<td>0.33</td>
</tr>
</tbody>
</table>

Remark: The $IG(\tilde{\alpha}, \tilde{\beta})$ distribution on $\sigma$ is the derived distribution from the assumption that $\sigma^2 \sim IG(\alpha, \beta)$. It is sometimes referred to as the Inverse Gamma type-I distribution.

in the (unnormalized) posterior ordinates at the two modes is around 8.3 in the log scale (-192.12 at the low mode vs. -183.78 at the high mode). We can understand the implication of this difference on the relative probability mass in the two regions by the following simple exercise.

Denote the high mode by $\mu_1$ and the low mode by $\mu_2$. Assume that the underlying posterior distribution can be approximated by a two component mixture of 13-dimensional normals as follows

$$p \mathcal{N}(\theta | \mu_1, \Sigma_1) + (1 - p) \mathcal{N}(\theta | \mu_2, \Sigma_2)$$

where the $\Sigma$s are the negative inverse hessians at the respective modes. Under this working assumption, $p$ is the only unknown. From the knowledge of the ordinates at the two modes one can write

$$p \mathcal{N}(\mu_1 | \mu_1, \Sigma_1) + (1 - p) \mathcal{N}(\mu_1 | \mu_2, \Sigma_2) = \exp\{8\}$$

which yields $p = 0.99995$. This suggests that the mass in the second mode is around 0.005%, which is small enough to be ignorable. Arguably, this calculation is only suggestive of the actual probability of the low modal region. Nonetheless, it leads to the
conjecture that the mass in the low modal region is not substantial as one might intuitively expect from the disparity in the heights of the two modes.

Table 7–Posterior sampling results using the TaRBMJ-MH algorithm for the An-Schorfheide (2007) model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mode 1</th>
<th>Mode 2</th>
<th>Mean</th>
<th>90 percent interval</th>
<th>Inefficiency factors</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau$</td>
<td>2.05</td>
<td>1.41</td>
<td>2.12</td>
<td>[1.04,3.74]</td>
<td>29.91</td>
</tr>
<tr>
<td>$\kappa$</td>
<td>0.16</td>
<td>0.18</td>
<td>0.17</td>
<td>[0.03,0.42]</td>
<td>34.54</td>
</tr>
<tr>
<td>$\psi_1$</td>
<td>1.55</td>
<td>1.57</td>
<td>1.66</td>
<td>[1.12,2.42]</td>
<td>19.88</td>
</tr>
<tr>
<td>$\psi_2$</td>
<td>0.96</td>
<td>0.81</td>
<td>1.00</td>
<td>[0.70,1.35]</td>
<td>36.38</td>
</tr>
<tr>
<td>$\rho_r$</td>
<td>0.59</td>
<td>0.49</td>
<td>0.59</td>
<td>[0.41,0.72]</td>
<td>49.38</td>
</tr>
<tr>
<td>$\rho_g$</td>
<td>0.94</td>
<td>0.97</td>
<td>0.92</td>
<td>[0.79,0.98]</td>
<td>41.57</td>
</tr>
<tr>
<td>$\rho_z$</td>
<td>0.58</td>
<td>0.80</td>
<td>0.54</td>
<td>[0.21,0.83]</td>
<td>31.57</td>
</tr>
<tr>
<td>$r_A$</td>
<td>0.64</td>
<td>0.62</td>
<td>0.68</td>
<td>[0.07,1.43]</td>
<td>5.54</td>
</tr>
<tr>
<td>$\pi_A$</td>
<td>4.06</td>
<td>4.00</td>
<td>4.16</td>
<td>[3.28,5.52]</td>
<td>14.48</td>
</tr>
<tr>
<td>$\gamma_Q$</td>
<td>0.50</td>
<td>0.54</td>
<td>0.48</td>
<td>[0.11,0.80]</td>
<td>8.655</td>
</tr>
<tr>
<td>$\sigma_r$</td>
<td>0.22</td>
<td>0.24</td>
<td>0.23</td>
<td>[0.18,0.32]</td>
<td>11.61</td>
</tr>
<tr>
<td>$\sigma_g$</td>
<td>0.76</td>
<td>1.07</td>
<td>0.76</td>
<td>[0.45,1.33]</td>
<td>28.57</td>
</tr>
<tr>
<td>$\sigma_z$</td>
<td>0.54</td>
<td>0.30</td>
<td>0.61</td>
<td>[0.30,1.01]</td>
<td>38.86</td>
</tr>
</tbody>
</table>

Remark: The results reported in this table are based on the low mode as the starting value. The TaRBMJ-MH algorithm was run for 10,000 iterations, with the mode jumping step using the mixture distribution proposed every 100th iteration.

For sampling the posterior distribution in this model we initialized the TaRBMJ-MH sampler at the low mode.\textsuperscript{6} For the usual TaRB-MH iterations with fully randomized blocks, the parameters for the simulated annealing algorithm were chosen as follows. We set the number of stages $K$ to 10, the increment of stage length $b$ to 15, the initial temperature to 75 and the linear temperature reduction factor $a$ to 0.8. Finally, the degrees of freedom in the $t$-proposal density was set to 2. We also assigned equal probability ($p = 0.5$) to both modes in the mode jumping proposal which was called in every 100th iteration. The mixture proposal used in this step was a $t$-density with 5 degrees of freedom. The sampler was run for 10,000 iterations without any burn-in to observe the movement of the sampler throughout the course of the MCMC iterations. While the acceptance rate is not very sensitive to these settings, it is worth noting that the resulting acceptance rate was around 45% for all the parameters.

\textsuperscript{6}The results from two other chains, one initialized at the dominant mode and the other initialized at the prior mean are virtually identical to those from this chain.
As expected, the transition from the low modal region to the high modal region was almost instantaneous (in that the first time a value from the high mode component of the mixture distribution was proposed it was accepted). This jump occurred at the 200th iteration. However, the reverse transition through the mode jumping step occurred only once in the 300th iteration. Beyond this, any value generated through the low mode component in the mode jumping step got rejected. However, in the non-mode jumping steps, the usual TaRB-MH sampler continued exploring both modal regions efficiently. This points to the potential of this algorithm in the global exploration of the posterior distribution.

The results from the TaRBMJ-MH sampler are summarized in Table 7. This table also contains the modal information of the parameters. Particularly noteworthy in this table is the movement of the sampler across the two modal regions as indicated by the 90 percent intervals (column 6 in table). Also worth recognizing is the fine mixing of the sampler as indicated by the inefficiency factors. The highest inefficiency factor is
around 50, with the average inefficiency factor across all the parameters being 27. These factors combined together suggest an efficient, well mixing sampler. The movement of the sampler across the support of the posterior is better illustrated in Figure 7. The plots show the sampled draws from the posterior distribution for the six parameters that differ across the two modes, together with the respective modal values (indicated by the red lines). As one can see, the sampler visited the low modal region a few times. However, the proportion of draws from this region are not substantial enough to define a distinct mode in the marginal density plots. Overall, this further strengthens our belief that the low modal region has negligible probability content.

Table 8–Posterior sampling results using the mode jumping algorithm in the rejoinder section of An-Schorfheide (2007)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Implementation I</th>
<th>Implementation II</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>90 percent interval</td>
<td>90 percent interval</td>
</tr>
<tr>
<td>$\tau$</td>
<td>2.12 [1.51, 2.90]</td>
<td>2.15 [1.51, 3.02]</td>
</tr>
<tr>
<td>$\kappa$</td>
<td>0.18 [0.09, 0.30]</td>
<td>0.18 [0.09, 0.30]</td>
</tr>
<tr>
<td>$\psi_1$</td>
<td>1.62 [1.28, 2.03]</td>
<td>1.61 [1.29, 2.04]</td>
</tr>
<tr>
<td>$\psi_2$</td>
<td>0.98 [0.81, 1.17]</td>
<td>0.97 [0.80, 1.17]</td>
</tr>
<tr>
<td>$\rho_r$</td>
<td>0.59 [0.49, 0.67]</td>
<td>0.59 [0.49, 0.67]</td>
</tr>
<tr>
<td>$\rho_g$</td>
<td>0.93 [0.88, 0.97]</td>
<td>0.93 [0.89, 0.97]</td>
</tr>
<tr>
<td>$\rho_z$</td>
<td>0.54 [0.30, 0.75]</td>
<td>0.54 [0.32, 0.75]</td>
</tr>
<tr>
<td>$r^A$</td>
<td>0.64 [0.14, 1.20]</td>
<td>0.63 [0.11, 1.19]</td>
</tr>
<tr>
<td>$\pi^A$</td>
<td>4.08 [3.45, 4.83]</td>
<td>4.08 [3.40, 4.77]</td>
</tr>
<tr>
<td>$\gamma^Q$</td>
<td>0.50 [0.26, 0.71]</td>
<td>0.51 [0.27, 0.74]</td>
</tr>
<tr>
<td>$\sigma_r$</td>
<td>0.23 [0.19, 0.28]</td>
<td>0.23 [0.20, 0.28]</td>
</tr>
<tr>
<td>$\sigma_g$</td>
<td>0.75 [0.53, 0.99]</td>
<td>0.74 [0.50, 0.99]</td>
</tr>
<tr>
<td>$\sigma_z$</td>
<td>0.59 [0.42, 0.81]</td>
<td>0.59 [0.41, 0.77]</td>
</tr>
</tbody>
</table>

Remark: Implementation I refers to the settings reported in the rejoinder section of An and Schorfheide (2007). Implementation II refers to the settings mentioned in this paper that result in slightly higher acceptance rates.

A couple of remarks follow.

1. The first is that embedding the mode jumping step within the RW-MH scheme does not result in similar expansive exploration of the posterior distribution. As

5These inefficiency factors are low given the irregularity of the posterior surface in this problem and in relation to what can be achieved without recourse to our sampling scheme.
noted earlier, this is because it is practically infeasible to tune a RW chain to traverse both modal regions. Thus, once in the high modal (low modal) region, the RW sampler fails to move to the low modal (high modal) region by itself. In combination with the fact that the mode jumping step does not lead to frequent transitions to the low modal region means that exploration is limited to mostly the high modal region. In addition, the RW sampler does not explore the high modal region very effectively. Neither problem occurs in our TaRBMJ-MH algorithm. The TaRB-MH steps within that algorithm explore the high-modal region more effectively and by itself produces moves intermittently to the low modal region (quite apart from the moves to that region in the mode-jumping step), thus leading to an exploration of the posterior that is difficult to achieve by any other means.

2. The second remark relates to the mode jumping method discussed in the rejoinder section of An and Schorfheide (2007). In our efforts to replicate their results, we first executed that method with the settings that are reported in the rejoinder. We refer to this as Implementation I. In detail, An and Schorfheide (2007) construct the following mixture density

\[ q(\theta | y) = 0.5 \ t(\theta | \mu_l, c_l, V_l, \nu_l) + 0.5 \ t(\theta | \mu_h, c_h, V_h, \nu_h), \]

where \( \mu_l (\mu_h) \) and \( V_l (V_h) \) denote the low (high) mode and the variance at the low (high) mode, \( c_l = 5, c_h = 4, \nu_l = 3, \) and \( \nu_h = 2. \) Once a proposal \( \theta^\dagger \) is generated from the \( j \)-th component of the mixture, the acceptance probability is calculated as

\[ \tilde{\alpha}_j(\theta, \theta^\dagger | y) = \min \left\{ \frac{f(y|\theta^\dagger)\pi(\theta^\dagger)}{f(y|\theta)\pi(\theta)} \frac{i(\theta^\dagger|\mu_j, V_j, \nu_j)}{i(\theta|\mu_j, V_j, \nu_j)} \right\}. \]

An and Schorfheide (2007) report that 25% of the draws generated from the high mode component were accepted. In our implementation of this method we were unable to reproduce that rate. Instead, we find that less than 2% of the draws from the high mode component are accepted. We can increase the acceptance rate to 6.6% in Implementation II with the parameters \( c_l = 2, c_h = 2, \nu_l = 2, \) and \( \nu_h = 1. \) Neither implementation, however, is competitive with the the TaRBMJ-MH algorithm as can be seen from comparing the results in Table 7 with those
in Table 8. One clear difference is that the 90% intervals are narrower in Table 8 than those in Table 7. This suggests that the rejoinder method does not adequately explore the posterior surface. To illustrate this point, we provide in Figure 8 a plot of the (unnormalized) posterior ordinate at each sampled draw. The top panel corresponds to the draws from the method in the rejoinder whereas the bottom panel has the results from the TaRB-MH algorithm. These plots show the extensive movement of the TaRB-MH sampler, particularly in the low modal region. In sum, these results suggest that the TaRB-MH algorithm supplemented by a mode jumping step is a substantial improvement over the mode jumping
approach discussed in the rejoinder of An and Schorfheide (2007).

We conclude this section by calculating the marginal likelihood. As in the Ireland example, we present the marginal likelihood estimates from both the two and three stage implementations of the Chib method and different MCMC sample sizes. For comparison, we also include the marginal likelihood estimate based on the RW-MH output. The estimates are summarized in Table 9. Just as in the previous example, one can see from the numerical standard errors (which are given in parentheses) that the RW-MH based estimate is relatively inefficient. It is evident from the results that 75,000 iterations of the RW-MH algorithm are not sufficient to estimate the marginal likelihood. In contrast, a stable and efficient estimate can be obtained with a sample size of 5000 from the TaRB-MH algorithm, with either 2 or 3 stages.

Table 9–Log marginal likelihood estimates (with numerical standard errors) for the An and Schorfheide (2007) model based on the outputs from the TaRB-MH and RW-MH algorithms

<table>
<thead>
<tr>
<th></th>
<th>TaRB-MH</th>
<th>RW-MH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2 stage</td>
<td>3 stage</td>
</tr>
<tr>
<td>5,000</td>
<td>-206.64</td>
<td>-206.18</td>
</tr>
<tr>
<td></td>
<td>(0.1009)</td>
<td>(0.0944)</td>
</tr>
<tr>
<td>10,000</td>
<td>-206.58</td>
<td>-206.29</td>
</tr>
<tr>
<td></td>
<td>(0.0729)</td>
<td>(0.0671)</td>
</tr>
<tr>
<td>15,000</td>
<td>-206.55</td>
<td>-206.34</td>
</tr>
<tr>
<td></td>
<td>(0.0586)</td>
<td>(0.0544)</td>
</tr>
</tbody>
</table>

4.3 Smets and Wouters (2007) model

Our final application is the model in Smets and Wouters (2007) (SW07). The SW07 model, which is close in spirit to that of Christiano, Eichenbaum and Evans (2005), is fit to U.S. macroeconomic data for the period 1966:1 - 2004:4. These models, together with those in Smets and Wouters (2003) and Del Negro, Schorfheide, Smets and Wouters (2006) among others, constitute an important class of evolving, large scale, DSGE models featuring sticky nominal price and wage contracts, habit formation, variable capital utilization and investment adjustment costs.

The linearized model in SW07 comprises of 14 equations in 14 endogenous variables: output ($y_t$), consumption ($c_t$), investment ($i_t$), value of capital stock ($q_t$), capital ($k_t^*$),
installed capital \((k_t)\), capital utilization \((z_t)\), rental rate of capital \((r^t_k)\), price mark-up \((\mu^p_t)\), inflation \((\pi_t)\), wage mark-up \((\mu^w_t)\), real wage \((w_t)\), labor hours \((l_t)\) and interest rate \((r_t)\) under a sticky-price-wage setting, and 7 exogenous driving processes. This system is stacked with the corresponding version of the flexible-price-wage model that determines the potential output, \(y^*_t\), together with \(c^*_t\), \(i^*_t\), \(l^*_t\), \(k_t^*\), \(k_s^*\), \(z_t^*\), \(w_t^*\), \(r^k_t\), \(q_t^*\), \(\mu_t^w\), \(\mu_t^p\) and \(r_t^*\). A complete description of the model is given in Appendix A. From that description the canonical form of \((1)\) can be constructed. The canonical form is quite high-dimensional and involves a 53-dimensional state vector.

For estimation, the data comprises of quarterly time series of the log difference of real GDP, log difference of real consumption, log difference of real investment, log difference of real wage, log hours worked, log difference of GDP deflator and the federal funds rate for the US economy from 1966:I to 2004:IV. The endogenous variables relate to the vector of observables through the relation

\[
y_t = \begin{bmatrix}
dl\text{GDP}_t \\
\text{dlCONST}_t \\
\text{dlINV}_t \\
\text{dlWAG}_t \\
\text{lHOURS}_t \\
dlP_t \\
\text{FEDFUNDS}_t
\end{bmatrix} = \begin{bmatrix}
\bar{\gamma} \\
\bar{\gamma} \\
\bar{\gamma} \\
\bar{\gamma} \\
\bar{l} \\
\bar{\pi} \\
\bar{r}
\end{bmatrix} + \begin{bmatrix}
y_t - y_{t-1} \\
c_t - c_{t-1} \\
i_t - i_{t-1} \\
w_t - w_{t-1} \\
l_t \\
\pi_t \\
r_t
\end{bmatrix}
\]  

where \(l\) denotes 100 times log and \(dl\) refers to the log difference. Also, \(\bar{\gamma}\) is the quarterly trend growth rate of real GDP, consumption, investment and wages, \(\bar{\pi}\) is the quarterly steady-state inflation rate and \(\bar{r}\) is the steady-state nominal interest rate. Of the 41 underlying parameters in the model, 5 are held fixed. The prior distribution of the remaining parameters (as specified in SW07) are summarized in Table 10. Our interest centers around the 36 dimensional posterior distribution resulting from the sampling density and prior distribution of the parameters.

4.3.1 Sampling results from the TaRB-MH algorithm

As one might anticipate, it is significantly more challenging to sample the posterior distribution of the parameters in this model, as compared to, say, in the Ireland model in example 1, on account of the high dimensionality of the parameter space. To appreciate the complexity of the problem, we begin by focusing on the task of finding the posterior
Table 10 – Prior distribution in the Smets and Wouters (2007) model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Density</th>
<th>Mean</th>
<th>Std dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varphi$</td>
<td>Normal</td>
<td>4.00</td>
<td>1.50</td>
</tr>
<tr>
<td>$\sigma_c$</td>
<td>Normal</td>
<td>1.50</td>
<td>0.37</td>
</tr>
<tr>
<td>$h$</td>
<td>Beta</td>
<td>0.70</td>
<td>0.10</td>
</tr>
<tr>
<td>$\xi_w$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.10</td>
</tr>
<tr>
<td>$\sigma_l$</td>
<td>Normal</td>
<td>2.00</td>
<td>0.75</td>
</tr>
<tr>
<td>$\xi_p$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.10</td>
</tr>
<tr>
<td>$\iota_w$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.15</td>
</tr>
<tr>
<td>$\iota_p$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.15</td>
</tr>
<tr>
<td>$\psi$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.15</td>
</tr>
<tr>
<td>$\Phi$</td>
<td>Normal</td>
<td>1.25</td>
<td>0.12</td>
</tr>
<tr>
<td>$r_\pi$</td>
<td>Normal</td>
<td>1.50</td>
<td>0.250</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Beta</td>
<td>0.75</td>
<td>0.10</td>
</tr>
<tr>
<td>$r_y$</td>
<td>Normal</td>
<td>0.12</td>
<td>0.05</td>
</tr>
<tr>
<td>$r_{\Delta y}$</td>
<td>Normal</td>
<td>0.12</td>
<td>0.05</td>
</tr>
<tr>
<td>$\bar{\pi}$</td>
<td>Gamma</td>
<td>0.62</td>
<td>0.10</td>
</tr>
<tr>
<td>$100(\beta^{-1} - 1)$</td>
<td>Gamma</td>
<td>0.25</td>
<td>0.10</td>
</tr>
<tr>
<td>$\bar{l}$</td>
<td>Normal</td>
<td>0.00</td>
<td>2.00</td>
</tr>
<tr>
<td>$\bar{\gamma}$</td>
<td>Normal</td>
<td>0.40</td>
<td>0.10</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Normal</td>
<td>0.30</td>
<td>0.05</td>
</tr>
<tr>
<td>$\sigma_a$</td>
<td>Inverse Gamma</td>
<td>0.10</td>
<td>2.00</td>
</tr>
<tr>
<td>$\sigma_b$</td>
<td>Inverse Gamma</td>
<td>0.10</td>
<td>2.00</td>
</tr>
<tr>
<td>$\sigma_g$</td>
<td>Inverse Gamma</td>
<td>0.10</td>
<td>2.00</td>
</tr>
<tr>
<td>$\sigma_l$</td>
<td>Inverse Gamma</td>
<td>0.10</td>
<td>2.00</td>
</tr>
<tr>
<td>$\sigma_r$</td>
<td>Inverse Gamma</td>
<td>0.10</td>
<td>2.00</td>
</tr>
<tr>
<td>$\sigma_p$</td>
<td>Inverse Gamma</td>
<td>0.10</td>
<td>2.00</td>
</tr>
<tr>
<td>$\sigma_w$</td>
<td>Inverse Gamma</td>
<td>0.10</td>
<td>2.00</td>
</tr>
<tr>
<td>$\rho_a$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td>$\rho_b$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td>$\rho_g$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td>$\rho_l$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td>$\rho_r$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td>$\rho_p$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td>$\rho_w$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td>$\mu_p$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td>$\mu_w$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td>$\rho_{qa}$</td>
<td>Beta</td>
<td>0.50</td>
<td>0.20</td>
</tr>
</tbody>
</table>

Remark: The Inverse Gamma distribution here refers to the Inverse Gamma type-I distribution as in the AS07 model.

mode. Our extensive experimentation with the simulated annealing optimizer using various starting values, stage lengths, scale factors, initial temperatures and temperature reduction factors produced an unnormalized posterior modal ordinate in the vicinity of
-877.72 on the log scale. In comparison, the modal ordinate in SW07 is -906.29. This reiterates the point in Sims, Waggoner and Zha (forthcoming) that it is difficult to find the dominant mode in large scale macroeconomic models. Consequently, as noted in the context of the Ireland model above, relying on the mode as the starting value of a RW chain, with the variance of the increment obtained from the negative inverse of the Hessian at the mode, is not a reliable strategy. As we have argued above, the TaRB-MH (because it does not work with the entire parameter in one block) does not suffer from these problems. As we show below, it finds even higher density regions than those found by the SA algorithm and improves significantly on the RW-MH algorithm.

Table 11a–Posterior summary: structural parameters in SW07 model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>SW07 Posterior Mean</th>
<th>90 percent interval</th>
<th>TaRB-MH Posterior Mean</th>
<th>90 percent interval</th>
<th>Inefficiency factors</th>
</tr>
</thead>
<tbody>
<tr>
<td>ϕ</td>
<td>5.74</td>
<td>[3.97, 7.42]</td>
<td>5.77</td>
<td>[3.69, 8.06]</td>
<td>9.73</td>
</tr>
<tr>
<td>σ&lt;sub&gt;c&lt;/sub&gt;</td>
<td>1.38</td>
<td>[1.16, 1.59]</td>
<td>1.36</td>
<td>[1.03, 1.71]</td>
<td>12.14</td>
</tr>
<tr>
<td>h</td>
<td>0.71</td>
<td>[0.64, 0.78]</td>
<td>0.75</td>
<td>[0.66, 0.82]</td>
<td>15.64</td>
</tr>
<tr>
<td>ξ&lt;sub&gt;ω&lt;/sub&gt;</td>
<td>0.70</td>
<td>[0.60, 0.81]</td>
<td>0.65</td>
<td>[0.52, 0.79]</td>
<td>54.28</td>
</tr>
<tr>
<td>σ&lt;sub&gt;l&lt;/sub&gt;</td>
<td>1.83</td>
<td>[0.91, 2.78]</td>
<td>1.98</td>
<td>[0.96, 3.21]</td>
<td>28.79</td>
</tr>
<tr>
<td>ξ&lt;sub&gt;p&lt;/sub&gt;</td>
<td>0.66</td>
<td>[0.56, 0.74]</td>
<td>0.62</td>
<td>[0.49, 0.75]</td>
<td>45.22</td>
</tr>
<tr>
<td>τ&lt;sub&gt;ω&lt;/sub&gt;</td>
<td>0.58</td>
<td>[0.38, 0.78]</td>
<td>0.59</td>
<td>[0.31, 0.83]</td>
<td>8.65</td>
</tr>
<tr>
<td>τ&lt;sub&gt;p&lt;/sub&gt;</td>
<td>0.24</td>
<td>[0.10, 0.38]</td>
<td>0.23</td>
<td>[0.08, 0.41]</td>
<td>18.13</td>
</tr>
<tr>
<td>ψ</td>
<td>0.54</td>
<td>[0.36, 0.72]</td>
<td>0.59</td>
<td>[0.36, 0.81]</td>
<td>5.93</td>
</tr>
<tr>
<td>Φ</td>
<td>1.60</td>
<td>[1.48, 1.73]</td>
<td>1.57</td>
<td>[1.42, 1.74]</td>
<td>6.44</td>
</tr>
<tr>
<td>r&lt;sub&gt;π&lt;/sub&gt;</td>
<td>2.04</td>
<td>[1.74, 2.33]</td>
<td>2.00</td>
<td>[1.64, 2.37]</td>
<td>11.21</td>
</tr>
<tr>
<td>ρ</td>
<td>0.81</td>
<td>[0.77, 0.85]</td>
<td>0.80</td>
<td>[0.75, 0.85]</td>
<td>11.42</td>
</tr>
<tr>
<td>r&lt;sub&gt;y&lt;/sub&gt;</td>
<td>0.08</td>
<td>[0.05, 0.12]</td>
<td>0.08</td>
<td>[0.03, 0.13]</td>
<td>26.12</td>
</tr>
<tr>
<td>r&lt;sub&gt;Δy&lt;/sub&gt;</td>
<td>0.22</td>
<td>[0.18, 0.27]</td>
<td>0.23</td>
<td>[0.17, 0.29]</td>
<td>5.68</td>
</tr>
<tr>
<td>π&lt;sub&gt;ω&lt;/sub&gt;</td>
<td>0.78</td>
<td>[0.61, 0.96]</td>
<td>0.66</td>
<td>[0.48, 0.85]</td>
<td>3.81</td>
</tr>
<tr>
<td>100(β&lt;sup&gt;-1&lt;/sup&gt; − 1)</td>
<td>0.16</td>
<td>[0.07, 0.26]</td>
<td>0.16</td>
<td>[0.06, 0.29]</td>
<td>6.09</td>
</tr>
<tr>
<td>l</td>
<td>0.53</td>
<td>[-1.3, 2.32]</td>
<td>0.95</td>
<td>[-0.07, 2.56]</td>
<td>9.39</td>
</tr>
<tr>
<td>γ&lt;sub&gt;ω&lt;/sub&gt;</td>
<td>0.43</td>
<td>[0.40, 0.45]</td>
<td>0.41</td>
<td>[0.37, 0.46]</td>
<td>9.51</td>
</tr>
<tr>
<td>α</td>
<td>0.19</td>
<td>[0.16, 0.21]</td>
<td>0.19</td>
<td>[0.15, 0.23]</td>
<td>4.68</td>
</tr>
</tbody>
</table>

Remark: The TaRB-MH sampler resulted in acceptance rates between 40 and 45 % for all the parameters.

As in the Ireland example, we initialized the TaRB-MH chain at the prior mean. We also chose similar values for the SA parameters following the template outlined in our discussion of the Ireland model. Because it is more time consuming to evaluate the likelihood function in this problem, we reduced the number of stages \( n \) to 4 and the stage
length increment $b$ to 6.\(^8\) While these choices imply that the tailoring is coarser, the results are not sensitive to these choices. Finally, we also set the degrees of freedom in the student-$t$ proposal density to 10. The results are summarized in Tables 11a and 11b. For comparison, these tables also include the results from SW07. Our results are based on 10,000 iterations of the TaRB-MH algorithm following a burn-in of 1000 iterations, whereas the reported RW-MH results are based on 250,000 iterations beyond a burn-in of 50,000 iterations. The important differences in the results from the two algorithms are listed below.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>SW07 Posterior Mean 90 percent interval</th>
<th>TaRB-MH Posterior Mean 90 percent interval</th>
<th>Inefficiency factors</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_a$</td>
<td>0.45 [0.41, 0.50]</td>
<td>0.46 [0.41, 0.53]</td>
<td>4.17</td>
</tr>
<tr>
<td>$\sigma_b$</td>
<td>0.23 [0.19, 0.27]</td>
<td>0.25 [0.18, 0.30]</td>
<td>15.23</td>
</tr>
<tr>
<td>$\sigma_g$</td>
<td>0.53 [0.48, 0.58]</td>
<td>0.53 [0.47, 0.59]</td>
<td>2.57</td>
</tr>
<tr>
<td>$\sigma_I$</td>
<td>0.45 [0.37, 0.53]</td>
<td>0.43 [0.34, 0.55]</td>
<td>33.14</td>
</tr>
<tr>
<td>$\sigma_r$</td>
<td>0.24 [0.22, 0.27]</td>
<td>0.25 [0.22, 0.28]</td>
<td>4.30</td>
</tr>
<tr>
<td>$\sigma_p$</td>
<td>0.14 [0.11, 0.16]</td>
<td>0.14 [0.10, 0.18]</td>
<td>14.89</td>
</tr>
<tr>
<td>$\sigma_w$</td>
<td>0.24 [0.20, 0.28]</td>
<td>0.26 [0.21, 0.32]</td>
<td>11.62</td>
</tr>
<tr>
<td>$\rho_a$</td>
<td>0.95 [0.94, 0.97]</td>
<td>0.96 [0.93, 0.98]</td>
<td>6.15</td>
</tr>
<tr>
<td>$\rho_b$</td>
<td>0.22 [0.07, 0.36]</td>
<td>0.21 [0.04, 0.49]</td>
<td>24.42</td>
</tr>
<tr>
<td>$\rho_g$</td>
<td>0.97 [0.96, 0.99]</td>
<td>0.98 [0.96, 0.99]</td>
<td>7.53</td>
</tr>
<tr>
<td>$\rho_I$</td>
<td>0.71 [0.61, 0.80]</td>
<td>0.74 [0.61, 0.86]</td>
<td>37.54</td>
</tr>
<tr>
<td>$\rho_r$</td>
<td>0.15 [0.04, 0.24]</td>
<td>0.15 [0.04, 0.30]</td>
<td>6.64</td>
</tr>
<tr>
<td>$\rho_p$</td>
<td>0.89 [0.80, 0.96]</td>
<td>0.89 [0.75, 0.98]</td>
<td>48.92</td>
</tr>
<tr>
<td>$\rho_w$</td>
<td>0.96 [0.94, 0.99]</td>
<td>0.98 [0.96, 1.00]</td>
<td>21.80</td>
</tr>
<tr>
<td>$\rho_p$</td>
<td>0.69 [0.54, 0.85]</td>
<td>0.66 [0.38, 0.84]</td>
<td>38.23</td>
</tr>
<tr>
<td>$\rho_{w_a}$</td>
<td>0.84 [0.75, 0.93]</td>
<td>0.83 [0.63, 0.94]</td>
<td>43.72</td>
</tr>
<tr>
<td>$\rho_{ga}$</td>
<td>0.52 [0.37, 0.66]</td>
<td>0.50 [0.32, 0.69]</td>
<td>2.61</td>
</tr>
</tbody>
</table>

- Posterior mean is significantly different for parameters $\bar{\pi}$ and $\bar{l}$. Note that the corresponding 90\% intervals are shifted for these parameters. Interestingly, the posterior ordinate at the mean of the TaRB-MH sample is significantly higher that that at the mean of the SW07 RW-MH sample (-871.66 compared to -888.84). A comparative summary of the unnormalized posterior ordinates at the mode and

\(^8\)Specifically, the likelihood evaluation in the SW07 model takes roughly 30 times longer than in the Ireland model.
mean in the published results and the results that we obtain using the SA algorithm for finding the mode and the TaRB-MH algorithm for finding the posterior mean are given in Table 12.

Table 12–Summary of posterior ordinates at the mode and mean in the Smets and Wouters (2007) model

<table>
<thead>
<tr>
<th></th>
<th>SW07</th>
<th>SA/TaRB-MH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>-906.29</td>
<td>-877.72</td>
</tr>
<tr>
<td>Mean</td>
<td>-888.84</td>
<td>-871.66</td>
</tr>
</tbody>
</table>

• For the remaining parameters, the posterior means are roughly comparable. However, the TaRB-MH algorithm explores more of the posterior, as indicated by the wider 90% intervals for all but one parameter ($\rho_w$).

• In the case of the TaRB-MH algorithm, the inefficiency factors are in the range of 2.61 to 54.28, with most values below 30. In contrast, the inefficiency factors from our well tuned implementation of the RW-MH sampler (initialized at the mode found by simulated annealing) are above 2500. This points to the high degree of autocorrelation in the draws from the RW-MH sampler, as also noted in SW07. To illustrate the differences in the mixing of the two samplers over the parameters space, we plot the sampled draws from the posterior distribution for select parameters together with their ACFs in Figure 9. The top panel corresponds to the draws from the TaRB-MH sampler and the bottom panel corresponds to the RW-MH sampler. As these plots show, the RW-MH chain is highly persistent with the ACFs retaining significant mass even at lag 1000. On the other hand, the TaRB-MH ACFs decay quickly (within 30-40 lags for most parameters).

It is worthwhile mentioning that 10,000 iterations of the TaRB-MH algorithm take on average 6 to 7 times longer to complete than 250,000 iterations of the RW-MH algorithm. This computing time however does not reflect the time required to tune the RW-MH algorithm, which, as noted above, increases substantially with the dimensionality of the parameter space. A more serious problem with a naive comparison of computing times is
that is fails to recognize that the RW-MH simply does not work in many circumstances. It is therefore vital to have the correct answer even if it is costly to obtain in terms of computing time. A final point is that computing time is not a constant and has rapidly
declined over time and can be expected to continue to fall in the near future. As interest in more complicated DSGE models grows, the strategies outlined in this paper should prove ever more valuable.

5 Conclusion

In this paper we provide new MCMC approaches for estimating DSGE models. The organizing principle is the sampling of the posterior density by a specially constructed version of a (randomized) multiple-block M-H algorithm that is designed to deal with the sort of considerations that arise in the context of these models. Specifically, the blocks and its constituents are constructed randomly and the proposal density in the M-H sampling is found from the output of simulated annealing. The combined thrust of these innovations proves extremely effective as we show in the context of our three examples. We also provide a straightforward extension of our general approach that can be used for sampling distributions in which one suspects multiple modes. On both dimensions, this improves significantly on what has been achieved so far. Finally, we show how the model marginal likelihood, which is needed in the comparison of alternative DSGE models, can be computed efficiently from the output of our tailored randomized block sampling schemes. In sum, given that the methods proposed here are reliable and efficient, they have the potential for broadening the appeal and practicality of Bayesian methods in the fitting of DSGE models.

A Appendix: Linearized model in Smets and Wouters (2007)

This appendix provides the complete SW07 model that includes the system of equations under both the sticky-price-wage and flexible price-wage settings (the latter marked by asterisks), together with the steady states of the relevant variables. We also include the state vector \( s_t \), the vector of innovations \( \varepsilon_t \) and the vector of expectational errors \( \eta_t \) so that the canonical form of the model can be constructed. The writing of this section has benefited considerably from Giorgio Primiceri’s Matlab code of the canonical representation.

The following auxiliary parameters (in terms of the parameters in Table 10) are
useful for the calculations below: \( \beta = 100(\hat{\beta} + 100)^{-1}, \gamma = \bar{\gamma}/100, \lambda_p = \phi_p - 1 \). In addition, the following parameters are fixed as in SW07: \( \delta = 0.025, \lambda_w = \phi_w - 1 = 0.5, g_y = 0.18, \varepsilon_w = \varepsilon_p = 10 \). Also, for the steady state calculations below, define \( I^{ss} = \bar{I} \).

Subsequently, \( l^{ss} = \bar{l} \).

The steady state values:

\[
\tilde{\tau} = 100(\beta^{-1} e^{\gamma \sigma_p} - 1)
\]

\[
\bar{\pi} + \tilde{\tau}
\]

\[
g_y = 1/(1 - g_y)
\]

\[
r^{kss} = e^{\gamma \sigma_p} - 1 + \delta
\]

\[
\mu^{pss} = 1/(1 + \lambda_p)
\]

\[
u^{ss} = \left[ \frac{\mu^{pss}(1 - \alpha)(1 - \gamma)(1 - \alpha)}{(\alpha - \gamma)(1 - \alpha)} \right]^{(l/1 - \alpha)}
\]

\[
kl^{ss} = (w^{ss}/r^{kss})(\alpha/(1 - \alpha))
\]

\[
F^{lss} = (kl^{ss})^{\alpha} - r^{kss}kl^{ss} - w^{ss}
\]

\[
y^{lss} = kl^{ss} - F^{lss}
\]

\[
k^{ss} = kl^{ss}l^{ss}
\]

\[
i^{ss} = (1 - (1 - \delta)e^{-\gamma})k^{ss}e^{\gamma}
\]

\[
F = F^{lss}l^{ss}
\]

\[
y^{ss} = y^{lss}l^{ss}
\]

\[
c^{ss} = (y^{ss}/g_y) - i^{ss}
\]

The linearized model:

\[
y_t = \frac{c^{ss}}{y^{ss}c_t} + \frac{i^{ss}}{y^{ss}l_t} + \frac{k^{ss}r^{kss}}{y^{ss}l_t} z_t + u^g_t
\]

\[
y^*_t = \frac{c^{ss}}{y^{ss}c^*_t} + \frac{i^{ss}}{y^{ss}l^*_t} + \frac{k^{ss}r^{kss}}{y^{ss}l^*_t} z^*_t + u^g_t
\]

\[
c_t = \frac{\lambda e^{-\gamma}}{1 + \lambda e^{-\gamma}}c_{t-1} + \frac{1}{1 + \lambda e^{-\gamma}}Et_{c_{t+1}} + \frac{(\sigma_c - 1)w^{ss}I^{ss}}{c^{ss}e_c(1 + \lambda e^{-\gamma})(1 + \lambda_w)}(l_t - E_t l_{t+1})
\]

\[
- \frac{1 - \lambda e^{-\gamma}}{\sigma_c(1 + \lambda e^{-\gamma})} (r_t - E_t \bar{\pi}_{t+1}) + u^b_t
\]

\[
c^*_t = \frac{\lambda e^{-\gamma}}{1 + \lambda e^{-\gamma}}c^*_{t-1} + \frac{1}{1 + \lambda e^{-\gamma}}Et_{c^*_{t+1}} + \frac{(\sigma_c - 1)w^{ss}I^{ss}}{c^{ss}e_c(1 + \lambda e^{-\gamma})(1 + \lambda_w)}(l_t - E_t l^*_{t+1})
\]

\[
- \frac{1 - \lambda e^{-\gamma}}{\sigma_c(1 + \lambda e^{-\gamma})} r^*_{t+1} + u^b_t
\]
\( i_t = \frac{1}{1 + \beta e^{1-\sigma \gamma}} i_{t-1} + \frac{\beta e^{1-\sigma \gamma}}{1 + \beta e^{1-\sigma \gamma}} E_t i_{t+1} + \frac{1}{(1 + \beta e^{1-\sigma \gamma})} \varphi e^{2\gamma} q_t + u^i_t \)

\( i^*_t = \frac{1}{1 + \beta e^{1-\sigma \gamma}} i^*_{t-1} + \frac{\beta e^{1-\sigma \gamma}}{1 + \beta e^{1-\sigma \gamma}} E_t i^*_{t+1} + \frac{1}{(1 + \beta e^{1-\sigma \gamma})} \varphi e^{2\gamma} q^*_t + u^i_t \)

\( q_t = \beta (1 - \delta) e^{-\gamma} 1_t - (1 - \delta) e^{-\gamma} 1_{t+1} + E_t t_{t+1}^k - t^i_t + E_t \pi_{t+1} \)

\( q^*_t = \beta (1 - \delta) e^{-\gamma} 1_t - (1 - \delta) e^{-\gamma} 1_{t+1} + E_t t^*_{t+1} - t^*_{t} + \frac{\sigma_c (1 + \lambda e^{-\gamma})}{1 - \lambda e^{-\gamma}} u^b_t \)

\( y_t = \frac{y^{ss} + F}{y^{ss} \alpha k^*_t} + \frac{y^{ss} + F}{y^{ss}} (1 - \alpha) l_t + \frac{y^{ss} + F}{y^{ss}} u^a_t \)

\( y^*_t = \frac{y^{ss} + F}{y^{ss} \alpha k^{ss}_t} + \frac{y^{ss} + F}{y^{ss}} (1 - \alpha) l^*_t + \frac{y^{ss} + F}{y^{ss}} u^a_t \)

\( k_t^* = z^* + k_{t-1} \)

\( k_t = (1 - \delta) e^{-\gamma} k_{t-1} + (1 - (1 - \delta) e^{-\gamma}) i_t \\
+ (1 - (1 - \delta) e^{-\gamma})(1 + \beta e^{1-\sigma \gamma}) \varphi e^{2\gamma} u^i_t \)

\( k^{ss}_t = (1 - \delta) e^{-\gamma} k^{ss}_{t-1} + (1 - (1 - \delta) e^{-\gamma}) i^*_t \\
+ (1 - (1 - \delta) e^{-\gamma})(1 + \beta e^{1-\sigma \gamma}) \varphi e^{2\gamma} u^i_t \)

\( \mu^b_t = \alpha r^k_t + (1 - \alpha) w_t - u^a_t \)

\( \mu^{ps}_t = \alpha r^{ks}_t + (1 - \alpha) w^*_t - u^a_t \)

\( \pi_t = \frac{\ell_p}{1 + \ell_p \beta e^{1-\sigma \gamma} \pi_{t-1} + \beta e^{1-\sigma \gamma}} E_t \pi_{t+1} + \frac{1 - \beta e^{1-\sigma \gamma} \xi_p (1 - \xi_p)}{(1 + \ell_p e^{1-\sigma \gamma} \beta) \xi_p ((\phi_p - 1) \xi_p + 1)} \mu^p_t + u^{ps}_t \)

\( \mu^{ps}_t = \frac{1}{1} \)

\( r^k_t = w_t - k_t + l_t \)

\( r^{ks}_t = w^*_t - k^*_t + l^*_t \)

\( \mu^w_t = w_t - \sigma l_t - \frac{1}{1 - \lambda e^{-\gamma} c_t} + \frac{\lambda e^{-\gamma}}{1 - \lambda e^{-\gamma} c^*_{t-1}} \)

\( \mu^{ws}_t = -\sigma l^*_t - \frac{1}{1 - \lambda e^{-\gamma} c^*_t} + \frac{\lambda e^{-\gamma}}{1 - \lambda e^{-\gamma} c^*_{t-1}} \)
\[ w_t = \frac{1}{1 + \beta e^{(1-\sigma)\gamma}} w_{t-1} + \left(1 - \frac{1}{1 + \beta e^{(1-\sigma)\gamma}}\right) (E_t w_{t+1} + E_t \pi_{t+1}) \]
\[ - \frac{1 + \lambda \beta e^{(1-\sigma)\gamma}}{1 + \beta e^{(1-\sigma)\gamma}} \pi_t + \frac{\lambda}{1 + \beta e^{(1-\sigma)\gamma}} \pi_{t-1} \]
\[ - \frac{(1 - \xi_w)(1 - \xi_w/\beta e^{(1-\sigma)\gamma})}{(1 + \beta e^{(1-\sigma)\gamma})\xi_w((\phi_w - 1)\varepsilon_w + 1)} \mu_t^w + u_t^w \]
\[ w_t^\pi = \mu_t^w \]
\[ r_t = \rho r_{t-1} + (1 - \rho)(\pi \pi_t) - [(1 - \rho)(r y + r \Delta y) y_t - [(1 - \rho)r y + r \Delta y] y_t^* \]
\[ - r \Delta y y_{t-1} + r \Delta y y_{t-1} + u_t^r \]

The exogenous shock processes:

\[ u_t^a = \rho_a u_{t-1}^a + \varepsilon_t^a \]
\[ u_t^b = \rho_b u_{t-1}^b + \varepsilon_t^b \]
\[ u_t^g = \rho_g u_{t-1}^g + \varepsilon_t^g + \rho_g \varepsilon_t^a \]
\[ u_t^i = \rho_i u_{t-1}^i + \varepsilon_t^i \]
\[ u_t^r = \rho_r u_{t-1}^r + \varepsilon_t^r \]
\[ u_t^p = \rho_p u_{t-1}^p - \mu_p \varepsilon_t^p + \varepsilon_t^p \]
\[ \varepsilon_t^p = \varepsilon_t^p \]
\[ u_t^w = \rho_w u_{t-1}^w - \mu_w \varepsilon_t^w + \varepsilon_t^w \]
\[ \varepsilon_t^w = \varepsilon_t^w \]

The expectational error terms:

\[ \pi_t = E_{t-1} \pi_t + \eta_t^\pi \]
\[ c_t = E_{t-1} c_t + \eta_t^c \]
\[ l_t = E_{t-1} l_t + \eta_t^l \]
\[ q_t = E_{t-1} q_t + \eta_t^q \]
\[ r_t^k = E_{t-1} r_t^k + \eta_t^r \]
\[ i_t = E_{t-1} i_t + \eta_t^i \]
\[ w_t = E_{t-1} w_t + \eta_t^w \]
\[ c_t^* = E_{t-1} c_t^* + \eta_t^c \]
\[ l_t^* = E_{t-1} l_t^* + \eta_t^l \]
\[ q_t^* = E_{t-1} q_t^* + \eta_t^q \]
\[ r_t^k = E_{t-1} r_t^k + \eta_t^k \]
\[ i_t^* = E_{t-1} i_t^* + \eta_t^i \]

Dummy equations for lagged variables:
\[ y_{t-1} = y_{t-1} \]
\[ c_{t-1} = c_{t-1} \]
\[ i_{t-1} = i_{t-1} \]
\[ w_{t-1} = w_{t-1} \]
\[ y_{t-1} = y_{t-1} \]

To cast the model in the canonical form (1), define the 53 dimensional state vector
\[ s_t = [y_t, k_t^*, l_t, r_t^k, w_t, \pi_t, \mu_t^p, c_t, r_t, z_t, q_t, k_t, \mu_t^w, E_t \pi_{t+1}, E_t c_{t+1}, E_t l_{t+1}, E_t q_{t+1}, E_t r_{t+1}, E_t w_{t+1}, y_{t-1}, c_{t-1}, i_{t-1}, w_{t-1}, u_{t, a}^b, u_{t, a}^q, u_{t, i}^a, u_{t, i}^q, u_{t, w}^p, u_{t, w}^e, \varepsilon_{t}^a, \varepsilon_{t}^b, \varepsilon_{t}^q, \varepsilon_{t}^i, \varepsilon_{t}^w, \pi_t - E_{t-1} \pi_t, c_t - E_{t-1} c_t, l_t - E_{t-1} l_t, q_t - E_{t-1} q_t, r_t - E_{t-1} r_t, i_t - E_{t-1} i_t, \]
\[ E_t r_{t+1}^k, \]
\[ E_t w_{t+1}^* = \mu_t^w, c_t^*, r_t^*, z_t^*, q_t^*, i_t^*, k_t^*, \mu_t^w, E_t l_{t+1}^*, E_t q_{t+1}^*, E_t r_{t+1}^k, E_t w_{t+1}^* = y_{t-1} \]

the vector of innovations
\[ \varepsilon_t = [\varepsilon_t^a, \varepsilon_t^b, \varepsilon_t^q, \varepsilon_t^i, \varepsilon_t^w] \]

and the vector of endogenous rational expectations errors
\[ \eta_t = [\pi_t - E_{t-1} \pi_t, c_t - E_{t-1} c_t, l_t - E_{t-1} l_t, q_t - E_{t-1} q_t, r_t^k - E_{t-1} r_t^k, i_t - E_{t-1} i_t, \]
\[ w_t - E_{t-1} w_t, c_t^* - E_{t-1} c_t^*, l_t^* - E_{t-1} l_t^*, q_t^* - E_{t-1} q_t^*, r_t^k - E_{t-1} r_t^k, i_t^* - E_{t-1} i_t^*] \]

Based on this information, the canonical form of the model can be readily formulated.

References


60