Bayesian Inference in a Non-linear/Non-Gaussian Switching State Space Model: Regime-dependent Leverage Effect in the U.S. Stock Market

Kim, Jaeho

University of Oklahoma

9 October 2015

Bayesian Inference in a Non-linear/Non-Gaussian Switching State Space Model: Regime-dependent Leverage Effect in the U.S. Stock Market

Jaeho Kim*

University of Oklahoma

September, 2015

Abstract

This paper provides two Bayesian algorithms to efficiently estimate non-linear/non-Gaussian switching state space models by extending a standard Particle Markov chain Monte Carlo (PMCMC) method. Instead of iteratively running separate PMCMC steps using conventional approaches, the proposed methods generate continuous-state and discrete-regime indicator variables together from their joint smoothing distribution in one Gibbs block. The proposed Bayesian algorithms that are built upon the novel ideas of ancestor sampling and particle rejuvenation are robust to small numbers of particles and degenerate state transition equations. Moreover, the algorithms are applicable to any switching state space models, regardless of the Markovian property. The difficulty in conducting Bayesian model comparisons is overcome by adopting the Deviance Information Criterion (DIC). For illustration, a regime-dependent leverage effect in the U.S. stock market is investigated using the newly developed methods. A conventional regime switching stochastic volatility model is generalized to encompass the regime-dependent leverage effect and is applied to Standard and Poor's 500 and NASDAQ daily return data. The resulting Bayesian posterior estimates indicate that the stronger (weaker) financial leverage effect is associated with a high (low) volatility regime.

Keywords: Particle Markov Chain Monte Carlo, Regime switching, State space model, Leverage effect

JEL classification: C15

*Department of Economics, University of Oklahoma, Norman, OK 73019, U.S.A. [E-mail: jaeho@ou.edu]

I gratefully acknowledge financial support through the Junior Faculty Research Program at the University of Oklahoma.

1 Introduction

The dynamics of many economic and financial time series often dramatically change, in association with important economic events, such as economic policy changes, economic recessions, and financial crises. Since the seminar article by Hamilton (1989), numerous studies have statistically handled such abrupt changes in fundamental economic structures. In particular, linear/Gaussian switching state space models (LG-SSSMs) have been of great use in the economic literature due to their flexibility in encompassing a broad range of economic models[1]. However, though LG-SSSMs have been proved to be quite useful in the literature, they have some drawbacks. Most importantly, they impose linearity and Gaussianity assumptions that are too restrictive to handle fundamentally non-linear economic variables with non-Gaussian innovations[2]. On this ground, it is important to develop an efficient method to estimate the novel class of non-linear/non-Gaussian switching state space models (NLG-SSSMs), and this paper attempts to achieve this goal by extending the standard Particle Markov Chain Monte Carlo method[3] (PMCMC) by Andrieu et al. (2010).

One of main difficulties in estimating NLG-SSSMs is that latent continuous-state and discrete-regime indicator variables that drive a dynamic system usually have high dimensions and complex patterns of dependence. Consequently, posterior distributions of model parameters do not admit closed-form expressions in most cases, which makes Bayesian inference very difficult, in practice. Conventional Bayesian methods make use of Metropolis-Hastings and standard Gibbs sampling approaches to mitigate the complex inference problem. For instance, Flury and Shephard (2011) develop a PMCMC algorithm using a Particle Marginal Metropolis-Hastings (PMMH) approach and apply it to three popular economic models. Even though their PMCMC method can be extended for Bayesian inference of NLG-SSSMs, convergence of their sampler will extremely slow especially without a large number of particles[4]. Of cause, one may achieve satisfactory convergence by increasing the number of particles, which is computationally very demanding for complex dynamic models. Moreover, since their PMMH algorithm employs random walk proposals in generating model parameters, it requires to tune variances of the proposals, aiming for a certain acceptance probability. This can be usually done through trial and error, which is extremely time consuming.

[1] See Fruhwirth-Schnatter (2006), Kim and Nelson (1999), and Giordani et al. (2007) and references therein.

[2] Dynamic stochastic general equilibrium (DSGE) models by Smets and Wouters (2007) and An and Schorfheide (2007) and stochastic volatility (SV) models by Kim et al. (1998) are prominent examples of non-linear/non-Gaussian state space models, among many others.

[3] In a PMCMC algorithm, a sequential Monte Carlo method, also known as a particle filter, is applied to numerically approximate posterior distributions of interest using random samples called particles. The various particle trajectories of latent state variables form the approximate distributions and are used to construct proposal kernels for an MCMC sampler.

[4] Pitt et al. (2012) provide detailed analysis of the trade-off between the convergence speed and computational cost of PMMH algorithms.

Nonejad (2014) recently proposed a PMCMC method based on a Gibbs sampling approach to estimate NLG-SSSMs. The proposed method is implemented by first drawing a continuous state variable, say x_t, given a regime indicator variable s_t and then drawing the regime indicator variable s_t without conditioning on x_t in the second step. The second step of the algorithm generates s_t simply by replacing the true likelihood with the approximate likelihood using a sequential Monte Carlo (SMC) method to integrate out x_t. However, because the approximation errors generated by a SMC method are completely ignored, the errors will introduce some bias by propagating through the resulting MCMC sampler.

The problem can be solved by properly combining Particle Marginal Metropolis-Hastings (PMMH) and Particle Gibbs (PG) methods according to a general PMCMC scheme suggested by Mendes et al. (2014). More specifically, a PG step is employed to generate x_t conditional on s_t, and then, a separate PMMH step is performed for posterior simulation on s_t. One critical shortcoming of this remedy, however, is that the PMMH step is based on a single-move sampler. Liu et al. (1994) and Scott (2002) theoretically showed that a single-move sampler produces significantly high autocorrelations among successive posterior draws of the regime indicator variable and other model parameters in regime switching models. Moreover, Kim and Kim (2014) empirically showed that a correct stationary distribution is almost never achieved if the regime indicator variable is generated based on a single-move sampler when the regime indicator variable is very persistent or has absorbing states.

Song (2014) developed a PMCMC algorithm by exploiting the partially linear structure of a switching state space model and incorporating Kim's (1994) approximate filtering and smoothing algorithms. An efficient PMCMC algorithm is proposed by Whiteley et al. (2010) to estimate linear/Gaussian SSSM. Note that the empirical models of U.S. stock returns in Section 4 involve fully non-linear transition and measurement equations. Moreover, the posterior estimates of the transition probabilities for regime changes indicate that the regime indicator variables are indeed highly persistent. Hence, all the aforementioned PMCMC schemes are not directly applicable in this article.

One of the main contributions of this paper is developing efficient Bayesian algorithms to estimate NLG-SSSMs to address the practical problems. For this purpose, I adopt a PG sampling approach for the latent variables. Because PG sampling does not require an unnecessary accept/reject step, it produces mixing superior to that obtained using PMMH samplers. The PG algorithms proposed by this article sequentially generate all the latent variables together from the joint smoothing distribution of x_t and s_t, in contrast to the conventional approaches, which iteratively run separate PMCMC steps. The joint sampling can be effectively done in one Gibbs block by exploiting the hierarchical structure of NLG-SSSMs. Properly designed MCMC kernels in the new approach target the joint posterior distribution. Therefore, the dependence between the continuous-state and the discrete-regime indicator variables does not affect the mixing properties of the resulting sampler. Furthermore, the proposed methods can be easily applied to general NLG-SSSMs regardless of the Markovian property. All necessary MCMC kernels associated with the proposed PG algorithms are derived for theoretical justification.

The standard PG sampler by Andrieu et al. (2010) is extended to accommodate regime changes in a non-linear dynamic system. This basic algorithm is treated as a benchmark PG algorithm throughout this paper. A modified sequence Monte Carlo (SMC) method is derived to incorporate a regime indicator variable, which targets the joint smoothing distribution of the whole sequence of latent states x_t and s_t. However, the benchmark PG sampler seriously suffers from poor mixing when it is applied to NLG-SSSM, as demonstrated in Section 3. This phenomenon is mainly caused by path degeneracy[5]. The approximate joint smoothing distribution obtained with an SMC method is often unreliable, which produces MCMC output that mixes poorly. In particular, when a dynamic system depends on dramatic regime changes, path degeneracy becomes a serious issue, as shown by Andrieu et al. (2003) and Driessen and Boers (2005). While increasing the number of particles can mitigate path degeneracy, it induces huge computation costs because the modified SMC is to be performed at every MCMC iteration.

Building on the idea of Whiteley (2010), I introduce an alternative PG sampler that is robust to path degeneracy. In the proposed sampler, I implicitly incorporate additional backward recursion to the modified SMC method by employing ancestor sampling as described by Lindsten and Schon (2012) and Lindsten et al. (2014). The ancestor sampling step is designed to increase the number of unique particles by re-shuffling the previous particle trajectories in an existing particle swarm. The PG with ancestor sampling therefore significantly improves the approximation of the joint smoothing distribution of x_t and s_t by preventing path degeneracy. The proposed PG sampler achieves satisfactory mixing with a reasonably small number of particles.

A main limitation of the proposed PG method is that the values of the continuous-state and discrete-regime indicator variables are always restricted to the output of the modified SMC method with ancestor sampling. This restrictive feature eliminates the substantial advantage of the ancestor sampling approach when a degenerate transition equation[6] comprises an NLG-SSSM. Note that the only previous particle trajectory that consists with a particular future particle is that from which the future particle was originally generated if an NLG-SSSM contains a degenerate transition equation. This means that re-sampling previous particle trajectories in the ancestor sampling step is essentially useless because the probability of updating

[5] Path degeneracy refers to a phenomenon whereby particle genealogies coalesce, or degenerate, to a single path.

[6] If a transition density associated with a transition equation describes the probability mass function for a low-dimensional manifold, the transition equation is said to be degenerate.

the particle trajectories is exactly zero at every time period. In this case, the proposed PG sampler with ancestor sampling becomes equivalent to the benchmark PG sampler.

To solve this problem, a further advancement to the proposed PG sampler is made by accommodating particle rejuvenation, as proposed by Lindsten et al. (2014), Carter et al. (2014), and Bunch et al. (2015). The additional particle rejuvenation step generates new values of the latent variables targeting the joint smoothing distribution, which allows for more flexibility for the resulting PG sampler. This novel approach increases the probability of substituting previous particle histories with newly generated values and thus improving the mixing of the resulting Markov chain, even an NLG-SSSM, with a degenerate transition equation. It is shown in Section 3 that the PG sampler with particle rejuvenation vastly outperforms the benchmark PG sampler.

Another goal of this paper is to properly investigate the relationship between volatility and return in the U.S. stock markets in the presence of regime switching by employing the econometric tool developed. Black (1976) and Christie (1982) found empirical evidence that volatility tends to rise in response to bad news on returns but falls in response to good news on returns. This phenomenon is usually explained by the financial leverage effect[7]. Omori et al. (2007) empirically showed that the leverage effect is an important feature of the U.S. stock market by using stochastic volatility models with leverage. In the stochastic volatility literature, the leverage effect is often assumed to be constant, including as described in Omori et al. (2007).

Following Bandi and Renò (2012), I allow for a regime-dependent leverage effect in the context of a discrete time stochastic volatility model. In the proposed stochastic volatility model, the correlation parameter that captures the leverage effect is specified by a function of the regime specific means of log volatility and regime changes that are endogenously estimated within a Bayesian framework. Bandi and Renò (2012), in contrast, arbitrarily chose some deterministic threshold values of the spot volatility to distinguish different regimes in their parametric models. The empirical model is applied to daily S&P 500 and NASDAQ returns from the first week of January 1975 to the first week of August 2015. The Bayesian posterior means of the correlation parameters turn out to be significantly different across high- and low-volatility regimes. In particular, the Bayesian estimates indicate that the stronger (weaker) leverage effect is associated with a high (low)-volatility regime. Based on the Deviance Information Criterion (DIC) by Spiegelhalter et al. (2002), it is shown that the models with the regime-dependent leverage effect are always preferred to those with the constant leverage effect, regardless of the number of regimes. This empirical result confirms the time-varying leverage effect in the U.S. stock market described by Bandi and Renò (2012) within the parametric stochastic volatility models.

[7] Christie (1982) provides a theoretical justification of leverage effect using a Modigliani/Miller economy.

The rest of the paper is organized as follows. In Section 2, I introduce model specification and derive modified sequential Monte Carlo and backward simulation algorithms for a general NLG-SSSM. Section 3 provides details of the proposed PG sampler and illustrates its performance using a simulation study. In Section 4, I demonstrate the proposed technique on data from the U.S. stock market. Concluding remarks are provided in Section 5.

2 Model Specification and Particle Filtering-Smoothing

Non-linear/non-Gaussian Switching State-Space Models (NLG-SSSM) are a class of models in which the structure and the parameters of a non-linear state-space model switch according to discrete latent processes[8]. A state space model consists of the measurement equation $F(.)$ and the transition equation $H(.)$:

$$y_t = F_{s_{0:t}}(x_{0:t}, \epsilon_t) \tag{1}$$

$$x_t = H_{s_{0:t}}(x_{0:t-1}, u_t)$$

where the dynamic system is observed over a time interval $t = 1, 2, ..., T$; $x_t \in \mathbf{X}$ is the unobserved state vector; $Y_t \in \mathbf{Y}$ is the observation vector; $x_{0:t} = \{x_0, x_1, ..., x_t\}$, and $s_{0:t} = \{s_0, s_1, ..., s_t\}$; and u_t and ϵ_t are identically distributed random variables with zero means and are not serially correlated[9]. The properties of the state space model such as dimensions, functional forms, and model parameters shift over time according to a set of discrete latent variables $s_{0:t} = \{s_0, s_1, ..., s_t\}$. The NLG-SSSM is parameterized by unknown parameters β_{s_t}, subject to the discrete latent variable s_t. The latent variable s_t follows a K-state first-order Markov switching process with the following transition probabilities:

$$p(s_t = j|s_{t-1} = k) = \pi_{kj}, \ \sum_{j=1}^{K} \pi_{kj} = 1, \ i, k = 1, 2, ..., K. \tag{2}$$

The model parameters under K-regimes and the transition probabilities are denoted by $\theta = \{\beta_1, \beta_2, ..., \beta_K, \pi\} \in \Theta$. The hierarchical structure of the non-linear/non-Gaussian SSSM specified by equations (1) and (2) is the main difference from that of a conventional non-linear/non-Gaussian state-space model with discrete states. The distributions of the initial states are associated with the prior densities $g_\theta(x_0, s_0) = g_\theta(x_0|s_0)g_\theta(s_0)$. The above NLG-SSSM does not possess the Markovian property. Although the measurement and transition equations often depend on just a few latent states in practice, I adhere to the general model specification throughout this paper for generality of exposition.

[8] The class of Switching State-Space Models is also referred to as Jump Markov Systems in the literature.

[9] The functions $F(.)$ and $H(.)$ can contain additional exogenous variables, but potential exogenous variables are omitted for notational simplicity.

Our primary concern is to perform Bayesian inference in an NLG-SSSM. The two sets of latent variables $x_{0:T} = \{x_0, x_1, ..., x_T\}$, $s_{0:T} = \{s_0, s_1, ..., s_T\}$ and the model parameters θ are treated as unknowns and jointly estimated based on the posterior density given as:

$$p(\theta, x_{0:T}, s_{0:T}|Y_{1:T}) \propto \left[\prod_{t=1}^{T} f_\theta(y_t|x_{0:t}, s_{0:t})g_\theta(x_t|x_{0:t-1}, s_{0:t})g_\theta(s_t|s_{t-1})\right] g_\theta(x_0|s_0)g_\theta(s_0)\pi(\theta) \tag{3}$$

where $f_\theta(.)$ and $g_\theta(.)$ denote probability densities associated with equations (1) and (2), given θ; $\pi(\theta)$ is the prior density of θ. Because the posterior is not available in closed form, Bayesian inference is often infeasible without simulation-based methods.

2.1 Particle Filtering for a Non-linear/Non-Gaussian SSSM

To develop an efficient PMCMC algorithm, it is crucial to sample from the joint smoothing density of the latent state variables $p_\theta(x_{0:T}, s_{0:T}|y_{1:T})$, where $y_{1:T} = \{y_1, y_2, ..., y_T\}$. To obtain the joint smoothing density, consider the following decomposition of the joint filtering density $p_\theta(x_{0:t}, s_{0:t}|y_{1:t})$:

$$\begin{aligned} p_\theta(x_{0:t}, s_{0:t}|y_{1:t}) &= p_\theta(x_t, x_{0:t-1}, s_t, s_{0:t-1}|y_t, y_{1:t-1}) \\ &= \frac{p_\theta(y_t, x_t, x_{0:t-1}, s_t, s_{0:t-1}|y_{1:t-1})}{p_\theta(y_t|y_{1:t-1})} \\ &= \frac{f_\theta(y_t|x_{0:t}, s_{0:t})g_\theta(x_t|x_{1:t-1}, s_{0:t})g_\theta(s_t|s_{t-1})}{p_\theta(y_t|y_{1:t-1})} p_\theta(x_{0:t-1}, s_{0:t-1}|y_{1:t-1}). \end{aligned} \tag{4}$$

Equation (4) shows that the joint filtering density can be defined recursively. Except for a few cases such as linear/Gaussian state space models, the exact joint filtering density cannot be obtained because $p_\theta(x_{0:t-1}, s_{0:t-1}|y_{1:t-1})$ and $f_\theta(y_t|y_{1:t-1})$ are not analytically tractable.

A sequential importance sampling (SIS) algorithm is developed to recursively approximate $p_\theta(x_{0:t}, s_{0:t}|y_{1:t})$ using random samples called 'particles'. A set of particles is denoted by $\{X_{0:t}, S_{0:t}\} = \{x_{0:t}^{(i)}, s_{0:t}^{(i)}\}_{i=1}^N$, in which N represents the total number of particles. The N particles are generated from the following importance distribution in an SIS algorithm:

$$q(x_{0:t}, s_{0:t}) = q(x_t|x_{0:t-1}, s_{0:t})q(s_t|x_{0:t-1}, s_{0:t-1})q(x_{0:t-1}, s_{0:t-1}) \tag{5}$$

where $q(.)$'s denote importance densities possibly depending upon the observation sequence $y_{1:t}$. Usually, $q(x_{0:t-1}, s_{0:t-1})$ is numerically approximated by a Dirac measure $\delta_{\{x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)}\}}(x_{0:t-1}, s_{0:t-1})$ in a SIS algorithm. The Dirac measure places a unit probability mass on each path in $\{X_{0:t-1}, S_{0:t-1}\} = \{x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)}\}_{i=1}^N$ that has been already generated up to time $t-1$. New states $\{X_t, S_t\} = \{x_t^{(i)}, s_t^{(i)}\}_{i=1}^N$ are sequentially generated from $q(s_t|x_{0:t-1}, s_{0:t-1})$ and $q(x_t|x_{0:t-1}, s_{0:t})$ conditional on the corresponding past sequence $\{X_{0:t-1}, S_{0:t-1}\} = \{x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)}\}_{i=1}^{N}$. By combining the new particles at time t with the old particle trajectories at time $t-1$, we obtain a new set of particle paths $\{X_{0:t}^{(i)}, S_{0:t}^{(i)}\} = \{x_{0:t}^{(i)}, s_{0:t}^{(i)}\}_{i=1}^{N}$. A candidate distribution to generate new particles at time t is called an incremental importance distribution.

In practice, the incremental importance distributions $q(x_t|x_{0:t-1}, s_{0:t})$ and $q(s_t|x_{0:t-1}, s_{0:t-1})$ in equation (5) should be well designed to closely approximate the target joint filtering distribution. The simplest approach is to exploit the transition densities associated with equations (1) and (2) and ignore information in the observation sequence $y_{1:t}$:

$$\begin{aligned} q(x_t|x_{0:t-1}, s_{0:t}) &= g_\theta(x_t|x_{0:t-1}, s_{0:t}), \\ q(s_t|x_{0:t-1}, s_{0:t-1}) &= g_\theta(s_t|s_{t-1}). \end{aligned} \tag{6}$$

In the Appendix, the optimal incremental importance distributions are derived, including all relevant information to obtain the closest approximation to the target distribution. The optimal incremental importance distributions can be constructed using a modified unscented Kalman filter[10] (UKF) by Andrieu et al. (2003). However, I confirm via a simulation that the computational costs of sampling from the optimal importance distributions far exceed its benefits, especially when combined with a PMCMC sampler. Therefore, the incremental importance distributions in equation (6) are employed in forward filtering for all simulations and applications throughout this paper.

As an importance distribution is usually not identical to the target distribution, we need to correct the corresponding approximations by imposing the following importance weights to generated particles:

$$\begin{aligned} \omega_t^{(i)} &= \frac{p_\theta(x_{0:t}^{(i)}, s_{0:t}^{(i)}|y_{1:t})}{q(x_{0:t}^{(i)}, s_{0:t}^{(i)})} \\ &= \frac{f_\theta(y_t|x_{0:t}^{(i)}, s_{0:t}^{(i)}) g_\theta(x_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t}^{(i)}) g_\theta(s_t^{(i)}|s_{t-1}^{(i)})}{p_\theta(y_t|y_{1:t-1}) q(x_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t}^{(i)}) q(s_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)})} \frac{p_\theta(x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)}|y_{1:t-1})}{q(x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)})} \\ &\propto \frac{f_\theta(y_t|x_{0:t}^{(i)}, s_{0:t}^{(i)}) g_\theta(x_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t}^{(i)}) g_\theta(s_t^{(i)}|s_{t-1}^{(i)})}{q(x_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t}^{(i)}) q(s_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)})} \omega_{t-1}^{(i)}. \end{aligned} \tag{7}$$

The so-called incremental importance weight $\bar{\omega}_t^{(i)}$ is defined as:

$$\bar{\omega}_t^{(i)} = \frac{f_\theta(y_t|x_{0:t}^{(i)}, s_{0:t}^{(i)}) g_\theta(x_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t}^{(i)}) g_\theta(s_t^{(i)}|s_{t-1}^{(i)})}{q(x_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t}^{(i)}) q(s_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)})}.$$

[10] As pointed out by many authors, linear/Gaussian approximation to a general non-linear/non-Gaussian state-space model through UKF is not accurate when the non-linearity is severe. The approximation errors by UKF quickly accumulate as the sample size increases. Wan and van der Merwe (2001) therefore suggested using UKF to design the importance distribution of a particle filter. They also empirically show that the resulting particle filtering algorithm performs well in capturing unobserved states and estimating model parameters.

Because the importance weight $\omega_t^{(i)}$ is only proportional to $\bar{\omega}_t^{(i)}\omega_{t-1}^{(i)}$ due to the unknown normalizing constant $p_\theta(y_t|y_{1:t-1})$, $\bar{\omega}_t^{(i)}\omega_{t-1}^{(i)}$ is self-normalized as:

$$\hat{\omega}_t^{(i)} = \frac{\bar{\omega}_t^{(i)}\omega_{t-1}^{(i)}}{\sum_{j=1}^N \bar{\omega}_t^{(j)}\omega_{t-1}^{(j)}}$$

which yields our estimate of the importance weight at time t. Additionally, note that the normalizing constant can be approximated by:

$$\hat{p}_\theta(y_t|y_{t-1}) = \sum_{i=1}^N \bar{\omega}_t^{(i)}\omega_{t-1}^{(i)}.$$

A critical problem of the SIS algorithm is weight degeneracy[11]. Gordon et al. (1993) originally developed a standard particle filter to solve weight degeneracy by including a resampling step in which N random particles $\{\tilde{x}_{0:t}^{(i)}, \tilde{s}_{0:t}^{(i)}\}_{i=1}^N$ are re-drawn from the existing particles $\{x_{0:t}^{(i)}, s_{0:t}^{(i)}\}_{i=1}^N$ according to the normalized importance weight $\{\hat{\omega}_t^{(i)}\}_{i=1}^N$. The additional resampling step replicates particles with high importance weights while removing particles with low importance weights to prevent weight degeneracy. It is worth mentioning that the standard particle filter described by Gordon et al. (1993) can be considered a special case of the auxiliary particle filter of Pitt and Shephard (1999)[12]. These particle filters are also known as sequential Monte Carlo (SMC) methods. Because the resampling step allows us to obtain equally weighted particles approximately distributed from $p_\theta(x_{0:t}, s_{0:t}|y_{1:t})$, a new set of weights $\{\tilde{\omega}_t^{(i)} = \frac{1}{N}\}_{i=1}^N$ is assigned to resampled particles $\{\tilde{x}_{0:t}^{(i)}, \tilde{s}_{0:t}^{(i)}\}_{i=1}^N$. The joint smoothing density $p_\theta(x_{0:T}, s_{0:T}|y_{1:T})$ of interest can be obtained by the recursive structure in equations (4) and (5) as $t = T$. In what follows, I provide a summary of an SMC algorithm for NLG-SSSM.

Algorithm 1-1: Sequential Monte Carlo (SMC)

i) Draw $\{s_0^{(i)}\}_{i=1}^N$ from $q(s_0)$ and draw $\{x_0^{(i)}\}_{i=1}^N$ from $q(x_0|s_0^{(i)})$. Save the normalized importance weights $\{\hat{\omega}_0^{(i)} = \frac{\bar{\omega}_0^{(i)}}{\sum_{j=1}^N \bar{\omega}_0^{(i)}}\}_{i=1}^N$ where $\bar{\omega}_0^{(i)} = \frac{p_\theta(x_0^{(i)}|s_0^{(i)})p_\theta(s_0^{(i)})}{q(x_0^{(i)}|s_0^{(i)})q(s_0^{(i)})}$.

- Iterate step ii), iii), and vi) for $t = 1, 2, ..., T$.

ii) Resample N particles $\{\tilde{x}_{0:t-1}^{(i)}, \tilde{s}_{0:t-1}^{(i)}\}_{i=1}^N$ from $\{x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)}\}_{i=1}^N$ with probability $\{\hat{\omega}_{t-1}^{(i)}\}_{i=1}^N$ and assign new importance weights $\{\tilde{\omega}_{t-1}^{(i)} = \frac{1}{N}\}_{i=1}^N$. Rename the particles $\{\tilde{x}_{0:t-1}^{(i)}, \tilde{s}_{0:t-1}^{(i)}\}_{i=1}^N$ into $\{x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)}\}_{i=1}^N$ and the importance weights $\{\tilde{\omega}_{t-1}^{(i)}\}_{i=1}^N$ into $\{\omega_{t-1}^{(i)}\}_{i=1}^N$.

[11] Weight degeneracy is a phenomenon that most of the particles $\{x_{0:t}^{(i)}, s_{0:t}^{(i)}\}_{i=1}^N$ diverge from their true latent states over time, increasing the variance of importance weights, and eventually, all but one of the importance weights converge to zero.

[12] The auxiliary particle filter uses updated importance weights with information on y_{t+1} in resampling x_t and s_t. However, this article does not explicitly attempt to implement the auxiliary particle filter due to high computational costs of estimating NLG-SSSM.

iii) Draw $\{s_t^{(i)}\}_{i=1}^N$ from $g(s_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)})$ and draw $\{x_t^{(i)}\}_{i=1}^N$ from $g(x_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t}^{(i)})$. Set $\{x_{0:t}^{(i)}\}_{i=1}^N = \{x_{0:t-1}^{(i)}, x_t^{(i)}\}_{i=1}^N$ and $\{s_{0:t}^{(i)}\}_{i=1}^N = \{s_{0:t-1}^{(i)}, s_t^{(i)}\}_{i=1}^N$.

vi) Calculate the unnormalized weights: $\bar{\omega}_t^{(i)}\hat{\omega}_{t-1}^{(i)} = \frac{f_\theta(y_t|x_{0:t}^{(i)}, s_{0:t}^{(i)})p_\theta(x_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t}^{(i)})p_\theta(s_t^{(i)}|s_{t-1}^{(i)})}{q(x_t^{(i)}|s_{0:t-1}^{(i)}, s_{0:t}^{(i)})q(s_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)})}\hat{\omega}_{t-1}^{(i)}$ and obtain the normalized weights: $\hat{\omega}_t^{(i)} = \frac{\bar{\omega}_t^{(i)}\hat{\omega}_{t-1}^{(i)}}{\sum_{j=1}^N \bar{\omega}_t^{(i)}\hat{\omega}_{t-1}^{(i)}}$.

In the proposed SMC method, the importance sampling is repeatedly operated at each time period to generate various particle realizations $\{x_{0:T}^{(i)}, s_{0:T}^{(i)}\}_{i=1}^N$ from $p_\theta(x_{0:T}, s_{0,T}|y_{1:T})$. The target joint smoothing distribution is approximated by:

$$p_\theta(x_{0:T}, s_{0,T}|y_{1:T}) \approx \sum_{i=1}^N \hat{\omega}_T^{(i)} \delta_{\{x_{0:T}^{(i)}, s_{0:T}^{(i)}\}}(x_{0:T}, s_{0:T})$$

where $\delta_{\{x_{0:T}^{(i)}, s_{0:T}^{(i)}\}}(x_{0:T}, s_{0:T})$ places a unit probability mass on each path of $\{x_{0:T}^{(i)}, s_{0:T}^{(i)}\}_{i=1}^N$. Accordingly, we draw M particle trajectories from $\{x_{0:T}^{(i)}, s_{0:T}^{(i)}\}_{i=1}^N$ with the normalized weight $\{\hat{\omega}_T^{(i)}\}_{i=1}^N$ to simulate from the joint smoothing distribution $p_\theta(x_{0:T}, s_{0,T}|y_{1:T})$.

Algorithm 1-2: Forward Filtering for $p_\theta(x_{0:T}, s_{0:T}|y_{1:T})$

- Run *Algorithm 1-1* **(SMC algorithm)** and save the particle set $\{x_{0:T}^{(i)}, s_{0:T}^{(i)}\}_{i=1}^N$ along with the normalized importance weights $\{\hat{\omega}_T^{(i)}\}_{i=1}^N$ at time T.

i) Draw $\{\tilde{x}_{0:T}^{(j)}, \tilde{s}_{0:T}^{(j)}\}_{j=1}^M$ from $\{x_{0:T}^{(i)}, s_{0:T}^{(i)}\}_{i=1}^N$ according to the normalized importance weights $\{\hat{\omega}_T^{(i)}\}_{i=1}^N$.

2.2 Backward Smoothing for a Non-linear/Non-Gaussian SSSM

The approximate joint smoothing distribution obtained using an SMC algorithm is often unreliable due to path degeneracy. For instance, when re-sampling x_t and s_t in the SMC method, we discard many past trajectories in $\{x_{0:t}^{(i)}, s_{0:t}^{(i)}\}_{i=1}^N$, decreasing the number of unique particles at each time period. Consequently, the resulting particles $\{x_{0:t}^{(i)}, s_{0:t}^{(i)}\}_{i=1}^N$ share just a few common ancestors as t increases. This inevitably leads to a poor approximation of the joint smoothing distribution $p_\theta(x_{0:T}, s_{0:T}|y_{1:T})$. As regimes change more frequently or the number of regimes is large, the path degeneracy problem gets worse, as shown by Andrieu et al. (2003) and Driessen and Boers (2005). Even if an increase in the number of particles can mitigate path degeneracy, huge computation costs are required for a PMCMC algorithm.

Based on the ideas of Godsill et al.(2004), we can effectively address the problem of path degeneracy by complementing forward filtering with additional backward smoothing for NLG-SSSM. Consider the following factorization for backward smoothing:

$$p_\theta(x_{0:T}, s_{0:T}|y_{1:T}) = p_\theta(x_T, s_T|y_{1:T}) \prod_{t=0}^{T-1} p_\theta(x_t, s_t|y_{1:T}, x_{t+1:T}, s_{t+1:T}) \tag{8}$$

Theoretically, the above decomposition suggests that one can sequentially generate x_T, s_T from $p_\theta(x_T, s_T|y_{1:T})$ and then x_t, s_t from $p_\theta(x_t, s_t|x_{t+1:T}, s_{t+1:T}, y_{1:T})$ for $t = T-1, ..., 1, 0$. The conditional density at time t can be decomposed as:

$$\begin{aligned} p_\theta(x_t, s_t|y_{1:T}, x_{t+1:T}, s_{t+1:T}) &= p_\theta(x_t, s_t|y_{1:t}, y_{t+1:T}, x_{t+1:T}, s_{t+1:T}) \\ &= \frac{p_\theta(y_{t+1:T}, x_t, s_t|y_{1:t}, x_{t+1:T}, s_{t+1:T})}{p_\theta(y_{t+1:T}|y_{1:t}, x_{t+1:T}, s_{t+1:T})} \\ &\propto p_\theta(y_{t+1:T}|y_{1:t}, x_{t:T}, s_{t:T}) p_\theta(x_t, s_t|y_{1:t}, x_{t+1:T}, s_{t+1:T}) \\ &= p_\theta(y_{t+1:T}|y_{1:t}, x_{t:T}, s_{t:T}) \frac{p_\theta(x_t, x_{t+1:T}, s_t, s_{t+1:T}|y_{1:t})}{p_\theta(x_{t+1:T}, s_{t+1:T}|y_{1:t})} \\ &\propto p_\theta(y_{t+1:T}|y_{1:t}, x_{t:T}, s_{t:T}) p_\theta(x_{t:T}, s_{t:T}|y_{1:t}) \\ &= p_\theta(y_{t+1:T}|y_{1:t}, x_{t:T}, s_{t:T}) g_\theta(x_{t+1:T}, s_{t+1:T}|x_t, s_t) p_\theta(x_t, s_t|y_{1:t}) \end{aligned} \tag{9}$$

where $p_\theta(x_{t+1:T}, s_{t+1:T}|x_t, s_t) = [\prod_{\tau=t}^{T} g_\theta(x_\tau|x_{t:\tau-1}, s_{t:\tau})] g_\theta(s_{t+1}|s_t)$ due to the hierarchical structure of NLG-SSSM.

As shown in equation (9), the smoothing recursion requires the joint marginal filtering density $p_\theta(x_t, s_t|y_{1:t})$. The SMC algorithm introduced in the previous section can provide a numerical approximation of $p_\theta(x_t, s_t|y_{1:t})$ as a direct application. The joint marginal density $p_\theta(x_t, s_t|y_{1:t})$ is given by:

$$p_\theta(x_{0:t}, s_{0:t}|y_{1:t}) = \sum_{s_{0:t-1}} \int p_\theta(x_{0:t}, s_{0:t}|y_{1:t}) dx_{0:t-1}.$$

In practice, integrating over the all the past states can be easily done by simply discarding $\{x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)}\}_{i=1}^N$ up to time $t-1$ and keeping only $\{x_t^{(i)}, s_t^{(i)}\}_{i=1}^N$ at time t with the normalized importance weights $\{\hat{\omega}_t^{(i)}\}_{i=1}^N$. The saved particles and importance weights approximate the joint marginal density $p_\theta(x_t, s_t|y_{1:t})$:

$$p_\theta(x_t, s_t|y_{1:t}) \approx \sum_{i=1}^N \hat{\omega}_t^{(i)} \delta_{\{x_t^{(i)}, s_t^{(i)}\}}(x_t, s_t)$$

where $\delta_{\{x_t^{(i)}, s_t^{(i)}\}}(x_t, s_t)$ is the Dirac measure and $\hat{\omega}_t^{(i)}$ is the normalized weight attached to particles $x_t^{(i)}$ and $s_t^{(i)}$.

Particles at time t are updated conditional on $x_{t+1:T}$ and $s_{t+1:T}$ according to equation (9) using additional importance sampling and resampling steps as follows:

$$p_\theta(x_t, s_t|x_{t+1:T}, s_{t+1:T}, y_{1:T}) \approx \sum_{i=1}^N \hat{\omega}_{t|T}^{(i)} \delta_{\{x_t^{(i)}, s_t^{(i)}\}}(x_t, s_t). \tag{10}$$

The modified importance weight $\hat{\omega}_{t|T}^{(i)}$ is defined as:

$$\hat{\omega}_{t|T}^{(i)} = \frac{p_\theta(y_{t+1:T}|y_{1:t}, x_{t:T}^{(i)}, s_{t:T}^{(i)}) p_\theta(x_{t+1:T}^{(i)}, s_{t+1:T}^{(i)}|x_t^{(i)}, s_t^{(i)})\ \hat{\omega}_t^{(i)}}{\sum_{j=1}^N p_\theta(y_{t+1:T}|y_{1:t}, x_{t:T}^{(j)}, s_{t:T}^{(j)}) p_\theta(x_{t+1:T}^{(j)}, s_{t+1:T}^{(j)}|x_t^{(j)}, s_t^{(j)})\ \hat{\omega}_t^{(j)}}$$

where $p_\theta(x_{t+1:T}, s_{t+1:T}|x_t, s_t) = [\prod_{\tau=t}^T g_\theta(x_\tau|x_{t:\tau-1}, s_{t:\tau})] g_\theta(s_{t+1}|s_t)$. The empirical distribution in equation (10) is employed to generate particles $\{\tilde{x}_t^{(i)}, \tilde{s}_t^{(i)}\}_{i=1}^M$ sequentially backward in time conditional on $\{x_{t+1:T}^{(i)}, s_{t+1:T}^{(i)}\}_{i=1}^M$ and $y_{1:T}$. The following is the summary of the backward simulation for NLG-SSSM.

Algorithm 1-3: Backward Smoothing for $p_\theta(x_{0:T}, s_{0:T}|y_{1:T})$

- Run *Algorithm 1-1* **(SMC algorithm)** and save the particle set $\{x_t^{(i)}, s_t^{(i)}\}_{i=1}^N$ along with the normalized importance weights $\{\hat{\omega}_t^{(i)}\}_{i=1}^N$ for $t = 1, 2, ..., T$.

i) Draw $\{\tilde{x}_T, \tilde{s}_T\}$ from $\{x_T^{(i)}, s_T^{(i)}\}_{i=1}^N$ with the normalized importance weights $\{\hat{\omega}_T^{(i)}\}_{i=1}^N$.

- Iterate step ii), and iii) for $t = T-1, T-2, .., 0$

ii) Calculate the modified normalized weights $\hat{\omega}_{t|T}^{(i)}$ conditional on $\tilde{x}_{t+1:T}$ and $\tilde{s}_{t+1:T}$.

iii) Draw $\tilde{x}_t, \tilde{s}_t$ from $\{x_t^{(i)}, s_t^{(i)}\}_{i=1}^N$ according to the modified importance weight $\{\hat{\omega}_{t|T}^{(i)}\}_{i=1}^N$.

- Repeat step i),ii), and iii) M times and save $\{\tilde{x}_{0:T}^{(i)}, \tilde{s}_{0:T}^{(i)}\}_{i=1}^M$.

The backward smoothing algorithm can be operated to generate various particle realizations $\{\tilde{x}_{0:T}^{(i)}, \tilde{s}_{0:T}^{(i)}\}_{i=1}^M$ from the joint smoothing distribution $p_\theta(x_{0:T}, s_{0,T}|y_{1:T})$. As briefly discussed in the previous section, the forward filtering algorithm to approximate $p_\theta(x_{0:T}, s_{0,T}|y_{1:T})$ seriously suffers from path degeneracy. The backward smoothing algorithm, however, is free from path degeneracy because it exploits al ofl the generated particles at each time, going backward in time. This superior feature of the backward smoothing algorithm is the key to successfully developing an efficient PG sampler in the next section.

3 Particle Markov Chain Monte Carlo Methods for a non-linear/non-Gaussian SSSM

To simulate the posterior distribution $p(\theta, x_{0:T}, s_{0:T}|y_{1:T})$, one may attempt to alternatively sample θ from $p(\theta|x_{0:T}, s_{0:T}, y_{1:T})$, $x_{0:T}$ from $p(x_{0:T}|s_{0:T}, \theta, y_{1:T})$ and $s_{0:T}$ from $p(s_{0:T}|\theta, x_{0:T}, y_{1:T})$. While this conventional Gibbs sampling approach seems straightforward to implement, there are some practical problems due to the presence of the regime indicator variable.

First, a multi-move sampler is not available for drawing $s_{0:T}$ from $p(s_{0:T}|\theta, x_{0:T}, y_{1:T})$ when the path dependence problem is encountered in the transition and measurement equations. Note that the current observation and the continuous state in equation (1) are dependent on the entire sequence of the regime indicator variable up to time t. As the regime indicator variables are unobservable, we need to integrate over all possible regime paths when computing the likelihood, which is essential for a multi-move sampler. However, as the number of possible cases increases exponentially with t, evaluating the likelihood is not feasible in this case. A single-move sampler provides another option. However, as shown in Liu et al. (1994), Scott (2002), and Kim and Kim (2015), it is hard to obtain a correct stationary distribution via a single-move sampler if the regime indicator variable is very persistent or has absorbing states. In Section 2.4, it will be shown that the proposed PG sampler in this article produces reliable Bayesian estimates and achieves fast mixing even for an NLG-SSSM with the path dependence feature.

Second, the MCMC transition kernel becomes degenerate when the continuous state and the discrete regime indicator variables are perfectly correlated. A prominent example is provided by the Bayesian change-point models in Pesaran et al. (2006) and Koop and Potter (2007). As $x_{0:T}$ is generated in a block conditional on $s_{0:T}$ and $y_{1:T}$ and then $s_{0:T}$ is generated conditional on $x_{0:T}$ and $y_{1:T}$, this sampling scheme is degenerate. This is because x_t does not change if $s_t = 0$; conversely, if x_t is constant, then s_t is generated to be 0.

All these practical problems are solved by draw $x_{0:T}$ and $s_{0:T}$ from the joint smoothing distribution $p(x_{0:T}, s_{0:T}|y_{1:T}, \theta)$ using Gibbs sampling methods. The main difficulties in deriving a proper Gibbs sampler are that the joint smoothing distribution shows complex patterns of dependence among the latent variables, and sampling $\{x_{0:T}, s_{0:T}\}$ directly from the joint smoothing distribution $p(x_{0:T}, s_{0:T}|y_{1:T}, \theta)$ is not possible in general as a result of non-linearity and non-Gaussianity. I adopt a PG sampling approach to estimate NLG-SSSM following Andrieu et al. (2010) and illustrate that the proposed PG sampler performs well in various cases. Like any other Gibbs samplers, the PG sampling method does not require additional accept/reject steps, which produce mixing properties that are better than those of particle Metropolis-Hastings samplers.

For a valid particle approximation to a Gibbs sampler, I use an artificial target distribution $\Psi(.)$ that incorporates all of the randomness generated by an SMC method. If the new extended target distribution $\Psi(.)$ admits the original posterior $p(\theta, x_{0:T}, s_{0:T}|y_{1:T})$ as a marginal, a valid Gibbs sampler can be designed by $\Psi(.)$. For this purpose, the particle Gibbs sampler is augmented by auxiliary variables to capture the additional randomness of an SMC method:

$$\mathbf{A}_t = \{a_t^{(i)}\}_{i=1}^N.$$

The so-called ancestor index $a_t^{(i)} \in \{1, 2, ..., N\}$ is the index variable of the ancestor at time $t-1$ of i-th particles $\{x_t^{(i)}, s_t^{(i)}\}$. For example, if $x_{t-1}^{(5)}$ and $s_{t-1}^{(5)}$ are drawn for $x_t^{(i)}$ and $s_t^{(i)}$ in the resampling step of an SMC method, the index variable yields $a_t^{(i)} = 5$. Using the ancestor index, entire particle trajectories are constructed by tracing back to their ancestral lineages recursively:

$$x_{0:t}^{(i)} = \{x_{0:t-1}^{(a_t^{(i)})}, x_t^{(i)}\},\ s_{0:t}^{(i)} = \{s_{0:t-1}^{(a_t^{(i)})}, s_t^{(i)}\}\ for\ i = 1, 2, ..., N.$$

Now, let $K \in \{1, 2, ..., N\}$ be the index of a fixed reference trajectory. We can keep track of its ancestral lineage as follows:

$$x_{0:T}^{(K)} = \{x_{0:T-1}^{(a_T^{(K)})}, x_T^{(K)}\} = \{x_{0:T-2}^{(a_{T-1}^{(a_T^{(K)})})}, x_{T-1}^{(a_T^{(K)})}, x_T^{(K)}\} = ...$$

$$s_{0:T}^{(K)} = \{s_{0:T-1}^{(a_T^{(K)})}, s_T^{(K)}\} = \{s_{0:T-2}^{(a_{T-1}^{(a_T^{(K)})})}, s_{T-1}^{(a_T^{(K)})}, s_T^{(K)}\} = ...$$

For the fixed reference trajectory, an additional index b_t is used for notational simplicity. The index for the reference trajectory is defined as:

$$x_{0:T}^{(K)} = x_{0:T}^{(b_{0:T})} = \{x_0^{(b_0)}, x_1^{(b_1)}, ..., x_{T-1}^{(b_{T-1})}, x_T^{(b_T)}\}$$

$$s_{0:T}^{(K)} = s_{0:T}^{(b_{0:T})} = \{s_0^{(b_0)}, s_1^{(b_1)}, ..., s_{T-1}^{(b_{T-1})}, s_T^{(b_T)}\}$$

According to the definition of b_t, we can rewrite the index b_t in terms of the ancestor index as $b_t = K$ for $t = T$ and $b_t = a_{t+1}^{(b_{t+1})}$ for $t = 0, ..., T-1$. The introduced indices are auxiliary variables in an SMC sampler and will play a key role later in deriving valid MCMC transition kernels. Finally, the remaining latent states except the reference trajectory are denoted by $X_{0:T}^{(-b_{0:T})}$ and $S_{0:T}^{(-b_{0:T})}$.

Using the ancestor index variables, the density of the SMC in Section 2.1 is defined as:

$$\begin{aligned}\Phi(X_{0:T}, S_{0:T}, A_{1:T}|\theta) &= \prod_{i=1}^{N} q(x_0^{(i)}|s_0^{(i)})q(s_0^{(i)}) \prod_{t=1}^{T} \Bigg[\prod_{i=1}^{N} \frac{\bar{\omega}_{t-1}^{(i)}}{\sum_j \bar{\omega}_{t-1}^{(j)}} q(x_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t}^{(i)}) q(s_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})}) \Bigg] \\ &= \prod_{i=1}^{N} q(x_0^{(i)}|s_0^{(i)})q(s_0^{(i)}) \prod_{t=1}^{T} \Bigg[\prod_{i=1}^{N} M_t^{\theta}(a_t^{(i)}, x_t^{(i)}, s_t^{(i)}) \Bigg]\end{aligned} \tag{12}$$

in which $M_t^{\theta}(a_t^{(i)}, x_t^{(i)}, s_t^{(i)})$ is given as follows:

$$M_t^{\theta}(a_t^{(i)}, x_t^{(i)}, s_t^{(i)}) = \frac{\bar{\omega}_{t-1}^{(i)}}{\sum_j \bar{\omega}_{t-1}^{(j)}} q(x_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t}^{(i)}) q(s_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})}),$$

and $X_{0:T} = \{x_{0:T}^{(i)}\}_{i=1}^{N}$; $S_{0:T} = \{s_{0:T}^{(i)}\}_{i=1}^{N}$; $A_{1:T} = \{a_{1:T}^{(i)}\}_{i=1}^{N}$; $q(.)$ denote importance densities that may depend on the observation sequence $y_{1:t}$. The unnormalized importance weights in equation (12) are given as follows:

$$\bar{\omega}_t^{(i)} = \frac{f_\theta(y_t|x_{0:t}^{(i)}, s_{0:t}^{(i)}) p_\theta(x_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t}^{(i)}) p_\theta(s_t^{(i)}|s_{t-1}^{(i)})}{q(x_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t}^{(i)}) q(s_t^{(i)}|x_{0:t-1}^{(i)}, s_{0:t-1}^{(i)})}.$$

I note that the normalized importance weight $\hat{\omega}_t^{(i)} = \frac{\bar{\omega}_t^{(i)}\hat{\omega}_{t-1}^{(i)}}{\sum_j \bar{\omega}_t^{(j)}\hat{\omega}_{t-1}^{(j)}}$ can be rewritten as $\hat{\omega}_t^{(i)} = \frac{\bar{\omega}_t^{(i)}}{\sum_j \bar{\omega}_t^{(j)}}$ because we assign $\frac{1}{N}$ to $\hat{\omega}_{t-1}^{(i)}$ after resampling. Similarly, we can easily determine the conditional density of the SMC given a reference trajectory $x_{0:T}^{(b_{0:T})}$ and $s_{0:T}^{(b_{0:T})}$:

$$\begin{aligned}\Phi(X_{0:T}^{(-b_{0:T})}, &S_{0:T}^{(-b_{0:T})}, A_{1:T}^{(-b_{1:T})}|\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T}) \\ &= \frac{\Phi(X_{0:T}, S_{0:T}, A_{1:T}|\theta)}{q(x_0^{(b_0)}|s_0^{(b_0)})q(s_0^{(b_0)})\prod_{t=1}^T \left[\frac{\bar{\omega}_{t-1}^{(b_t)}}{\sum_j \bar{\omega}_{t-1}^{(j)}} q(x_t^{(b_t)}|x_{0:t-1}^{(b_{0:t-1})}, s_{0:t}^{(b_{0:t})})q(s_t^{(b_t)}|x_{0:t-1}^{(b_{0:t-1})}, s_{0:t-1}^{(b_{0:t-1})})\right]} \\ &= \prod_{\substack{i=1\\ i\neq b_0}}^N q(x_0^{(i)}|s_0^{(i)})q(s_0^{(i)}) \times \prod_{t=1}^T \left[\prod_{\substack{i=1\\ i\neq b_t}}^N M_t^\theta(a_t^{(i)}, x_t^{(i)}, s_t^{(i)})\right]\end{aligned} \tag{13}$$

The extended target distribution to construct valid MCMC kernels is given by:

$$\begin{aligned}\Phi(\theta, X_{0:T}, S_{0:T}, A_{1:T}, K) &\equiv \Phi(\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T})\Phi(X_{0:T}^{(-b_{0:T})}, S_{0:T}^{(-b_{0:T})}, A_{1:T}^{(-b_{0:T})}|\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T}) \\ &\equiv \frac{1}{N^{T+1}} p(\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}|y_{1:T}) \\ &\times \prod_{\substack{i=1\\ i\neq b_0}}^N q(x_0^{(i)}|s_0^{(i)})q(s_0^{(i)}) \times \prod_{t=1}^T \left[\prod_{\substack{i=1\\ i\neq b_0}}^N M_t^\theta(a_t^{(i)}, x_t^{(i)}, s_t^{(i)})\right]\end{aligned} \tag{14}$$

where $X_{0:T} = \{x_{0:T}^{(b_{0:T})}, X_{0:T}^{(-b_{0:T})}\}$; $S_{0:T} = \{s_{0:T}^{(b_{0:T})}, S_{0:T}^{(-b_{0:T})}\}$; $K \in \{1, 2, ..., N\}$ is the index of a reference trajectory. A particle Gibbs sampler will be developed in this section to estimate NLG-SSSM by targeting the extended target distribution. The suggested sampling scheme will have $p(\theta, x_{0:T}, s_{0:T}|y_{1:T})$ as the stationary distribution by marginalizing over all the auxiliary variables by an SMC algorithm, as shown in Andrieu et al. (2010).

3.1 Benchmark Particle Gibbs

We are interested in sampling from $p(\theta, x_{0:T}, s_{0:T}|y_{1:T})$ based on a Particle Gibbs (PG) sampler. The main difficulty of the Bayesian inference is that the target density has a high-dimensional parameter space $\Theta \times \mathbf{X}^{(T+1)} \times \mathbf{S}^{(T+1)}$. This problem can be alleviated by building a multi-stage Gibbs sampler including the auxiliary variables as suggested by Andrieu et al. (2010). In what follows, I provide details of the benchmark PG sampler to estimate NLG-SSSM.

The first step of the benchmark PG sampler is to sample the index K of a reference trajectory. This is exactly the same as drawing one particle trajectory from all generated particle trajectories using an SMC method. The conditional distribution for the index K is given by:

$$\Phi(K|\theta, X_{0:T}, S_{0:T}, A_{1:T}) = \frac{\bar{w}_T^{(K)}}{\sum_{j=1}^{N} \bar{w}_T^{(j)}} \tag{15}$$

based on the following *Proposition 1*.

Proposition 1 *The conditional $\Phi(K|\theta, X_{0:T}, S_{0:T}, A_{1:T})$ under the extended target $\Phi(\theta, X_{0:T}, S_{0:T}, A_{1:T}, K)$ is proportional to the importance weight at T:*

$$\Phi(K|\theta, X_{0:T}, S_{0:T}, A_{1:T}) \propto \bar{w}_T^{(K)}.$$

The proof of *Proposition 1* is given in the Appendix. Based on *Proposition 1*, it is straightforward to sample a reference index K from its conditional in equation (15).

In the second step of the standard PG sampler, we sample θ based on a partially collapsed Gibbs step, which means that some of the random variables are marginalized before conditioning. It does not violate the invariance of the corresponding sampler. For more details, see van Dyk and Park (2008). From constructing he extended target distribution, the conditional distribution for θ is given by:

$$\Phi(\theta|x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T}) = p(\theta|x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, y_{1:T}) \tag{16}$$

Note that in practice, sampling θ from $p(\theta|x_{0:T}, s_{0:T}, y_{1:T})$ is much simpler than sampling θ conditional only on the observations $y_{1:T}$[13]. I assume that sampling θ from its conditional distribution under $\Phi(.)$ is straightforward by either using conjugate priors or Metropolis-Hastings given $x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}$.

The conditional distribution for the third step of the benchmark PG sampler is given in equation (13). More specifically, we generate $N-1$ particle trajectories conditional on θ and a reference trajectory from the conditional distribution $\Phi(X_{0:T}^{(-b_{0:T})}, S_{0:T}^{(-b_{0:T})}, A_{1:T}^{(-b_{1:T})}|\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T})$. To achieve this goal, we employ a so-called conditional SMC algorithm. Simply speaking, the conditional SMC method is an algorithm that generates $N-1$ particles with the reference trajectory $\{x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}\}$ fixed throughout the sampling process. Before introducing the conditional SMC sampler, it is worth mentioning that the actual values of the indices $b_{0:T}$ are not important et all, and therefore, we can assign an alternative sequence to $b_{0:T} = \{N, N, ..., N\}$ as a matter of convenience. This is because the index sequence $b_{0:T}$ is just a convenient tool to keep track of the past latent variables in entire particle sets. The following algorithm is the conditional SMC method used in the benchmark PG sampler.

[13] For instance, the transition probabilities for s_t can be easily generated from the beta distributions when using conjugate priors. When non-conjugate priors are used or conditional posteriors do not belong to well-known distributions for some parameters, we can employ Metropolis-Hastings algorithms within a Particle Gibbs sampling approach conditional on the latent states.

Algorithm 2-1: Conditional Sequential Monte Carlo (CSMC)

i) Draw $\{s_0^{(i)}\}_{i=1}^{N-1}$ from $q(s_0)$ and draw $\{x_0^{(i)}\}_{i=1}^{N-1}$ from $q(x_0|s_0^{(i)})$ sequentially. Set $\{x_0^{(N)}, s_0^{(N)}\} = \{x_0^{(b_0)}, s_0^{(b_0)}\}$. Save the normalized importance weights $\{\hat{\omega}_0^{(i)} = \frac{\bar{\omega}_0^{(i)}}{\sum_{j=1}^{N} \bar{\omega}_0^{(i)}}\}_{i=1}^{N}$ where $\bar{\omega}_0^{(i)} = \frac{p_\theta(x_0^{(i)}|s_0^{(i)})p_\theta(s_0^{(i)})}{q(x_0^{(i)}|s_0^{(i)})q(s_0^{(i)})}$.

- Iterate step ii), iii), and vi) for $t = 1, 2, ..., T$.

ii) Draw ancestor indices $\{a_t^{(i)}\}_{i=1}^{N-1}$ with probability $\{\hat{\omega}_{t-1}^{(i)}\}_{i=1}^{N}$. Draw $\{s_t^{(i)}\}_{i=1}^{N-1}$ from $g(s_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})})$ and $\{x_t^{(i)}\}_{i=1}^{N-1}$ from $g(x_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})}, s_t^{(i)})$ sequentially.

iii) Set $a_t^{(N)} = N$ and $\{x_t^{(N)}, s_t^{(N)}\} = \{x_t^{(b_t)}, s_t^{(b_t)}\}$. New trajectories are set by $x_{0:t}^{(i)} = \{x_{0:t-1}^{(a_t^{(i)})}, x_t^{(i)}\}$ and $s_{0:t}^{(i)} = \{s_{0:t-1}^{(a_t^{(i)})}, s_t^{(i)}\}$ for $i = 1, 2, ..., N$.

vi) Calculate the unnormalized weights: $\bar{\omega}_t^{(i)} = \frac{f_\theta(y_t|x_{0:t}^{(i)}, s_{0:t}^{(i)}, y_{1:t-1})p_\theta(x_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t}^{(i)})p_\theta(s_t^{(i)}|s_{t-1}^{(a_t^{(i)})})}{q(x_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t}^{(i)})q(s_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})})}$.

and obtain the normalized weights: $\hat{\omega}_t^{(i)} = \frac{\bar{\omega}_t^{(i)}}{\sum_{j=1}^{N} \bar{\omega}_t^{(i)}}$ for $i = 1, 2, ..., N$.

Note that sampling ancestor indices $\{a_t^{(i)}\}_{i=1}^{N-1}$ in step ii) is equivalent to resampling $N - 1$ particles $\{\tilde{x}_{t-1}^{(i)}, \tilde{s}_{t-1}^{(i)}\}_{i=1}^{N-1}$ from $\{x_{t-1}^{(i)}, s_{t-1}^{(i)}\}_{i=1}^{N}$ with probability $\{\hat{\omega}_{t-1}^{(i)}\}_{i=1}^{N}$. This completes the benchmark PG sampler for NLG-SSSM. The summary of the benchmark PG algorithm is given by the following.

Algorithm 2-2: Benchmark PG for Non-linear/non-Gaussian SSSM

Choose θ arbitrarily and draw $\{X_{0:T}, S_{0:T}, A_{1:T}\}$ by running *Algorithm 1-1* **(SMC algorithm)**: $\{X_{0:T}, S_{0:T}, A_{1:T}\} \sim \Phi(X_{0:T}, S_{0:T}, A_{1:T}|\theta)$

- Iterate step i), step ii), and step iii) for $r = 1, 2, ..., R$.

i) Draw $K \in \{1, 2, ..., N\}$ (a reference trajectory) from: $K \sim \Phi(K|\theta, X_{0:T}, S_{0:T}, A_{1:T})$
And set $\{x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}\} = \{x_{0:T}^{(K)}, s_{0:T}^{(K)}\}$.

ii) Draw θ from: $\theta \sim \Phi(\theta|x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T})$

iii) Draw $\{X_{0:T}^{(-b_{0:T})}, S_{0:T}^{(-b_{0:T})}, A_{1:T}^{(-b_{1:T})}\}$ by running *Algorithm 2-1* **(CSMC algorithm)** from:
$\{X_{0:T}^{(-b_{0:T})}, S_{0:T}^{(-b_{0:T})}, A_{1:T}^{(-b_{1:T})}\} \sim \Phi(X_{0:T}^{(-b_{0:T})}, S_{0:T}^{(-b_{0:T})}, A_{1:T}^{(-b_{1:T})}|\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T})$
And set $X_{0:T} = \{X_{0:T}^{(-b_{0:T})}, x_{0:T}^{(b_{0:T})}\}$, $S_{0:T} = \{S_{0:T}^{(-b_{0:T})}, s_{0:T}^{(b_{0:T})}\}$, $A_{1:T} = \{A_{1:T}^{(-b_{1:T})}, b_{0:T-1}\}$.

The variable K represents the index of a reference particle trajectory; R is the total number of MCMC iterations.

Many papers such as those by Whiteley (2010), Fredrik and Schon (2012), and Whiteley et al. (2011) recognize that a standard PG sampler seriously suffers from poor mixing due to path degeneracy. The same problem arises in the benchmark PG sampler when it is applied to NLG-SSSM. To address the issue of path degeneracy and poor mixing, I introduce an alternative PG sampler in the next section.

3.2 Proposed Particle Gibbs

An SCM algorithm with additional backward simulation substantially alleviates path degeneracy by shuffling particle trajectories backward in time. A PG sampler with ancestor sampling (PGAS) developed by Fredrik and Schon (2012) and Lindsten et al. (2014) implicitly incorporates the backward simulation by updating particle trajectories forward in time without adding explicit backward recursion. In this section, I propose a PGAS sampler for NLG-SSSM that targets the extended target distribution in equation (14) to resolve path degeneracy and improve mixing of the resulting MCMC chain.

The main difference between the proposed PG sampler and the benchmark PG sampler is in the treatment of the index variables $b_{0:T-1} = \{b_0, b_1, ..., b_{T-1}\}$ of a reference trajectory. While the benchmark PG sampler keeps a particular reference trajectory fixed at each MCMC iteration, the proposed PG sampler constructs a new particle trajectory by drawing the ancestor indices $b_{t-1}(= a_t^{b_t})$ at each time. For instance, if $b_{t-1} = 5$ is drawn in a supplementary procedure, we accordingly set a new reference trajectory as $x_{1:t}^{(b_{0:t})} = \{x_{1:t-2}^{(b_{0:t-2})}, x_{t-1}^{(b_{t-1}=5)}, x_t^{(b_t)}\}$. This additional step to update the indices $b_{0:T-1}$ has a similar effect to that of backward recursion, which will be shown in *Proposition 2*.

The first and second steps of the proposed PGAS sampler are exactly the same as those of the benchmark PG sampler. The index of a new reference trajectory K is sampled among $\{X_{0:T}, S_{0:T}, A_{1:T}\}$, which contains a previously accepted reference trajectory. Based on *Proposition1*, K is drawn according to the importance weight $\bar{\omega}_T^{(i)}$ at T. As before, we assume that sampling θ is straightforward based on the conditional in equation (16).

Using partially collapsed Gibbs steps and the extended target density in equation (14), we have the following conditional to generate particles given a reference trajectory for $t = 0$:

$$\Phi(X_0^{(-b_0)}, S_0^{(-b_0)}|\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T}) = \prod_{\substack{i=1 \\ i \neq b_0}}^{N} q(x_0^{(i)}|s_0^{(i)})q(s_0^{(i)}) \tag{17}$$

and, for $t = 1, 2, ..., T$

$$\begin{aligned} \Phi(X_t^{(-b_t)}, &S_t^{(-b_t)}, A_t^{(-b_t)}|\theta, X_{0:t-1}, S_{0:t-1}, A_{1:t-1}, x_{t:T}^{(b_{t:T})}, s_{t:T}^{(b_{t:T})}, b_{t-1:T}) \\ &= \Phi(X_t^{(-b_t)}, S_t^{(-b_t)}, A_t^{(-b_t)}|\theta, X_{0:t-1}^{(-b_{0:t-1})}, S_{0:t-1}^{(-b_{0:t-1})}, A_{1:t-1}^{(-b_{1:t-1})}, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T}) \\ &= \frac{\Phi(X_{0:t}^{(-b_{0:t})}, S_{0:t}^{(-b_{0:t})}, A_{0:t}^{(-b_{0:t})}|\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T})}{\Phi(X_{0:t-1}^{(-b_{0:t-1})}, S_{0:t-1}^{(-b_{0:t-1})}, A_{0:t-1}^{(-b_{0:t-1})}|\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T})} \\ &= \prod_{\substack{i=1 \\ i \neq b_t}}^{N} \frac{\bar{\omega}_{t-1}^{(i)}}{\sum_j \bar{\omega}_{t-1}^{(j)}} q(x_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})}, s_t^{(i)}) q(s_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})}) = \prod_{\substack{i=1 \\ i \neq b_t}}^{N} M_t^{\theta}(a_t^{(i)}, x_t^{(i)}, s_t^{(i)}) \end{aligned} \tag{18}$$

Equations (17) and (18) show that we can draw $\{X_0^{(-b_0)}, S_0^{(-b_0)}\}$ using $q(x_0^{(i)}|s_0^{(i)})q(s_0^{(i)})$ and then draw $\{X_{0:t}^{(-b_{0:t})}, S_{0:t}^{(-b_{0:t})}, A_{1:t}^{(-b_{1:t})}\}$ from the combination of the resampling weight and the importance distributions, $\frac{\bar{\omega}_{t-1}^{(i)}}{\sum_j \bar{\omega}_{t-1}^{(j)}} q(x_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})}, s_t^{(i)})q(s_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})})$. Lastly, the transition kernel to produce a new ancestor index $b_{t-1}(= a_t^{b_t})$ is given in *Proposition 2*.

Proposition 2 *The conditional* $\Phi(b_{t-1}|\theta, X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t:T}^{(b_{t:T})}, s_{t:T}^{(b_{t:T})}, b_{t:T})$ *under the extended target* $\Phi(\theta, X_{0:T}, S_{0:T}, A_{1:T}, K)$ *is proportional to the following backward kernel at* $t-1$*:*

$$\begin{aligned}\Phi(b_{t-1}|\theta, &X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t:T}^{(b_{t:T})}, s_{t:T}^{(b_{t:T})}, b_{t:T}) \\ &\propto \left[\prod_{l=t}^{T} f_\theta(y_l|x_{0:l}^{(b_{0:l})}, s_{0:l}^{(b_{0:l})}) g_\theta(x_l|x_{0:l-1}^{(b_{0:l-1})}, s_{0:l}^{(b_{0:l})}) g_\theta(s_l^{(b_l)}|s_{l-1}^{(b_{l-1})})\right] \hat{\omega}_{t-1}^{(b_{t-1})}\end{aligned}$$

Thus, we draw b_{t-1} $(= a_t^{(b_t)}) \in \{1, 2, ..., N\}$ with the following probability:

$$\tilde{\omega}_{t-1|T}^{(i)} = \frac{\bar{\omega}_{t-1|T}^{(i)}}{\sum_{j=1}^{N} \bar{\omega}_{t-1|T}^{(j)}} \tag{19}$$

where

$$\bar{\omega}_{t-1|T}^{(i)} = \left[\prod_{l=t}^{T} f_\theta(y_l|x_{0:l}^{(b_{0:l})}, s_{0:l}^{(b_{0:l})}) g_\theta(x_l|x_{0:l-1}^{(b_{0:l-1})}, s_{0:l}^{(b_{0:l})}) g_\theta(s_l^{(b_l)}|s_{l-1}^{(b_{l-1})})\right] \hat{\omega}_{t-1}^{(b_{t-1})}.$$

Appendix provides the proof of *Proposition 2*. I note, in a special case, that the backward kernel in *Proposition 2* is equivalent to that of backward simulation in equation (9).

Lemma 1 *For a Markov state space model, the conditional* $\Phi(b_{t-1}|\theta, X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t:T}^{(b_{t:T})}, s_{t:T}^{(b_{t:T})}, b_{t:T})$ *is proportional to the backward kernel used in backward simulation:*

$$\begin{aligned}\Phi(b_{t-1}|\theta, &X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t:T}^{(b_{t:T})}, s_{t:T}^{(b_{t:T})}, b_{t:T}) \\ &\propto f_\theta(y_t|x_t^{(b_t)}, s_t^{(b_t)}) g_\theta(x_t|x_{t-1}^{(b_{t-1})}, s_t^{(b_t)}) g_\theta(s_t^{(b_t)}|s_{t-1}^{(b_{t-1})}) \hat{\omega}_{t-1}^{(b_{t-1})}\end{aligned}$$

The proof of *Lemma 1* is straightforward using the special dependence structure of a Markov state space model; I therefore skip the proof for brevity. Based on the derived MCMC kernels, a modified conditional SMC with ancestor sampling is introduced, which is crucial for implementing the proposed PG sampler with ancestor sampling.

Algorithm 3-1: CSMC with Ancestor Sampling (CSMC-AS)

i) Draw $\{s_0^{(i)}\}_{i=1}^{N-1}$ from $q(s_0)$ and draw $\{x_0^{(i)}\}_{i=1}^{N-1}$ from $q(x_0|s_0^{(i)})$ sequentially. Set $\{x_0^{(N)}, s_0^{(N)}\} = \{x_0^{(b_0)}, s_0^{(b_0)}\}$. Save the normalized importance weights $\{\hat{\omega}_0^{(i)} = \frac{\bar{\omega}_0^{(i)}}{\sum_{j=1}^{N} \bar{\omega}_0^{(i)}}\}_{i=1}^{N}$ where $\bar{\omega}_0^{(i)} = \frac{g_\theta(x_0^{(i)}|s_0^{(i)}) g_\theta(s_0^{(i)})}{q(x_0^{(i)}|s_0^{(i)}) q(s_0^{(i)})}$.

- Iterate step ii), iii), and vi) for $t = 1, 2, ..., T$.

ii) Draw ancestor indices $\{a_t^{(i)}\}_{i=1}^{N-1}$ according to probability $\{\hat{\omega}_{t-1}^{(i)}\}_{i=1}^{N}$. Draw $\{s_t^{(i)}\}_{i=1}^{N-1}$ from $q(s_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})})$ and $\{x_t^{(i)}\}_{i=1}^{N-1}$ from $q(x_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})}, s_t^{(i)})$ sequentially.

iii) Draw $b_{t-1}(= a_t^{(b_t)})$ with probability $\tilde{\omega}_{t-1|T}^{(i)}$ in equation (19). Set $a_t^{(N)} = b_{t-1}$ and $\{x_t^{(N)}, s_t^{(N)}\} = \{x_t^{(b_t)}, s_t^{(b_t)}\}$. The trajectories are set by $x_{0:t}^{(i)} = \{x_{0:t-1}^{(a_t^{(i)})}, x_t^{(i)}\}$ and $s_{0:t}^{(i)} = \{s_{0:t-1}^{(a_t^{(i)})}, s_t^{(i)}\}$ for $i = 1, 2, ..., N$.

vi) Calculate the unnormalized weights $\bar{\omega}_t^{(i)} = \frac{f_\theta(y_t|x_{0:t}^{(i)}, s_{0:t}^{(i)})g_\theta(x_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t}^{(i)})g_\theta(s_t^{(i)}|s_{t-1}^{(a_t^{(i)})})}{q(x_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t}^{(i)})q(s_t^{(i)}|x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})})}$.

and obtain the normalized weights: $\hat{\omega}_t^{(i)} = \frac{\bar{\omega}_t^{(i)}}{\sum_{j=1}^{N} \bar{\omega}_t^{(i)}}$ for $i = 1, 2, ..., N$.

Note that sampling ancestor indices $\{a_t^{(i)}\}_{i=1}^{N-1}$ in step ii) is equivalent to resampling $N - 1$ particles $\{\tilde{x}_{t-1}^{(i)}, \tilde{s}_{t-1}^{(i)}\}_{i=1}^{N-1}$ from $\{x_{t-1}^{(i)}, s_{t-1}^{(i)}\}_{i=1}^{N}$ with probability $\{\hat{\omega}_{t-1}^{(i)}\}_{i=1}^{N}$, as before. The summary of the proposed PG with ancestor sampling to estimate NLG-SSSM is given below.

Algorithm 3-2: PGAS for a Non-linear/non-Gaussian SSSM

Choose θ arbitrarily and draw $\{X_{0:T}, S_{0:T}, A_{1:T}\}$ by running *Algorithm 1-1* **(SMC algorithm)** from: $\{X_{0:T}, S_{0:T}, A_{1:T}\} \sim \Phi(X_{0:T}, S_{0:T}, A_{1:T}|\theta)$

- Iterate step i), step ii), and step iii) for $r = 1, 2, ..., R$.

i) Draw $K \in \{1, 2, ..., N\}$ (a reference trajectory) from: $K \sim \Phi(K|\theta, X_{0:T}, S_{0:T}, A_{1:T})$
And set $\{x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}\} = \{x_{0:T}^{(K)}, s_{0:T}^{(K)}\}$.

ii) Draw θ from: $\theta \sim \Phi(\theta|x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T})$

iii) Draw $\{X_0^{(-b_0)}, S_0^{(-b_0)}\}$ and $\{b_{t-1}(= a_t^{(b_t)}), X_t^{(-b_t)}, S_t^{(-b_t)}, A_t^{(-b_t)}\}$ for t = 1,2,...,T by running *Algorithm 3-1* **(CSMC-AS algorithm)**.

where $X_{0:t} = \{x_{0:t}^{(b_{0:t})}, X_{0:t}^{(-b_{0:t})}\}$; $S_{0:t} = \{s_{0:t}^{(b_{0:t})}, S_{0:t}^{(-b_{0:t})}\}$; $A_{1:t} = \{A_{1:t}^{(-b_{1:t})}, b_{0:t-1}\}$; K represents the index of a reference particle trajectory; R is the total number of MCMC iterations. The proposed PGAS sampler allows a reference trajectory to change its ancestry as the conditional SMC with ancestor sampling is operated in the forward direction. This approach is more robust to path degeneracy and enables a faster-mixing MCMC kernel than the benchmark PG sampler.

Though the ancestor sampling technique helps avoid path degeneracy by shuffling the ancestor indices and reducing the correlation between the particle trajectories, it cannot be applied to an NLG-SSSM with a degenerate transition equation. In such a model, the only previous particle trajectory that contains a particular future particle is that from which the future particle was originally generated. Hence, the probability of updating previous particle trajectories in ancestor sampling is always zero at every time period unless new values of the latent states are jointly generated along with the corresponding ancestor indices.

3.3 Proposed Particle Gibbs for a Degenerate Transition Equation

Building upon the idea of Lindsten et al. (2014), Carter et al. (2014) [14], and Bunch et al. (2015), I incorporate particle rejuvenation to estimate degenerate NLG-SSSM. The main idea of the new algorithm is to generates new values of latent states $\{x_t, s_t\}$ along with ancestor indices $b_{t-1}(= a_t^{(b_t)})$ of a reference trajectory at each MCMC iteration. In the proposed scheme, new values of the latent states are not restricted to the output of a SMC procedure and are also drawn with more relevant information than the conditional SMC method. More specifically, additional importance distributions are designed to approximate a properly chosen backward kernel to simultaneously sample an ancestor index and new values for the associated latent states $\{x_t, s_t\}$. Unlike the PGAS sampler in *Algorithm 5-2*, information on the future states $\{x_{t+1:T}, s_{t+1:T}\}$ is incorporated in drawing $\{x_t, s_t\}$.

All of the MCMC steps are the same as those of *Algorithm 5-2*, except fir step iii). The following proposition provides a justification for using the backward kernel to jointly generate new values of the latent states and the ancestor index.

Proposition 3 *The conditional* $\Phi(x_t^{(b_t)}, s_t^{(b_t)}, b_{t-1}|\theta, X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t:T}^{(b_{t:T})}, s_{t:T}^{(b_{t:T})}, b_{t:T})$ *under the extended target* $\Phi(\theta, X_{0:T}, S_{0:T}, A_{1:T}, K)$ *is proportional to the backward kernel at* $t-1$*:*

$$\begin{aligned}\Phi(x_t^{(b_t)}, s_t^{(b_t)}, b_{t-1}|\theta, &X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t+1:T}^{(b_{t+1:T})}, s_{t+1:T}^{(b_{t+1:T})}, b_{t:T})\\ &\propto \left[\prod_{l=t}^{T} f_\theta(y_l|x_{0:l}^{(b_{0:l})}, s_{0:l}^{(b_{0:l})}) g_\theta(x_l|x_{0:l-1}^{(b_{0:l-1})}, s_{0:l}^{(b_{0:l})}) g_\theta(s_l^{(b_l)}|s_{l-1}^{(b_{l-1})})\right]\hat{\omega}_{t-1}^{(b_{t-1})}\end{aligned}$$

The proof of *Proposition 3* is given in the Appendix, which is similar to the proof of *Proposition 2*. Analogously, it can be easily verified that the latent states of the initial period are sampled based on the following MCMC kernel.

$$\begin{aligned}\Phi(x_0^{(b_0)}, s_0^{(b_0)}|\theta, &x_{1:T}^{(b_{1:T})}, s_{1:T}^{(b_{1:T})}, b_{0:T})\\ &\propto \left[\prod_{l=1}^{T} f_\theta(y_l|x_{0:l}^{(b_{0:l})}, s_{0:l}^{(b_{0:l})}) g_\theta(x_l|x_{0:l-1}^{(b_{0:l-1})}, s_{0:l}^{(b_{0:l})}) g_\theta(s_l^{(b_l)}|s_{l-1}^{(b_{l-1})})\right] g_\theta(x_0^{(b_0)}|s_0^{(b_0)}) g_\theta(s_0^{(b_0)})\end{aligned}$$

We can also see that the backward kernel in *Proposition 3* is simplified for a Markov state space model.

[14] Carter et al. (2014) developed PMCMC samplers with a strategy similar to the particle rejuvenation technique. However, they specified a different extended target distribution to implement their methods and thus modified conditional SMC methods. Following Bunch et al. (2015), I do not change the original extended target distribution.

Lemma 2 *For a Markov state space model, the conditional* $\Phi(x_t^{(b_t)}, s_t^{(b_t)}, b_{t-1}|\theta, X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t:T}^{(b_{t:T})}, s_{t:T}^{(b_{t:T})}, b_{t:T})$ *under the extended target* $\Phi(\theta, X_{0:T}, S_{0:T}, A_{1:T}, K)$ *is proportional to the backward kernel used in backward simulation:*

$$\begin{aligned}\Phi(x_t^{(b_t)}, &s_t^{(b_t)}, b_{t-1}|\theta, X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t:T}^{(b_{t:T})}, s_{t:T}^{(b_{t:T})}, b_{t:T}) \\ &\propto g_\theta(x_{t+1}^{(b_{t+1})}|x_t^{(b_t)}, s_{t+1}^{(b_{t+1})}) g_\theta(s_{t+1}^{(b_{t+1})}|s_t^{(b_t)}) f_\theta(y_t|x_t^{(b_t)}, s_t^{(b_t)}) g_\theta(x_t|x_{t-1}^{(b_{t-1})}, s_t^{(b_t)}) g_\theta(s_t^{(b_t)}|s_{t-1}^{(b_{t-1})}) \hat{\omega}_{t-1}^{(b_{t-1})}\end{aligned}$$

I skip the derivation of *Lemma 2* for brevity. The MCMC procedure associated with the above backward kernel involves partially collapsed Gibbs steps and leaves $\Phi(.)$ invariant, as before. While the backward kernel in *Proposition 2* resembles that of *Proposition 3*, proper methods to implement those steps are quite different. The normalizing constant of the backward kernel in *Proposition 2* is easily calculated based on equation (19) in that we sample only the ancestor index b_{t-1} of a reference trajectory. On the other hand, the normalizing constant of the backward kernel in *Proposition 3* to jointly draw $\{x_t, s_t, b_{t-1}\}$ is not analytically tractable in general.

We can handle this problem by constructing MCMC kernels, leaving the conditional distributions in *Proposition 3* (for $t = 0$ and for $t = 1, 2, ..., T$) invariant. For instance, a conditional importance sampling (CIS) technique can be applied to approximate the conditional distributions at each time period. Consider the following importance distribution:

$$W(x_t, s_t, b_{t-1}) = \Psi(x_t, s_t|x_t^{*(b_t)}, s_t^{*(b_t)}, x_{0:t-1}^{(b_{t-1})}, s_{0:t-1}^{(b_{t-1})}, x_{t+1:T}^{(b_{t+1:T})}, s_{t+1:T}^{(b_{t+1:T})}) \frac{\nu_{t-1}^{(b_{t-1})}}{\sum_j^N \nu_{t-1}^{(j)}} \quad \textit{for } t = 1, 2, ..., T \tag{20}$$

where $\{x_t^{*(b_t)}, s_t^{*(b_t)}\}$ are the previously accepted MCMC draws and $\{x_t, s_t, b_{t-1}\}$ are newly generated draws. Similarly, the importance distribution for $t = 0$ and $t = T$ is properly designed by modifying equation (20). The above importance distribution indicates that a new ancestor index b_{t-1} of a reference trajectory is first sampled from a proposal weight $v_{t-1}^{(b_{t-1})}$, and new values of the latent states $\{x_t, s_t\}$ are then generated conditional on the previously accepted $\{x_t^{*(b_t)}, s_t^{*(b_t)}\}$, the new past reference trajectory $\{x_{0:t-1}^{(b_{t-1})}, s_{0:t-1}^{(b_{t-1})}\}$ up to time $t-1$ and all the future reference trajectory $\{x_{t+1:T}^{(b_{t+1:T})}, s_{t+1:T}^{(b_{t+1:T})}\}$. The summary is given below for the CIS algorithm in which C denotes the total number of particles used.

Algorithm 4-1: Conditional Importance Sampling (CIS) at time t

```
i) Set $\{x_t^{(C)}, s_t^{(C)}\} = \{x_t^{(b_t)}, s_t^{(b_t)}\}$.
• Iterate step ii) and step iii) for $c = 1, 2, ..., C-1$.
ii) Draw ancestor index $b_{t-1}^{(c)} \in \{1, 2, ..., N\}$ with probability $\frac{\nu_{t-1}^{(b_{t-1})}}{\sum_j^N \nu_{t-1}^{(j)}}$.
    Set $x_{0:t-1}^{(b_{t-1}^{(c)})} = \{x_{0:t-2}^{(b_{0:t-2})}, x_{t-1}^{(b_{t-1}^{(c)})}\}$ , $s_{0:t-1}^{(b_{t-1}^{(c)})} = \{s_{0:t-2}^{(b_{0:t-2})}, s_{t-1}^{(b_{t-1}^{(c)})}\}$.
```

iii) Draw $\{x_t^{(c)}, x_t^{(c)}\}$ from $\Psi(x_t, s_t | x_t^{(b_t)}, s_t^{(b_t)}, x_{0:t-1}^{(b_{t-1}^{(c)})}, s_{0:t-1}^{(b_{t-1}^{(c)})}, x_{t+1:T}^{(b_{t+1:T})}, s_{t+1:T}^{(b_{t+1:T})})$.
Set $x_{0:T}^{(c)} = \{x_{0:t-1}^{(b_{t-1}^{(c)})}, x_t^{(c)}, x_{t+1:T}^{(b_{t+1:T})}\}$ and $s_{0:T}^{(c)} = \{s_{0:t-1}^{(b_{t-1}^{(c)})}, s_t^{(c)}, s_{t+1:T}^{(b_{t+1:T})}\}$.

vi) Calculate the unnormalized weights for $c = 1, 2, ..., C$:

$$\bar{\tau}_t^{(c)} = \frac{\left[\prod_{l=t}^{T} f_\theta(y_l | x_{0:l}^{(c)}, s_{0:l}^{(c)}) p_\theta(x_l | x_{0:l-1}^{(c)}, s_{0:l}^{(c)}) p_\theta(s_l^{(c)} | s_{l-1}^{(c)})\right] \hat{\omega}_{t-1}^{(b_{t-1}^{(c)})}}{\Psi(x_t^{(c)}, s_t^{(c)} | x_t^{(b_t)}, s_t^{(b_t)}, x_{0:t-1}^{(b_{t-1}^{(c)})}, s_{0:t-1}^{(b_{t-1}^{(c)})}, x_{t+1:T}^{(b_{t+1:T})}, s_{t+1:T}^{(b_{t+1:T})}) \nu_{t-1}^{(b_{t-1}^{(c)})}}$$

Obtain the normalized weights: $\hat{\tau}_t^{(c)} = \frac{\bar{\tau}_t^{(c)}}{\sum_{j=1}^{C} \bar{\tau}_t^{(j)}}$.

v) Draw: $\{x_t, s_t, b_{t-1}\}$ from $\{x_t^{(c)}, s_t^{(c)}, b_{t-1}^{(c)}\}_{c=1}^{C}$ with probability $\hat{\tau}_t^{(c)}$.

We can perform conditional importance sampling with large C to enhance the mixing property in exchange for increasing computational costs. Using the derivation by Andrieu et al. (2010), it can be shown that the CIS admits the target Markov kernel in *Proposition 3* as a stationary distribution. Indeed, any valid MCMC kernel such as the Metropolis-Hastings kernel can be used to approximate the target kernel. As a proper importance distribution in the CIS is dependent on the structure of a state space model of interest, we will discuss this issue based on a particular SSSM in the next section.

Now, the details of step iii) of the proposed PG sampler with particle rejuvenation are introduced in the following algorithm with the CIS scheme.

Algorithm 4-2: CSMC with Particle Rejuvenation (CSMC-PR)

i) Draw $\{s_0^{(i)}\}_{i=1}^{N-1}$ from the $q(s_0)$ and draw $\{x_0^{(i)}\}_{i=1}^{N-1}$ from $q(x_0 | s_0^{(i)})$ sequentially.

ii) Draw $\{x_0^{(b_0)}, s_0^{(b_0)}\}$ from the Markov kernel $W(.)$ in equation (20) and set $\{x_0^{(N)}, s_0^{(N)}\} = \{x_0^{(b_0)}, s_0^{(b_0)}\}$.
Save the normalized importance weights $\{\hat{\omega}_0^{(i)} = \frac{\bar{\omega}_0^{(i)}}{\sum_{j=1}^{N} \bar{\omega}_0^{(i)}}\}_{i=1}^{N}$ where $\bar{\omega}_0^{(i)} = \frac{g_\theta(x_0^{(i)} | s_0^{(i)}) g_\theta(s_0^{(i)})}{q(x_0^{(i)} | s_0^{(i)}) q(s_0^{(i)})}$.

- Iterate step iii), vi), and v) for $t = 1, 2, ..., T$.

iii) Draw ancestor indices $\{a_t^{(i)}\}_{i=1}^{N-1}$ with probability $\{\hat{\omega}_{t-1}^{(i)}\}_{i=1}^{N}$. Draw $\{s_t^{(i)}\}_{i=1}^{N-1}$ from $q(s_t^{(i)} | x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})})$ and $\{x_t^{(i)}\}_{i=1}^{N-1}$ from $q(x_t^{(i)} | x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})}, s_t^{(i)})$ sequentially.

iii) Run *Algorithm 4-1* **(CIS algorithm)** to draw $\{x_t^{(b_t)}, s_t^{(b_t)}, b_{t-1}\}$. Set $a_t^{(N)} = b_{t-1}$ and $\{x_t^{(N)}, s_t^{(N)}\} = \{x_t^{(b_t)}, s_t^{(b_t)}\}$. New particle trajectories are set by $x_{0:t}^{(i)} = \{x_{0:t-1}^{(a_t^{(i)})}, x_t^{(i)}\}$ and $s_{0:t}^{(i)} = \{s_{0:t-1}^{(a_t^{(i)})}, s_t^{(i)}\}$ for $i = 1, 2, ..., N$.

vi) Calculate the unnormalized weights: $\bar{\omega}_t^{(i)} = \frac{f_\theta(y_t | x_{0:t}^{(i)}, s_{0:t}^{(i)}, y_{1:t-1}) g_\theta(x_t^{(i)} | x_{0:t-1}^{(a_t^{(i)})}, s_{0:t}^{(i)}) g_\theta(s_t^{(i)} | s_{t-1}^{(a_t^{(i)})})}{q(x_t^{(i)} | x_{0:t-1}^{(a_t^{(i)})}, s_{0:t}^{(i)}) q(s_t^{(i)} | x_{0:t-1}^{(a_t^{(i)})}, s_{0:t-1}^{(a_t^{(i)})})}$.
Obtain the normalized weights: $\hat{\omega}_t^{(i)} = \frac{\bar{\omega}_t^{(i)}}{\sum_{j=1}^{N} \bar{\omega}_t^{(i)}}$ for $i = 1, 2, ..., N$.

The proposed algorithm is an extension of the PG algorithm by Lindsten et al. (2014) to estimate NLG-SSSM. The uniform ergodicity for the suggested algorithm can be demonstrated using the work of Lindsten et al. (2014). The summary of the proposed PG sampler for degenerate NLG-SSSM is provided below.

Algorithm 4-3: PGAS with Particle Rejuvenation for a Non-linear/non-Gaussian SSSM

```
    Choose $\theta$ arbitrarily and draw $\{X_{0:T}, S_{0:T}, A_{1:T}\}$ by running Algorithm 1-1 (SMC algorithm) from:
    $\{X_{0:T}, S_{0:T}, A_{1:T}\} \sim \Phi(X_{0:T}, S_{0:T}, A_{1:T}|\theta)$
•   Iterate step i), step ii), and step iii) for $r = 1, 2, ..., R$.
i)   Draw $K \in \{1, 2, ..., N\}$ (a reference trajectory) from: $K \sim \Phi(K|\theta, X_{0:T}, S_{0:T}, A_{1:T})$
     And set $\{x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}\} = \{x_{0:T}^{(K)}, s_{0:T}^{(K)}\}$.
ii)  Draw $\theta$ from: $\theta \sim \Phi(\theta|x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}, b_{0:T})$
iii) Draw $\{X_0^{(-b_0)}, S_0^{(-b_0)}\}$ and $\{b_{t-1}(= a_t^{(b_t)}), x_t^{(b_t)}, s_t^{(b_t)}\}$, $\{X_t^{(-b_t)}, S_t^{(-b_t)}, A_t^{(-b_t)}\}$ for t = 1,2,...,T by running Algorithm 4-2 (CSMC-PR algorithm).
```

where $X_{0:t} = \{x_{0:t}^{(b_{0:t})}, x_{0:t}^{(-b_{0:t})}\}$; $S_{0:t} = \{s_{0:t}^{(b_{0:t})}, s_{0:t}^{(-b_{0:t})}\}$; $A_{1:t} = \{A_{1:t}^{(-b_{1:t})}, b_{0:t-1}\}$; K represents the index of a reference particle trajectory; and R is the total number of MCMC iterations.

3.4 Implementation of Proposed Particle Gibbs

In practice, the extended target density in (14) and associated backwards kernels can be simplified according to the structure of a particular NLG-SSSM of interest. This section provides more details on how the suggested PG algorithms are implemented using two specific examples. First, consider the regime switching stochastic volatility (SV) model by So et al. (1998): **Example 1**

$$y_t = \mu + exp(\frac{x_{t-1}}{2})\epsilon_t,\ \epsilon_t \sim N(0,1) \tag{21}$$

$$x_t = \delta_{s_t} + \phi(x_{t-1} - \delta_{s_{t-1}}) + u_t,\ u_t \sim N(0, \sigma_u^2)$$

where y_t is the equity return at time t; x_{t-1} is the latent log-volatility at time t; and $E[\epsilon_t u_t] = 0$. The current position of x_t is given by a function of x_{t-1}, s_t, and s_{t-1} in the transition equation, and the observation y_t is given by a nonlinear function of x_{t-1} in the measurement equation. Because x_t and y_t depend only on a few past states in such a model structure, the backward kernel in *Proposition 2* reduces to:

$$\begin{aligned}
&\Phi(b_{t-1}|\theta, X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t:T}^{(b_{t:T})}, s_{t:T}^{(b_{t:T})}, b_{t:T}) \\
&\propto \left[f_\theta(y_t|x_{t-1}^{(b_{t-1})}) g_\theta(x_t^{(b_t)}|x_{t-1}^{(b_{t-1})}, s_t^{(b_t)}, s_{t-1}^{(b_{t-1})}) g_\theta(s_t^{(b_t)}|s_{t-1}^{(b_{t-1})}) \right] \hat{\omega}_{t-1}^{(b_{t-1})} \qquad (22) \\
&\propto \omega_{t-1|T}^{(b_{t-1})}.
\end{aligned}$$

Therefore, $b_{t-1} \in \{1, 2, ..., N\}$ is drawn with the normalized importance weight in (19).

Similarly, new values of the latent states and their ancestor indices are jointly simulated by step iii) of *Algorithm 4-3* using the following MCMC kernel:

$$
\begin{aligned}
&\Phi(x_t^{(b_t)}, s_t^{(b_t)}, b_{t-1}|\theta, X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t+1:T}^{(b_{t+1:T})}, s_{t+1:T}^{(b_{t+1:T})}, b_{t:T}) \\
&\propto \Big[f_\theta(y_{t+1}|x_t^{(b_t)}) g_\theta(x_{t+1}^{(b_{t+1})}|x_t^{(b_t)}, s_{t+1}^{(b_{t+1})}, s_t^{(b_t)}) g_\theta(s_{t+1}^{(b_{t+1})}|s_t^{(b_t)}) \\
&\times f_\theta(y_t|x_{t-1}^{(b_{t-1})}) g_\theta(x_t^{(b_t)}|x_{t-1}^{(b_{t-1})}, s_t^{(b_t)}, s_{t-1}^{(b_{t-1})}) g_\theta(s_t^{(b_t)}|s_{t-1}^{(b_{t-1})}) \Big] \hat{\omega}_{t-1}^{(b_{t-1})}
\end{aligned}
\tag{23}
$$

Before implementing the CIS procedure in *Algorithm 4-1*, we should construct importance distributions that target the kernel in (23). First, the importance weight $\bar{\tau}_t^{(i)}$ is simplified by using $\nu_{t-1}^{(i)} = \hat{\omega}_{t-1}^{(i)}$ and the easily calculable likelihood function $f_\theta(y_t|x_{t-1}^{(b_{t-1})})$ in drawing b_{t-1}. Thus, the ancestor index $b_{t-1}^{(c)} \in \{1, 2, ..., N\}$ of a reference trajectory is drawn with the updated $\hat{\omega}_{t-1}^{(i)}$ as:

$$
b_{t-1}^{(c)} \sim \frac{f_\theta(y_t|x_{t-1}^{(b_{t-1})})\hat{\omega}_{t-1}^{(b_{t-1})}}{\sum_{j=1}^N f_\theta(y_t|x_{t-1}^{(j)})\hat{\omega}_{t-1}^{(j)}}.
$$

New values of the latent states are efficiently generated using their transition densities:

$$
s_t^{(c)} \sim p_\theta(s_t|s_{t+1}^{(b_{t+1})}, s_{t-1}^{(b_{t-1}^{(c)})}) = \frac{p_\theta(s_{t+1}^{(b_{t+1})}|s_t) p_\theta(s_t|s_{t-1}^{(b_{t-1}^{(c)})})}{\sum_{s_t} p_\theta(s_{t+1}^{(b_{t+1})}|s_t) p_\theta(s_t|s_{t-1}^{(b_{t-1}^{(c)})})}
$$

and

$$
x_t^{(c)} \sim p_\theta(x_t|x_{t+1}^{(b_{t+1})}, x_{t-1}^{(b_{t-1}^{(c)})} s_{t+1}^{(b_{t+1})}, s_t^{(c)}, s_{t-1}^{(b_{t-1}^{(c)})}) \propto p_\theta(x_{t+1}^{(b_{t+1})}|x_t, s_{t+1}^{(b_{t+1})}, s_t^{(c)}) p_\theta(x_t|x_{t-1}^{(b_{t-1}^{(c)})}, s_t^{(c)}, s_{t-1}^{(b_{t-1}^{(c)})})
$$

It is shown in Appendix that $x_t^{(c)}$ follows $N(\mu_t, V_t)$ where $V_t = \frac{\sigma_u^2}{(1+\phi^2)}$; $\mu_t = \delta_{s_t} + \frac{\phi}{(1+\phi^2)}((x_{t+1} - \delta_{s_{t+1}}) + (x_{t-1} - \delta_{s_{t-1}}))$. The unnormalized importance weight $\hat{\tau}^{(c)}$ in the CIS algorithm therefore is given by:

$$
\hat{\tau}^{(c)} = f_\theta(y_{t+1}|x_t^{(c)})[\sum_{s_t} p_\theta(s_{t+1}^{(b_{t+1})}|s_t) p_\theta(s_t|s_{t-1}^{(b_{t-1}^{(c)})})] p_\theta(x_{t+1}^{(b_{t+1})}|x_{t-1}^{(b_{t-1}^{(c)})} s_{t+1}^{(b_{t+1})}, s_t^{(c)}, s_{t-1}^{(b_{t-1}^{(c)})}).
$$

After sampling necessary random samples, the most likely particle set and the corresponding ancestor index are accepted with the importance weight $\hat{\tau}^{(c)}$.

Another example of a NLG-SSSM is given as follows:

Example 2

$$
\begin{aligned}
y_{1,t} &= \beta_0 + \beta_1 f_t + \beta_2 f_t^2 + \epsilon_{1,t} \ \epsilon_t \sim N(0, \sigma_{\epsilon_1}^2) \\
y_{2,t} &= \alpha_0 + f_t + \epsilon_{2,t} \ \epsilon_{2,t} \sim N(0, \sigma_{\epsilon_2}^2)
\end{aligned}
\tag{24}
$$

$$f_t = \phi f_t + u_t - \gamma u_{t-1},\ u_t \sim N(0, \sigma^2_{s_t})$$

where $y_{1,t}$ and $y_{2,t}$ are the two observable variables; f_t is the latent continuous state variable; and s_t presents the latent regime-indicator variable. The presented model can be considered a simple non-linear dynamic factor model[15] similar to ones considered by Bai and Ng (2008) and Ludvigson and Ng (2007). They discuss the use of squared factors in improving forecasting equations. Another important feature of the model in (24) is that the common factor f_t is allowed to have an infinite moving average (MA) representation with both autoregressive (AR) and MA parts, as shown by Forni et al. (2000). The proposed model also can be considered as a simple reduced-form term structure model in which an observable variable $y_{2,t}$ contains a macro factor f_t for a bond yield $y_{1,t}$; see Song (2014) for more details.

Bayesian inference in *Example 2* is quite complex mainly due to the ARMA structure of f_t. The recursive nature of the ARMA process makes f_t dependent upon the whole sequence of the latent regime indicator variable up to time t, which is referred to as the path dependence problem. The problem is partly solved by casting the ARMA process into a state space representation to recover the Markovian nature of f_t and s_t:

$$\begin{bmatrix} f_t \\ u_t \end{bmatrix} = \begin{bmatrix} \phi & -\gamma \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_{t-1} \\ u_{t-1} \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u_t,\ u_t \sim N(0, \sigma^2_{s_t}).$$

$$(x_t = F x_{t-1} + G u_t,\ u_t \sim N(0, \sigma^2_{s_t}))$$

However, note that the transition equation becomes degenerate as $rank(G) < dim(x_t)$ in the matrix form. As a result, the probability of updating the ancestor of a particular future state is exactly zero when the proposed PG sampler with ancestor sampling is applied. In fact, this type of degenerate model is pretty common in many applications, as many macro variables possess ARMA dynamics.

The aforementioned problem can be easily circumvented by employing the PGAS sampler with particle rejuvenation in *Algorithm 4-3*. Given the model structure in (24), we have the MCMC kernel for step iii) of *Algorithm 4-3*:

$$\begin{aligned} &\Phi(x_t^{(b_t)}, s_t^{(b_t)}, b_{t-1} | \theta, X_{0:t-1}, S_{0:t-1}, A_{0:t-1}, x_{t+1:T}^{(b_{t+1:T})}, s_{t+1:T}^{(b_{t+1:T})}, b_{t:T}) \\ &\propto \left[g_\theta(x_{t+1}^{(b_{t+1})} | x_t^{(b_t)}, s_t^{(b_t)}) g_\theta(s_{t+1}^{(b_{t+1})} | s_t^{(b_t)}) f_\theta(y_t | x_t^{(b_t)}) g_\theta(x_t^{(b_t)} | x_{t-1}^{(b_{t-1})}, s_t^{(b_t)}) g_\theta(s_t^{(b_t)} | s_{t-1}^{(b_{t-1})}) \right] \hat{\omega}_{t-1}^{(b_{t-1})} \end{aligned} \tag{25}$$

The importance distributions used in the CIS procedure are constructed to approximate the kernel in (25). The importance weight $\bar{\tau}_t^{(i)}$ is set by $\nu_{t-1}^{(i)} = \hat{\omega}_{t-1}^{(i)}$ in drawing b_{t-1}:

$$b_{t-1}^{(c)} \sim \frac{\hat{\omega}_{t-1}^{(b_{t-1})}}{\sum_{j=1}^N \hat{\omega}_{t-1}^{(j)}}.$$

[15] Admittedly, the proposed PG sampler is not suitable for the case in which a substantially large number of observable variables share common factors. I leave this analysis for future work.

By making use of the transition densities, new values of the latent states are generated as follows:

$$s_t^{(c)} \sim p_\theta(s_t|s_{t+1}^{(b_{t+1})}, s_{t-1}^{(b_{t-1}^{(c)})}) = \frac{p_\theta(s_{t+1}^{(b_{t+1})}|s_t)p_\theta(s_t|s_{t-1}^{(b_{t-1}^{(c)})})}{\sum_{s_t} p_\theta(s_{t+1}^{(b_{t+1})}|s_t)p_\theta(s_t|s_{t-1}^{(b_{t-1}^{(c)})})}$$

and

$$x_t^{(c)} \sim p_\theta(x_t|x_{t-1}^{(b_{t-1}^{(c)})}, s_t^{(c)})$$

For the degenerate transition equation, the future state x_{t+1} cannot be incorporated in the importance distribution of x_t as the inverse of the covariance matrix $\Sigma_{s_t} = \sigma_{s_t}^2 GG'$ does not exist. Thus, the importance weight is defined by:

$$\hat{\tau}^{(c)} = f_\theta(y_t|x_t^{(c)})[\sum_{s_t} p_\theta(s_{t+1}^{(b_{t+1})}|s_t)p_\theta(s_t|s_{t-1}^{(b_{t-1}^{(c)})})]p_\theta(x_{t+1}^{(b_{t+1})}|x_t^{(c)}, s_{t+1}^{(b_{t+1})})$$

The proposed PG samplers in this section are applied to two particular examples for illustration.

3.5 Performance of Proposed Algorithm: Simulation Study

The main goal of this section is to compare the performance of the proposed PG algorithms with that of the benckmark PG algorithm in estimating NLG-SSSM. For this purpose, I simulate the models in (21) and (24) for $T = 3000$ and $T = 500$, respectively. The first model is generated with $\{\mu = 0, \delta_1 = -1, \delta_2 = 0.5, \phi = 0.9, \sigma_u^2 = 0.01, \pi_{11} = 0.99, \pi_{22} = 0.99\}$, and the second model is generated with $\{\beta_0 = 0, \beta_1 = 1, \beta_2 = -0.5, \alpha_0 = 0, \phi = 0.9, \gamma = 0.3, \sigma_{\epsilon_1}^2 = 0.01, \sigma_{\epsilon_2}^2 = 0.01, \sigma_1^2 = 0.01, \sigma_1^2 = 0.05, \pi_{11} = 0.99, \pi_{22} = 0.99\}$, where p_{jj} is the transition probability for $j = 1, 2$. I run all the three PG samplers (*Algorithm 2-2, Algorithm 3-2, Algorithm 4-3*) for *Example 1* and the two PG samplers (*Algorithm 2-2, Algorithm 4-3*) for *Example 2*. Note that *Algorithm 2-2* and *Algorithm 3-2* become equivalent in *Example 2* due to the degenerate transition equation. The numbers of particles used in the benchmark PG and the PGAS samplers are $N = 1,000$, and $N = 20$, respectively. The numbers of particles for the PGAS with particle rejuvenation are set as $N = 20, M = 20$. I keep the latter 40,000 iterations and discard the initial 1,000 iterations as warm up for all simulations. All Bayesian estimates reported in this section are the averages of 5 simulations.

The inefficiency factor described by Geweke (1992) is one of the popular measures of MCMC efficiency. The inefficiency factor is defined as:

$$\kappa_J = 1 + 2\sum_{j=1}^{J} \rho_j, \quad J = 2,000,$$

where ρ_j is the correlation for lag j. As posterior draws become less and less serially correlated, the inefficient factor approaches the ideal minimum value of 1. On the other hand, in the case of an inefficient MCMC sampler that produces highly correlated MCMC draws, the ratio becomes larger than 1. The inefficiency factor serves to quantify the relative efficiency loss in computing posterior moments by comparing correlated and independent samples. Tables 1.A and 1.B show the inefficiency factors of the model parameters in (21) and (24).

The PG sampler with ancestor sampling in *Algorithm 3-2* exhibits the best performance for *Example 1.* The inefficiency factors are reported in Table 1.A. The inefficiency factors obtained based on the PGAS sampler are substantially smaller than those of the benchmark PG sampler and are comparable to those of the PGAS sampler with particle rejuvenation. Even though both of the PGAS samplers in *Algorithm 3-2* and *Algorithm 4-3* perform well with a small number of particles, *Algorithm 4-3* induces additional computational costs in implementing the CIS procedure in *Algorithm 4-1.* The relative computing times for *Algorithm 3-2* with $N = 20$ and *Algorithm 4-3* with $N = 20, M = 20$ are 0.2481 and 0.4340, respectively, compared with the benchmark PG sampler. Figure 1.A shows the autocorrelation functions (ACF) for the model parameters δ_1 and ϕ. The ACFs drop very quickly to zero with the small numbers of particles when *Algorithm 3-2* and *Algorithm 4-3* are applied. In contrast, the benchmark PG sampler in *Algorithm 2-2* does not mix well even with a large number of particles[16]. It is worth mentioning that the ancestor sampling in *Algorithm 3-1* and the particle rejuvenation in *Algorithm 4-2* significantly improve the mixing speed without explicitly incorporating the observation sequence $y_{1:t}$ in their importance distributions.

The simulation results for *Example 2* are presented in Table 1.B and Figure 1.B. The inefficiency factors in Table 1.B indicate that autocorrelations in MCMC draws are considerably removed when we work with the particle rejuvenation procedure to handle a degenerate NLG-SSSM. The PGAS sampler with particle rejuvenation yields significant efficiency gains even with the small number of particles. Conversely, the inefficiency factors of the standard PG sampler are noticeably high even with the large number of particles. The same conclusion is drawn from the ACFs in Figure 1.B where the ACFs based on *Algorithm 4-3* are always lower than those of *Algorithm 2-2.* In many applications, the particle rejuvenation in *Algorithm 4-3* can be simplified as in (25) and thus does not require huge computational costs. Nevertheless, it reduces the ACFs significantly and achieves faster mixing.

[16] The results from the benchmark PG sampler with a number of particles smaller than $N = 800$ are not reported because they do not converge to the correct stationary distribution even after a reasonable number of MCMC iterations.

4 Empirical Application: Regime-dependent Leverage Effect of U.S Stock Market

To illustrate the proposed estimation procedure, I estimate an extended version of the regime switching stochastic volatility (SV) model by So et al. (1998):

$$y_t = \mu + exp(\frac{x_{t-1}}{2})\epsilon_t, \tag{26}$$

$$x_t = \delta_{s_t} + \phi(x_{t-1} - \delta_{s_{t-1}}) + u_t, \tag{27}$$

$$\begin{bmatrix} \epsilon_t \\ u_t \end{bmatrix} \sim N(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \rho_{s_t}\sigma_u \\ \rho_{s_t}\sigma_u & \sigma_u^2 \end{bmatrix})$$

$$p(s_t = j|s_{t-1} = k) = \pi_{kj}, \ \sum_{j=1}^{K} \pi_{kj} = 1, \ i, k = 1, 2,$$

where y_t is the equity return at time t; x_{t-1} is the latent log-volatility at time t. Equivalently, the transition equation is represented by:

$$x_t = \delta_{s_t} + \phi(x_{t-1} - \delta_{s_{t-1}}) + \sigma_u(\rho_{s_t}\epsilon_t + \sqrt{1-\rho_{s_t}^2}\eta_t), \ \eta_t \sim N(0,1)$$

where $Corr(\epsilon_t, \eta_t) = 0$. By rewriting ϵ_t in terms of y_t, we have:

$$x_t = \delta_{s_t} + \phi(x_{t-1} - \delta_{s_{t-1}}) + \rho_{s_t}\sigma_u exp(-\frac{x_{t-1}}{2})(y_t - \mu) + \sigma_u\sqrt{1-\rho_{s_t}^2}\eta_t$$

The transition equation shows that the evolution of x_t is linear in y_t. Therefore, when ρ_{s_t} is negative, a drop in y_t leads to an increase in log-volatility at time $t+1$. This negative relationship is referred to as the leverage effect[17]. Yu (2005) demonstrated that discrete-time SV models with the leverage effect are theoretically[18] and empirically appealing.

In stochastic volatility literature, the volatility response to a shock to a stock return is often assumed to be constant over time. However, Bandi and Renò (2012) recently raised the possibility that the time-varying leverage effect as a higher leverage ratio should render a larger variance in the response to a shock to a return[19]. More specifically, they specify the leverage effect as a function of spot volatility and distinguish

[17] Harvey and Shephard (1996) first propose a discrete time SV model to capture the leverage effect and estimate it using a a quasi-maximum likelihood method.

[18] The SV model with the leverage effect is the Euler approximation to the well-known continuous-time asymmetric stochastic volatility model broadly used in the option price literature.

[19] Daouk and Ng (2011) also showed that the leverage effect is larger in down markets than in up markets. Yu (2012) emphasized that the magnitude of the leverage effect depends on the size and the direction of the shocks to the previous stock price.

volatility regimes by choosing deterministic threshold values of spot volatility in their parametric models. Following Bandi and Renò (2012), the proposed regime switching SV model in (26) and (27) accommodates the regime-dependent leverage effect. However, the main difference is that the leverage effect is specified as a function of the regime-specific means of log volatility, and corresponding regimes are endogenously estimated in the proposed model instead of using deterministic threshold values.

Bayesian inference using the proposed SV model is difficult in that the existing MCMC algorithms in the literature are not directly applicable. For example, Omori et al. (2007) used the idea of approximating the joint distribution of the two correlated innovations in the SV model with ten mixture normal distributions[20]. However, because the correlation parameter ρ_{s_t} shifts with unknown timings, the approximation of the joint distribution by the mixture normals becomes infeasible. Alternatively, one may attempt to use a single-move algorithm, such as the one adopted by Yu (2012). This approach is also problematic because the single-move approach is difficult to implement due to the very persistent latent regime-indicator variable.

The proposed PGAS algorithm is employed to estimate the regime switching SV model with the regime-dependent leverage effect. The proposed SV model with 2-state and 3-state regimes is applied to daily S&P 500 and NASDAQ returns from January 2, 1997 to August 5, 2015 to analyze how the leverage effect changes depending on volatility regimes. The empirical results for the 2-state regime switching model are reported in Table 2 and Figures 2 and 3. First, it can be seen in Figures 2.A and 3.A. that the posterior probability of high volatility regime very sharply changes, leaving a low uncertainty in the timings of regime shifts for S&P 500 and NASDAQ returns. Second, it can be clearly seen from Figures 2.B and 3.B that the posterior means of the regime-dependent correlation parameters are noticeably different across high- and low-volatility regimes. Table 2 shows that in a low-volatility regime, the posterior mean of the correlation parameter is estimated to be -0.5, while it is -0.672 in a high-volatility regime for S&P 500 returns. This difference is even more substantial for NASDAQ returns. The posterior mean of the correlation parameter is -0.269 in the low-volatility regime and -0.653 in the high-volatility regime. These Bayesian estimates produce ample evidence for the presence of the regime-dependent leverage effect.

The 3-state regime switching SV model produces similar results, which are shown in Table 3 and Figures 4 and 5. The posterior mean of the correlation parameter for NASDAQ returns is estimated to be -0.667 in a high-volatility regime, -0.596 in a medium-volatility regime and -0.221 in a low-volatility regime. For S&P 500 returns, the correlation parameter in a high-volatility regime cannot be clearly identified due to a small sample size. However, for medium- and low-volatility regimes where large samples are available, Figure 4.B

[20] For SV models without leverage, Kim et al. (1998) and Chib et al. (2002) propose efficient multi-move algorithms based on a log-squared transformation of return and mixture normal approximation to the distribution of a transformed shock.

illustrates a similar point. The posterior mean of the correlation parameter in a medium-volatility regime is reasonably larger in absolute value than that in a low-volatility regime.

To formally compare the considered SV models, I adopt the Deviance Information Criterion (DIC) by Spiegelhalter et al. (2002):

$$DIC = -2E_{\zeta|Y}[lnf(Y|\zeta)] + 2\{lnf(Y|\bar{\zeta})] - E_{\zeta|Y}[lnf(Y|\zeta)]\}$$

where Y is the whole observation sequence; ζ encompasses the model parameters and the latent variables; and $\bar{\zeta}$ is the posterior means of ζ. DIC has been widely used in the literature for comparing complex hierarchical models. It consists of two components, a term measuring Bayesian goodness of fit and a term imposing a penalty on model complexity. The first term $-2E_{\zeta|Y}[lnf(Y|\zeta)]$ is defined as the posterior expectation of the deviance, and the second term $2\{lnf(Y|\bar{\zeta})] - E_{\zeta|Y}[lnf(Y|\zeta)]\}$ is defined as the difference between the posterior mean of the deviance and the deviance evaluated at $\bar{\zeta}$. The posterior mean of the deviance is easily obtained using the MCMC draws for the latent state variables and the model parameters:

$$lnf(Y|\zeta)] \approx \frac{1}{R}\sum_{r=1}^{R} lnf(Y|\zeta^{(r)}).$$

The first and third columns of Table 4 show DICs for S&P 500 and NASDAQ returns. In Table 4, I also consider 2-state and 3-state regime-switching SV models with the constant leverage effect for complete comparison. Based on DIC, the most preferred model is the 3-state regime-switching SV model with the regime-dependent leverage effect. It is also interesting to see that the 3-state regime-switching SV models are always preferred to the 2-state regime-switching SV models, regardless of the nature of the leverage effect. A close look at the second and fourth columns of Table 4 reveals that the models including the regime-dependent leverage effect are preferred, given the same number of regimes. Therefore, the newly proposed SV models confirm that the time-varying leverage effect in the U.S. stock market is indeed an important feature and is governed by the long-run mean of the volatility process.

5 Concluding Remarks

In summary, this article has developed and illustrated two efficient Bayesian methods to estimate nonlinear/non-Gaussian switching state space models based on particle ancestor sampling and particle rejuvenation. In particular, a special attention has been paid to develop posterior simulation procedures for the continuous-state and discrete-regime indicator variables. It has been demonstrated that the proposed algorithms do not require a large number of particles to achieve fast mixing and thus allow for fast convergence to the posterior distribution, in contrast to the benchmark PG sampler and Particle Marginal Metropolis-Hastings methods, which usually involve unnecessary accept/reject steps. Importantly, the PG sampler with particle rejuvenation is shown to perform perfectly even when a non-linear/non-Gaussian switching state space model of interest involves a degenerate transition equation, which results in zero probability of updating an ancestor index in ancestor sampling. The proposed PG samplers are easy to implement in practice and can be applied to models without the Markovian property. According to a general PMCMC scheme suggested by Mendes et al. (2014), Particle Marginal Metropolis-Hastings (PMMH) can be incorporated to simulate the model parameters that are strongly correlated with the latent variables, which will result in further improvements in the proposed algorithms. By applying the proposed PG sampler with ancestor sampling to Standard and Poor's 500 and NASDAQ daily return data, this article shows that stronger (weaker) leverage effects are associated with a high (low) -volatility regime within parametric regime-switching stochastic volatility models, which is consistent with the important empirical findings of Bandi and Renò (2012).

A Appendix

A.1 Derivation of Optimal Incremental Importance Distribution

In generating the new states $\{x_t^{(i)}, s_t^{(i)}\}$ in forward filtering, the optimal incremental importance distribution is given as:

$$q(x_t, s_t|x_{0:t-1}, s_{0:t-1}) = p_\theta(x_t, s_t|x_{0:t-1}, s_{0:t-1}, y_{1:t})p_\theta(y_t|x_{0:t-1}, s_{0:t-1}, y_{1:t-1})$$

The former component are decomposed into two parts:

$$p_\theta(x_t, s_t|x_{0:t-1}, s_{0:t-1}, y_{1:t}) = p_\theta(x_t|x_{0:t-1}, s_{0:t}, y_{1:t})p_\theta(s_t|x_{0:t-1}, s_{0:t-1}, y_{1:t})$$

To derive an importance density close to the target, we can show:

$$\begin{aligned} p_\theta(s_t|x_{0:t-1}, s_{0:t-1}, y_{1:t}) &= \frac{p_\theta(s_t, y_t|x_{0:t-1}, s_{0:t-1}, y_{1:t-1})}{p_\theta(y_t|x_{0:t-1}, s_{0:t-1}, y_{1:t-1}))} \\ &\propto p_\theta(s_t, y_t|x_{0:t-1}, s_{0:t-1}, y_{1:t-1}) \\ &= p_\theta(y_t|x_{0:t-1}, s_{0:t}, y_{1:t-1})p_\theta(s_t|x_{0:t-1}, s_{0:t-1}, y_{1:t-1}) \\ &\propto p_\theta(y_t|x_{0:t-1}, s_{0:t}, y_{1:t-1})g_\theta(s_t|s_{t-1}) \end{aligned}$$

The validity of going from the second line to the third line is that all the past information on $y_{1:t-1}$, and

$x_{0:t-1}$ is not relevant for s_t conditional on s_{t-1}. The density $p_\theta(y_t|x_{0:t-1}, s_{0:t}, y_{1:t-1})$ is given as:

$$\begin{aligned} p_\theta(y_t|x_{0:t-1}, s_{0:t}, y_{1:t-1}) &= \int p_\theta(y_t, x_t|x_{0:t-1}, s_{0:t}, y_{1:t-1})dx_t \\ &= \int f_\theta(y_t|x_{0:t}, s_{0:t}, y_{1:t-1})p_\theta(x_t|x_{0:t-1}, s_{0:t}, y_{1:t-1})dx_t \end{aligned}$$

Finally, the second term in the target density is given as:

$$\begin{aligned} p_\theta(y_t|x_{0:t-1}, s_{0:t-1}, y_{1:t-1}) &= \sum_{s_t} p_\theta(y_t, s_t|x_{0:t-1}, s_{0:t-1}, y_{1:t-1}) \\ &= \sum_{s_t} p_\theta(y_t|x_{0:t-1}, s_{0:t}, y_{1:t-1})g_\theta(s_t|s_{t-1}) \end{aligned}$$

Notice that the density $p_\theta(y_t|x_{0:t-1}, s_{0:t}, y_{1:t-1})$ is included both in the second component of the importance distribution and in the conditional density for s_t as well. The density $p_\theta(y_t|x_{0:t-1}, s_{0:t}, y_{1:t-1})$ is not analytically tractable and thus the density should be approximated to construct the increment importance density.

A.2 Proof of Proposition 1

In what follows, the set of the latent variables $\{x_t, s_t\}$ is denoted by z_t for notational simplicity. By the hierarchical structure of the model in equation (1), the transition density of z_t conditional on θ is given by $g_\theta(z_t|z_{0:t-1}) = g_\theta(x_t|x_{0:t-1}, s_{0:t})g_\theta(s_t|s_{t-1})$.

Using the definition of the importance weight in equation (7), we rewrite the posterior density of the latent states

$$\begin{aligned} p_\theta(z_{0:t}|y_{1:t}) &= \frac{p_\theta(z_{0:t}, y_{1:t})}{p_\theta(y_{1:t})} \\ &= \frac{1}{p_\theta(y_{1:t})}p_\theta(z_0)\left[\prod_{l=1}^{t} f_\theta(y_l|z_{0:l})p_\theta(z_l|z_{0:l-1})\right] \qquad (A.1) \\ &= \frac{1}{p_\theta(y_{1:t})}\bar{\omega}_0 q(z_0)\left[\prod_{l=1}^{t} \bar{\omega}_l q(z_l|z_{0:l-1})\right] \end{aligned}$$

where $\bar{\omega}_l = \frac{f_\theta(y_l|z_{0:l})g_\theta(z_l|z_{0:l-1})}{q(z_l|z_{0:l-1})}$ for $l = 1, 2, ..., t$. Therefore, for a reference particle trajectory up to time t, we have

$$
\begin{aligned}
p_\theta(z_{0:t}^{(b_{0:t})}|y_{1:t}) &= \frac{1}{p_\theta(y_{1:t})}\bar{\omega}_0^{(b_0)}q(z_0^{(b_0)})\Bigg[\prod_{l=1}^{t}\bar{\omega}_l^{(b_l)}q(z_l^{(b_l)}|z_{0:l-1}^{(b_{0:l-1})})\Bigg] \\
&= \frac{1}{p_\theta(y_{1:t})}\Bigg[\prod_{l=0}^{t}\sum_{j}^{N}\bar{\omega}_l^{(j)}\Bigg]\frac{\bar{\omega}_0^{(b_0)}}{\sum_j^N \bar{\omega}_0^{(j)}}q(z_0^{(b_0)})\Bigg[\prod_{l=1}^{t}\frac{\bar{\omega}_l^{(b_l)}}{\sum_j^N \bar{\omega}_l^{(j)}}q(z_l^{(b_l)}|z_{0:l-1}^{(b_{0:l-1})})\Bigg] \\
&= \frac{1}{p_\theta(y_{1:t})}\Bigg[\prod_{l=0}^{t}\sum_{j}^{N}\bar{\omega}_l^{(j)}\Bigg]\hat{\omega}_0^{(b_0)}q(z_0^{(b_0)})\Bigg[\prod_{l=1}^{t}\hat{\omega}_l^{(b_l)}q(z_l^{(b_l)}|z_{0:l-1}^{(b_{0:l-1})})\Bigg] \\
&= \frac{1}{p_\theta(y_{1:t})}\Bigg[\prod_{l=0}^{t}\sum_{j}^{N}\bar{\omega}_l^{(j)}\Bigg]q(z_0^{(b_0)})\Bigg[\prod_{l=1}^{t}M_s^\theta(a_l^{(b_l)}, z_l^{(b_l)})\Bigg]\hat{\omega}_t^{(b_t)}
\end{aligned}
\tag{A.2}
$$

where $M_l^\theta(a_l^{(b_l)}, z_l^{(b_l)}) = \hat{\omega}_{l-1}^{(b_{l-1})}q(z_l^{(b_l)}|z_{0:l-1}^{(b_{0:l-1})})$ and $\hat{\omega}_l^{(b_l)} = \frac{\bar{\omega}_l^{(b_l)}}{\sum_j^N \bar{\omega}_l^{(j)}}$. By plugging equation (A.2) in the extended target density in equation (14), we have:

$$
\begin{aligned}
\Phi(\theta, Z_{0:T}, A_{1:T}, K) &\equiv \frac{1}{N^{T+1}}p(\theta, z_{0:T}^{(b_{0:T})}|y_{1:T})\prod_{\substack{i=1\\ i\neq b_0}}^{N}q(z_0^{(i)})\prod_{t=1}^{T}\Bigg[\prod_{\substack{i=1\\ i\neq b_t}}^{N}M_t^\theta(a_t^{(i)}, z_t^{(i)})\Bigg] \\
&= \frac{1}{N^{T+1}}\frac{p(\theta)p_\theta(y_{1:T})p_\theta(z_{0:T}^{(b_{0:T})}|y_{1:T})}{p(y_{1:T})}\prod_{\substack{i=1\\ i\neq b_0}}^{N}q(z_0^{(i)})\prod_{t=1}^{T}\Bigg[\prod_{\substack{i=1\\ i\neq b_t}}^{N}M_t^\theta(a_t^{(i)}, z_t^{(i)})\Bigg] \\
&\propto \frac{1}{N^{T+1}}p(\theta)p_\theta(y_{1:T})p_\theta(z_{0:T}^{(b_{0:T})}|y_{1:T})\prod_{\substack{i=1\\ i\neq b_0}}^{N}q(z_0^{(i)})\prod_{t=1}^{T}\Bigg[\prod_{\substack{i=1\\ i\neq b_t}}^{N}M_t^\theta(a_t^{(i)}, z_t^{(i)})\Bigg] \\
&= \frac{p(\theta)}{N^{T+1}}\Bigg[\prod_{t=0}^{T}\sum_{j}^{N}\bar{\omega}_t^{(j)}\Bigg]\prod_{i=1}^{N}q(z_0^{(i)})\prod_{t=1}^{T}\Bigg[\prod_{i=1}^{N}M_t^\theta(a_t^{(i)}, z_t^{(i)})\Bigg]\hat{\omega}_T^{(b_T)} \\
&= p(\theta)\hat{Z}_T^N(\theta)\Phi(Z_{0:T}, A_{1:T}|\theta)\hat{\omega}_T^{(K)}
\end{aligned}
\tag{A.3}
$$

where $\hat{Z}_T^N(\theta) = \Bigg[\prod_{t=0}^{T}\frac{1}{N}\sum_j^N \bar{\omega}_t^{(j)}\Bigg]$ and $K = b_T$. Notice that $\hat{Z}_T^N(\theta)$ is the particle estimate of $Z_T(\theta) = p_\theta(y_{1:T})$. Therefore,

$$\Phi(K|\theta, Z_{0:T}, A_{1:T}) \propto \Phi(\theta, Z_{0:T}, A_{1:T}, K) \propto \hat{w}_T^{(K)}.$$

A.3 Proof of Proposition 2

The conditional in *Proposition 2* is proportional to

$$\begin{aligned}
\Phi(b_{t-1}|\theta,&Z_{0:t-1}^{(-b_{0:t-1})}, A_{1:t-1}^{(-b_{1:t-1})}, z_{0:T}^{(b_{0:T})}, b_{0:t-2}, b_{t:T}) \\
&= \Phi(b_{t-1}|\theta, Z_{0:t-1}, A_{1:t-1}, z_{t:T}^{(b_{t:T})}, b_{t:T}) \\
&\propto \Phi(\theta, Z_{0:t-1}, A_{1:t-1}, z_{t:T}^{(b_{t:T})}, b_{t-1:T}) \\
&= \Phi(\theta, z_{0:T}^{(b_{0:T})}, b_{0:T})\Phi(Z_{0:t-1},{}^{(-b_{0:t-1})} A_{1:t-1}^{(-b_{1:t-1})}|\theta, z_{0:T}^{(b_{0:T})}, b_{0:T}) \\
&= \frac{1}{N^{T+1}} p(\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}|y_{1:T}) \prod_{\substack{i=1\\ i\neq b_0}}^{N} q(z_0^{(i)}) \times \prod_{l=1}^{t-1} \left[\prod_{\substack{i=1\\ i\neq b_l}}^{N} M_l(a_l^{(i)}, z_l^{(i)}) \right] \\
&= \frac{p(\theta|y_{1:T})}{N^{T+1}} p_\theta(z_{0:T}^{(b_{0:T})}|y_{1:T}) \prod_{\substack{i=1\\ i\neq b_0}}^{N} q(z_0^{(i)}) \times \prod_{l=1}^{t-1} \left[\prod_{\substack{i=1\\ i\neq b_s}}^{N} M_l(a_l^{(i)}, z_l^{(i)}) \right] \qquad (A.4) \\
&= \frac{p(\theta|y_{1:T})}{N^{T+1}} \frac{p_\theta(z_{0:T}^{(b_{0:T})}|y_{1:T})}{p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1})} p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1}) \prod_{\substack{i=1\\ i\neq b_0}}^{N} q(z_0^{(i)}) \times \prod_{l=1}^{t-1} \left[\prod_{\substack{i=1\\ i\neq b_s}}^{N} M_l(a_l^{(i)}, z_l^{(i)}) \right] \\
&\propto \frac{p_\theta(z_{0:T}^{(b_{0:T})}|y_{1:T})}{p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1})} p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1}) \\
&\propto \prod_{l=t}^{T} p_\theta(y_l|z_{0:l}^{(b_{0:l})}) p_\theta(z_l|z_{0:l-1}^{(b_{0:l-1})})\ \hat{\omega}_{t-1}^{(b_{t-1})}
\end{aligned}$$

because $p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1}) \propto \hat{\omega}_t^{(b_{t-1})}$ by equation (A.2) and

$$\begin{aligned}
\frac{p_\theta(z_{0:T}^{(b_{0:T})}|y_{1:T})}{p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1})} &= \frac{p_\theta(z_{0:t-1}^{(b_{0:t-1})}, z_{t:T}^{(b_{t:T})}|y_{1:t-1}, y_{t:T})}{p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1})} \\
&= \frac{p_\theta(z_{t:T}^{(b_{t:T})}, y_{t:T}|z_{0:t-1}^{(b_{0:t-1})}, y_{1:t-1}) p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1})}{p_\theta(y_{t:T}|y_{1:t-1}) p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1})} \\
&\propto p_\theta(z_{t:T}^{(b_{t:T})}, y_{t:T}|z_{0:t-1}^{(b_{0:t-1})}) \qquad (A.5) \\
&= \prod_{l=t}^{T} p_\theta(y_l|z_{0:l}^{(b_{0:l})}) p_\theta(z_l|z_{0:l-1}^{(b_{0:l-1})}) \\
&\propto \left[\prod_{l=t}^{T} f_\theta(y_l|x_{0:l}^{(b_{0:l})}, s_{0:l}^{(b_{0:l})}) g_\theta(x_l|x_{0:l-1}^{(b_{0:l-1})}, s_{0:l}^{(b_{0:l})}) g_\theta(s_l^{(b_l)}|s_{l-1}^{(b_{l-1})}) \right].
\end{aligned}$$

In the fourth line of equation (A.5), the irrelevant observation sequence $y_{1:t-1}$ is dropped. This completes the proof of proposition 2.

A.4 Proof of Proposition 3

The proof of proposition 3 is similar to the proof of Proposition 2.

$$\begin{aligned}
\Phi(z_t^{(b_t)}, b_{t-1}|\theta, & Z_{0:t-1}^{(-b_{0:t-1})}, A_{1:t-1}^{(-b_{1:t-1})}, z_{0:t-1}^{(b_{0:t-1})}, z_{t+1:T}^{(b_{t+1:T})}, b_{0:t-2}, b_{t:T}) \\
&= \Phi(z_t^{(b_t)}, b_{t-1}|\theta, Z_{0:t-1}, A_{1:t-1}, z_{t+1:T}^{(b_{t+1:T})}, b_{t:T}) \\
&\propto \Phi(\theta, Z_{0:t-1}, A_{1:t-1}, z_{t:T}^{(b_{t:T})}, b_{t-1:T}) \\
&= \Phi(\theta, z_{0:T}^{(b_{0:T})}, b_{0:T})\Phi(Z_{0:t-1},{}^{(-b_{0:t-1})} A_{1:t-1}^{(-b_{1:t-1})}|\theta, z_{0:T}^{(b_{0:T})}, b_{0:T}) \\
&= \frac{1}{N^{T+1}} p(\theta, x_{0:T}^{(b_{0:T})}, s_{0:T}^{(b_{0:T})}|y_{1:T}) \prod_{\substack{i=1 \\ i \neq b_0}}^{N} q(z_0^{(i)}) \times \prod_{l=1}^{t-1} \left[\prod_{\substack{i=1 \\ i \neq b_l}}^{N} M_l(a_l^{(i)}, z_l^{(i)}) \right] \\
&\propto \frac{p(\theta|y_{1:T})}{N^{T+1}} p_\theta(z_{0:T}^{(b_{0:T})}|y_{1:T}) = \frac{p(\theta|y_{1:T})}{N^{T+1}} \frac{p_\theta(z_{0:T}^{(b_{0:T})}|y_{1:T})}{p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1})} p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1}) \qquad (A.6) \\
&\propto \frac{p_\theta(z_{0:T}^{(b_{0:T})}|y_{1:T})}{p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1})} p_\theta(z_{0:t-1}^{(b_{0:t-1})}|y_{1:t-1}) \\
&\propto \prod_{l=t}^{T} p_\theta(y_l|z_{0:l}^{(b_{0:l})}) p_\theta(z_l|z_{0:l-1}^{(b_{0:l-1})}) \ \hat{\omega}_{t-1}^{(b_{t-1})} \\
&\propto \left[\prod_{l=t}^{T} f_\theta(y_l|x_{0:l}^{(b_{0:l})}, s_{0:l}^{(b_{0:l})}) g_\theta(x_l|x_{0:l-1}^{(b_{0:l-1})}, s_{0:l}^{(b_{0:l})}) g_\theta(s_l^{(b_l)}|s_{l-1}^{(b_{l-1})}) \right] \hat{\omega}_{t-1}^{(b_{t-1})}
\end{aligned}$$

In the seventh line, we use equation (A.5). This completes the proof of proposition 3.

A.5 Derivation of a Candidate Distribution for x_t

An importance distribution is derived for the latent variable x_t conditional on the regime indication variable s_t in a special case of a linear/Gaussian transition equation. The latent variable x_t follows:

$$x_t = \delta_{s_t} + H_{s_t}(x_t - x_{s_{t-1}}) + e_t, \quad e_t \sim i.i.d.N(0, \Sigma_{s_t})$$

where H is the transition matrix. The transition density at time $t+1$ and t is given by:

$$
\begin{aligned}
g(x_t) &= p_\theta(x_t|x_{t+1}, x_t, s_{t+1}, s_t, s_{t-1}) \\
&\propto p_\theta(x_{t+1}|x_t, s_{t+1}, s_t)p_\theta(x_t|x_{t-1}, s_t, s_{t-1}) \\
&\propto exp\Big[-\frac{1}{2}(x^*_{t+1} - H_{s_{t+1}}x^*_t)'\Sigma^{-1}_{s_{t+1}}(x^*_{t+1} - H_{s_{t+1}}x^*_t) - \frac{1}{2}(x^*_t - H_{s_t}x^*_{t-1})'\Sigma^{-1}_{s_t}(x^*_t - H_{s_t}x^*_{t-1})\Big] \\
&\propto exp\Big[-\frac{1}{2}\Big[x^{*'}_t H'_{s_{t+1}}\Sigma^{-1}_{s_{t+1}}H_{s_{t+1}}x^*_t - 2x^{*'}_t H'_{s_{t+1}}\Sigma^{-1}_{s_{t+1}}x^*_{t+1} + x^{*'}_t\Sigma^{-1}_{s_t}x^*_t - 2x^{*'}_t\Sigma^{-1}_{s_t}H_{s_t}x^*_{t-1}\Big]\Big] \\
&\propto exp\Big[-\frac{1}{2}\Big[x^{*'}_t(H'_{s_{t+1}}\Sigma^{-1}_{s_{t+1}}H_{s_{t+1}} + \Sigma^{-1}_{s_t})x^*_t - 2x^{*'}_t(H'_{s_{t+1}}\Sigma^{-1}_{s_{t+1}}x^*_{t+1} + \Sigma^{-1}_{s_t}H_{s_t}x^*_{t-1})\Big]\Big] \\
&\propto exp\Big[-\frac{1}{2}\ (x^*_t - \mu^*_t)'V^{-1}_t(x^*_t - \mu^*_t)\Big]
\end{aligned}
$$

where $x^*_t = x_t - \delta_{s_t}$; $V_t = (H'_{s_{t+1}}\Sigma^{-1}_{s_{t+1}}H_{s_{t+1}} + \Sigma^{-1}_{s_t})^{-1}$; $\mu^*_t = V_t(H'_{s_{t+1}}\Sigma^{-1}_{s_{t+1}}x^*_{t+1} + \Sigma^{-1}_{s_t}H_{s_t}x^*_{t-1})$. Thus,

$$x_t|x_{t+1}, x_t, s_{t+1}, s_t, s_{t-1} \sim N(\mu_t, V_t)$$

where $\mu_t = \delta_{s_t} + \mu^*_t$.

References

[1] An, Sungbae, and Frank Schorfheide. "Bayesian analysis of DSGE models." *Econometric reviews* 26.2-4 (2007): 113-172.

[2] Andrieu, Christophe, Arnaud Doucet, and Roman Holenstein. "Particle markov chain monte carlo methods." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 72.3 (2010): 269-342.

[3] Andrieu, Christophe, Manuel Davy, and Arnaud Doucet. "Efficient particle filtering for jump Markov systems. Application to time-varying autoregressions." *Signal Processing, IEEE Transactions* on 51.7 (2003): 1762-1770.

[4] Bai, Jushan, and Serena Ng. "Boosting diffusion indices." *Journal of Applied Econometrics* 24.4 (2009): 607-629.

[5] Bandi, Federico M., and Roberto Renò. "Time-varying leverage effects." *Journal of Econometrics* 169.1 (2012): 94-113.

[6] Black, F. "Studies of stock market volatility changes" *Proceedings of the American Statistical Association, Business and Economic Statistics Section* (1976): pp. 177-181

[7] Bunch, Pete, Fredrik Lindsten, and Sumeetpal Singh. "Particle Gibbs with refreshed backward simulation." *Acoustics, Speech and Signal Processing (ICASSP)*, 2015 IEEE International Conference on. IEEE, 2015.

[8] Carter, Christopher K., Eduardo F. Mendes, and Robert Kohn. "An extended space approach for particle Markov chain Monte Carlo methods." arXiv preprint arXiv:1406.5795 (2014).

[9] Chib, Siddhartha, Federico Nardari, and Neil Shephard. "Markov chain Monte Carlo methods for stochastic volatility models." *Journal of Econometrics* 108.2 (2002): 281-316.

[10] Christie, Andrew A. "The stochastic behavior of common stock variances: Value, leverage and interest rate effects." *Journal of financial Economics* 10.4 (1982): 407-432.

[11] Daouk, Hazem, and David Ng. "Is unlevered firm volatility asymmetric?." *Journal of Empirical Finance* 18.4 (2011): 634-651.

[12] Driessen, Hans, and Yvo Boers. "Efficient particle filter for jump Markov nonlinear systems." *IEE Proceedings-radar, sonar and navigation* 152.5 (2005): 323-326.

[13] Forni, Mario, Marc Hallin, Marco Lippi, and Lucrezia Reichlin. "The generalized dynamic-factor model: Identification and estimation." *Review of Economics and statistics* 82.4 (2000): 540-554.

[14] Fruhwirth-Schnatter, Sylvia. Finite mixture and Markov switching models. *Springer Science & Business Media*, (2006).

[15] Geweke, John. "Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments (with discussion)." *Bayesian Statistics* 4 (1992): 169-194.

[16] Giordani, Paolo, Robert Kohn, and Dick van Dijk. "A unified approach to nonlinearity, structural change, and outliers." *Journal of Econometrics* 137.1 (2007): 112-133.

[17] Godsill, Simon J., Arnaud Doucet, and Mike West. "Monte Carlo smoothing for nonlinear time series." *Journal of the american statistical association* 99.465 (2004).

[18] Gordon, Neil J., David J. Salmond, and Adrian FM Smith. "Novel approach to nonlinear/non-Gaussian Bayesian state estimation." *IEE Proceedings F (Radar and Signal Processing)*. Vol. 140. No. 2. IET Digital Library, 1993.

[19] Hamilton, James D. "A new approach to the economic analysis of nonstationary time series and the business cycle." *Econometrica: Journal of the Econometric Society* (1989): 357-384.

[20] Harvey, Andrew C., and Neil Shephard. ”Estimation of an asymmetric stochastic volatility model for asset returns.” *Journal of Business & Economic Statistics* 14.4 (1996): 429-434.

[21] Kim, Chang-Jin. ”Dynamic linear models with Markov-switching.” *Journal of Econometrics* 60.1 (1994): 1-22.

[22] Kim, Chang-Jin, and Charles R. Nelson. State-space models with regime switching: classical and Gibbs-sampling approaches with applications. Vol. 2. Cambridge: MIT press, (1999).

[23] Kim, Chang-Jin, and Jaeho Kim. ”Bayesian Inference in Regime-Switching ARMA Models with Absorbing States: The Dynamics of the Ex-Ante Real Interest Rate Under Structural Breaks.” Forthcoming *Journal of Business & Economic Statistics* (2014).

[24] Kim, Sangjoon, Neil Shephard, and Siddhartha Chib. ”Stochastic volatility: likelihood inference and comparison with ARCH models.” *The Review of Economic Studies* 65.3 (1998): 361-393.

[25] Koop, Gary, and Simon M. Potter. ”Estimation and forecasting in models with multiple breaks.” *The Review of Economic Studies* 74.3 (2007): 763-789.

[26] Lindsten, Fredrik, Michael I. Jordan, and Thomas B. Schon. ”Particle Gibbs with ancestor sampling.” *The Journal of Machine Learning Research* 15.1 (2014): 2145-2184.

[27] Lindsten, Fredrik, Pete Bunch, Sumeetpal S. Singh, and Thomas B. Schon. ”Particle ancestor sampling for near-degenerate or intractable state transition models.” arXiv preprint arXiv:1505.06356 (2015).

[28] Lindsten, Fredrik, and T. B. Schon. ”On the use of backward simulation in the particle Gibbs sampler.” *Acoustics, Speech and Signal Processing (ICASSP)*, 2012 IEEE International Conference on. IEEE, (2012).

[29] Liu, Jun S., Wing Hung Wong, and Augustine Kong. ”Covariance structure of the Gibbs sampler with applications to the comparisons of estimators and augmentation schemes.” *Biometrika* 81.1 (1994): 27-40.

[30] Ludvigson, Sydney C., and Serena Ng. ”The empirical risk-return relation: a factor analysis approach.” *Journal of Financial Economics* 83.1 (2007): 171-222.

[31] Mendes, Eduardo F., Christopher K. Carter, and Robert Kohn. ”On general sampling schemes for Particle Markov chain Monte Carlo methods.” arXiv preprint arXiv:1401.1667 (2014).

[32] Nonejad, Nima. ”Particle Gibbs with ancestor sampling for stochastic volatility models with: heavy tails, in mean effects, leverage, serial dependence and structural breaks.” *Studies in Nonlinear Dynamics & Econometrics* (2014).

[33] Omori, Yasuhiro, Siddhartha Chib, Neil Shephard, and Jouchi Nakajima. ”Stochastic volatility with leverage: Fast and efficient likelihood inference.” *Journal of Econometrics* 140, no. 2 (2007): 425-449.

[34] Pesaran, M. Hashem, Davide Pettenuzzo, and Allan Timmermann. ”Forecasting time series subject to multiple structural breaks.” *The Review of Economic Studies* 73.4 (2006): 1057-1084.

[35] Pitt, Michael K., et al. ”On some properties of Markov chain Monte Carlo simulation methods based on the particle filter.” *Journal of Econometrics* 171.2 (2012): 134-151.

[36] Pitt, Michael K., and Neil Shephard. ”Filtering via simulation: Auxiliary particle filters.” *Journal of the American statistical association* 94.446 (1999): 590-599.

[37] Scott, Steven L. ”Bayesian methods for hidden Markov models.” *Journal of the American Statistical Association* 97.457 (2002).

[38] Smets, Frank, and Rafael Wouters. ”Shocks and Frictions in US Business Cycles: A Bayesian DSGE Approach.” *The American Economic Review* 97.3 (2007): 586-606.

[39] So, Mike EC P., Kin Lam, and Wai Keung Li. ”A stochastic volatility model with Markov switching.” *Journal of Business & Economic Statistics* 16.2 (1998): 244-253.

[40] Song, Dongho. ”Bond Market Exposures to Macroeconomic and Monetary Policy Risks.” (2014).

[41] Spiegelhalter, David J., Nicola G. Best, Bradley P. Carlin, and Angelika Van Der Linde. ”Bayesian measures of model complexity and fit.” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 64.4 (2002): 583-639.

[42] Flury, Thomas, and Neil Shephard. ”Bayesian inference based only on simulated likelihood: particle filter analysis of dynamic economic models.” *Econometric Theory* 27.05 (2011): 933-956.

[43] Van Dyk, David A., and Taeyoung Park. ”Partially collapsed Gibbs samplers: Theory and methods.” *Journal of the American Statistical Association* 103.482 (2008): 790-796.

[44] Wan, Eric A., and Rudolph Van Der Merwe. ”The unscented Kalman filter.” *Kalman filtering and neural networks* (2001): 221-280.

[45] Whiteley, Nick. "Discussion on Particle markov chain monte carlo methods." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 72.3 (2010): 306-307.

[46] Whiteley, Nick, Christophe Andrieu, and Arnaud Doucet. "Efficient Bayesian inference for switching state-space models using discrete particle Markov chain Monte Carlo methods." working paper (2010).

[47] Yu, Jun. "On leverage in a stochastic volatility model." *Journal of Econometrics* 127.2 (2005): 165-178.

[48] Yu, Jun. "A semiparametric stochastic volatility model." *Journal of Econometrics* 167.2 (2012): 473-482.

Table 1.A. Efficiency Evaluation: Example 1 [Geweke's (1992) Inefficiency Factor]

$$y_t = \exp\left(\frac{x_{t-1}}{2}\right)\varepsilon_t, \;\; \varepsilon_t \sim N\,(0,1),$$

$$x_t = \delta_{s_t} + \phi\left(x_{t-1} - \delta_{s_{t-1}}\right) + u_t, \;\; u_t \sim N\,(0, \sigma_u^2),$$

$$\begin{bmatrix} \varepsilon_t \\ u_t \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & \sigma_u^2 \end{bmatrix}\right),$$

$$\Pr[S_t = j |\; S_{t-1} = \mathrm{i}] = \pi_{ij}, \quad for\ i = 1,2\ and\ j = 1,2.$$

$$[\delta_1 = -1, \delta_2 = 0.5, \phi = 0.5, \sigma_u^2 = 0.04, \pi_{11} = 0.99, \pi_{22} = 0.99]$$

	$\kappa_J = 1 + 2\sum_{j=1}^{J} \rho_j\,,\;\; J = 2{,}000$		
	PG	PGAS	PGAS-PR

Model Parameters	N = 1000	N = 20	N = 20
			M = 20

δ_1	439.62	22.28	29.80
δ_2	489.25	51.94	71.75
ϕ	2287.59	813.05	679.50
σ_u^2	3372.89	901.02	873.28
π_{11}	7.72	2.14	2.51
π_{22}	10.27	3.02	2.74
Relative Computing Time	1	0.2481	0.4340

Note: 1. The empirical autocorrelation functions are obtained based on 40,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in. Running time for the PG sampler is 50901.04sec.

2. The average of the inefficient factors from 5 simulations are reported.

3. PG refers to the benchmark PG sampler.

4. PGAS refers to the proposed PG sampler with ancestor sampling.

5. PGAS-PR refers to the proposed PG sampler with ancestor sampling and particle rejuvenation.

Table 1.B. Efficiency Evaluation: Example 2 [Geweke's (1992) Inefficiency Factor]

$$y_{1,t} = \beta_0 + \beta_1 f_t + \beta_2 {f_t}^2 + \varepsilon_{1,t}, \;\; \varepsilon_{1,t} \sim N\left(0, \sigma^2_{\varepsilon_1}\right),$$

$$y_{2,t} = \alpha_0 + f_t + \varepsilon_{2,t}, \;\; \varepsilon_{2,t} \sim N\left(0, \sigma^2_{\varepsilon_2}\right),$$

$$\begin{bmatrix} f_t \\ u_t \end{bmatrix} = \begin{bmatrix} \phi & -\gamma \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_{t-1} \\ u_{t-1} \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u_t, \;\; u_t \sim N\left(0\,, \sigma^2_u\right),$$

$$\Pr[S_t = j |\; S_{t-1} = \mathrm{i}] = \pi_{ij}, \quad for\; i = 1{,}2\; and\; j = 1{,}2.$$

$$[\delta_1 = -1, \delta_2 = 0.5, \phi = 0.5, \sigma^2_u = 0.04, \pi_{11} = 0.99, \pi_{22} = 0.99]$$

	$\kappa_J = 1 + 2\sum_{j=1}^{J} \rho_j\,,\; J = 2{,}000$	
	PG	PGAS-PR

Model
Parameters
	N = 1000	N = 20
		M = 20

β_0	1969.85	472.94
β_1	2518.01	530.16
β_2	591.28	33.24
$\sigma^2_{\varepsilon_1}$	152.75	3.35
α_0	2850.21	541.86
$\sigma^2_{\varepsilon_2}$	1342.20	14.27
ϕ	139.48	17.09
γ	165.44	8.15
σ^2_u	1699.27	18.58
π_{11}	39.72	2.38
π_{22}	35.58	3.32
Relative Computing Time	1	0.5751

Note:

1. The empirical autocorrelation functions are obtained based on 40,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in. Running time for the PG sampler is 10171.12sec.
2. The average of the inefficient factors from 5 simulations are reported.
3. PG refers to the benchmark PG sampler.
4. PGAS refers to the proposed PG sampler with ancestor sampling.
5. PGAS-PR refers to the proposed PG sampler with ancestor sampling and particle rejuvenation.

Table 2.A Bayesian Estimation of SV Model with Regime-dependent Leverage Effect for S&P 500: 2 State Case [Sample: Jan/02/1975 ~ Aug/05/2015]

$$y_t = \mu + \exp\left(\frac{x_{t-1}}{2}\right)\varepsilon_t,\ \ \varepsilon_t \sim i.i.d.N.(0,1)$$

$$x_t = \delta_{s_t} + \phi\left(x_{t-1} - \delta_{s_{t-1}}\right) + u_t,\ \ u_t \sim i.i.d\ N\ (0, \sigma_u^2)$$

$$\begin{bmatrix}\varepsilon_t \\ u_t\end{bmatrix} \sim N\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}1 & \rho_{s_t}\sigma_u \\ \rho_{s_t}\sigma_u & \sigma_u^2\end{bmatrix}\right)$$

$$\Pr[S_t = j|\ S_{t-1} = \mathrm{i}] = \pi_{ij}, \qquad \sum_{j=1}^{2} \pi_{ij} = 1,\ \ for\ i, j = 1{,}2.$$

Parameters	Prior		Posterior				
	Mean	SD	Mean	Median	SD	90 % HPDI	
π_{11}	0.99	0.01	0.998	0.997	0.001	(0.997	0.999)
π_{22}	0.99	0.01	0.997	0.995	0.001	(0.995	0.999)
μ	0	1	0.036	0.024	0.008	(0.024	0.049)
δ_1	-0.5	0.5	-0.396	-0.444	0.029	(-0.444	-0.350)
δ_2	0	0.5	0.159	0.108	0.031	(0.108	0.210)
ϕ	0	0.5	0.954	0.945	0.005	(0.945	0.962)
ρ_1	0	2	-0.500	-0.593	0.054	(-0.593	-0.407)
ρ_2	0	2	-0.672	-0.740	0.045	(-0.740	-0.590)
σ_u^2	0.01	0.5	0.010	0.008	0.001	(0.008	0.012)

Note:
1. Burn-in / Total iterations = 5,000 / 25,000
2. S.D. refers to the standard deviations of the posterior distributions.
3. A highest posterior density interval (HPDI) is an interval, the narrowest one possible with a chosen probability.

Table 2.B Bayesian Estimation of SV Model with Regime-dependent Leverage Effect for NASDAQ: 2 State Case [Sample: Jan/02/1975 ~ Aug/05/2015]

$$y_t = \mu + \exp\left(\frac{x_{t-1}}{2}\right)\varepsilon_t,\ \ \varepsilon_t \sim i.i.d.N.(0,1)$$

$$x_t = \delta_{s_t} + \phi\left(x_{t-1} - \delta_{s_{t-1}}\right) + u_t,\ \ u_t \sim i.i.d\ N\ (0, \sigma_u^2)$$

$$\begin{bmatrix}\varepsilon_t \\ u_t\end{bmatrix} \sim N\left(\begin{bmatrix}0 \\ 0\end{bmatrix}, \begin{bmatrix}1 & \rho_{s_t}\sigma_u \\ \rho_{s_t}\sigma_u & \sigma_u^2\end{bmatrix}\right)$$

$$\Pr[S_t = j|\ S_{t-1} = \mathrm{i}] = \pi_{ij}, \qquad \sum_{j=1}^{2} \pi_{ij} = 1,\ for\ i,j = 1{,}2.$$

Parameters	Prior		Posterior			
	Mean	SD	Mean	Median	SD	90 % HPDI
π_{11}	0.99	0.01	0.999	0.998	0.001	(0.998 1.000)
π_{22}	0.99	0.01	0.997	0.995	0.001	(0.995 0.999)
μ	0	1	0.089	0.077	0.008	(0.077 0.102)
δ_1	-0.5	0.5	-0.390	-0.443	0.032	(-0.443 -0.337)
δ_2	0	0.5	0.422	0.308	0.063	(0.308 0.517)
ϕ	0	0.5	0.958	0.947	0.007	(0.947 0.968)
ρ_1	0	2	-0.269	-0.329	0.037	(-0.329 -0.206)
ρ_2	0	2	-0.653	-0.719	0.044	(-0.719 -0.576)
σ_u^2	0.01	0.5	0.012	0.010	0.002	(0.010 0.015)

Note: 1. Burn-in / Total iterations = 5,000 / 25,000

2. S.D. refers to the standard deviations of the posterior distributions.

3. A highest posterior density interval (HPDI) is an interval, the narrowest one possible with a chosen probability.

Table 3.A Bayesian Estimation of SV Model with Regime Switching Leverage Effect for S&P 500: 3 State Case [Sample: Jan/02/1975 ~ Aug/05/2015]

$$y_t = \mu + \exp\left(\frac{x_{t-1}}{2}\right)\varepsilon_t,\ \ \varepsilon_t \sim i.i.d.N.(0,1)$$

$$x_t = \delta_{s_t} + \phi\left(x_{t-1} - \delta_{s_{t-1}}\right) + u_t,\ \ u_t \sim i.i.d\ N\ (0, \sigma_u^2)$$

$$\begin{bmatrix} \varepsilon_t \\ u_t \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \rho_{s_t}\sigma_u \\ \rho_{s_t}\sigma_u & \sigma_u^2 \end{bmatrix}\right)$$

$$\Pr[S_t = j|\ S_{t-1} = \mathrm{i}] = \pi_{ij}, \qquad \sum_{j=1}^{3} \pi_{ij} = 1,\ \ for\ i,j = 1,2,3.$$

Parameters	Prior		Posterior			
	Mean	SD	Mean	Median	SD	90 % HPDI
π_{11}	0.98	0.04	0.997	0.996	0.001	(0.996 0.999)
π_{12}	0.01	0.03	0.002	0.001	0.001	(0.001 0.003)
π_{21}	0.01	0.03	0.003	0.001	0.001	(0.001 0.005)
π_{22}	0.98	0.04	0.995	0.993	0.001	(0.993 0.997)
π_{31}	0.01	0.03	0.005	0.001	0.004	(0.001 0.013)
π_{32}	0.01	0.03	0.017	0.007	0.008	(0.007 0.032)
μ	0	1	0.035	0.022	0.008	(0.022 0.047)
δ_1	-0.5	0.5	-0.431	-0.469	0.024	(-0.469 -0.391)
δ_2	0	0.5	0.065	0.002	0.035	(0.002 0.117)
δ_3	0	0.5	0.871	0.698	0.116	(0.698 1.088)
ϕ	0	0.5	0.937	0.924	0.009	(0.924 0.951)
ρ_1	0	2	-0.565	-0.645	0.045	(-0.645 -0.499)
ρ_2	0	2	-0.713	-0.790	0.045	(-0.790 -0.636)
ρ_3	0	2	-0.588	-0.802	0.173	(-0.802 -0.221)
σ_u^2	0.01	0.5	0.010	0.007	0.001	(0.007 0.012)

Note:
1. Burn-in / Total iterations = 5,000 / 25,000
2. S.D. refers to the standard deviations of the posterior distributions.
3. A highest posterior density interval (HPDI) is an interval, the narrowest one possible with a chosen probability.

Table 3.B Bayesian Estimation of SV Model with Regime Switching Leverage Effect for NASDAQ: 3 State Case [Sample: Jan/02/1975 ~ Aug/05/2015]

$$y_t = \mu + \exp\left(\frac{x_{t-1}}{2}\right)\varepsilon_t,\ \ \varepsilon_t \sim i.i.d.N.(0,1)$$

$$x_t = \delta_{s_t} + \phi\left(x_{t-1} - \delta_{s_{t-1}}\right) + u_t,\ \ u_t \sim i.i.d\ N\ (0, \sigma_u^2)$$

$$\begin{bmatrix} \varepsilon_t \\ u_t \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \rho_{s_t}\sigma_u \\ \rho_{s_t}\sigma_u & \sigma_u^2 \end{bmatrix}\right)$$

$$\Pr[S_t = j|\ S_{t-1} = \mathrm{i}] = \pi_{ij}, \qquad \sum\nolimits_{j=1}^{3} \pi_{ij} = 1,\ \ for\ i,j = 1,2,3.$$

Parameters	Prior		Posterior				
	Mean	SD	Mean	Median	SD	90 % HPDI	
π_{11}	0.98	0.04	0.997	0.995	0.001	(0.995	0.999)
π_{12}	0.01	0.03	0.002	0.001	0.001	(0.001	0.004)
π_{21}	0.01	0.03	0.002	0.001	0.001	(0.001	0.004)
π_{22}	0.98	0.04	0.997	0.995	0.001	(0.995	0.998)
π_{31}	0.01	0.03	0.001	0.000	0.001	(0.000	0.003)
π_{32}	0.01	0.03	0.004	0.001	0.002	(0.001	0.006)
μ	0	1	0.088	0.076	0.008	(0.076	0.100)
δ_1	-0.5	0.5	-0.595	-0.654	0.035	(-0.654	-0.538)
δ_2	0	0.5	-0.092	-0.137	0.028	(-0.137	-0.043)
δ_3	0	0.5	0.697	0.634	0.038	(0.634	0.758)
ϕ	0	0.5	0.921	0.909	0.007	(0.909	0.933)
ρ_1	0	2	-0.221	-0.313	0.055	(-0.313	-0.131)
ρ_2	0	2	-0.596	-0.674	0.047	(-0.674	-0.513)
ρ_3	0	2	-0.667	-0.757	0.054	(-0.757	-0.580)
σ_u^2	0.01	0.5	0.016	0.013	0.002	(0.013	0.018)

Note:
1. Burn-in / Total iterations = 5,000 / 25,000
2. S.D. refers to the standard deviations of the posterior distributions.
3. A highest posterior density interval (HPDI) is an interval, the narrowest one possible with a chosen probability.

Table 4. Deviance Information Criterion: Bayesian Model Comparison

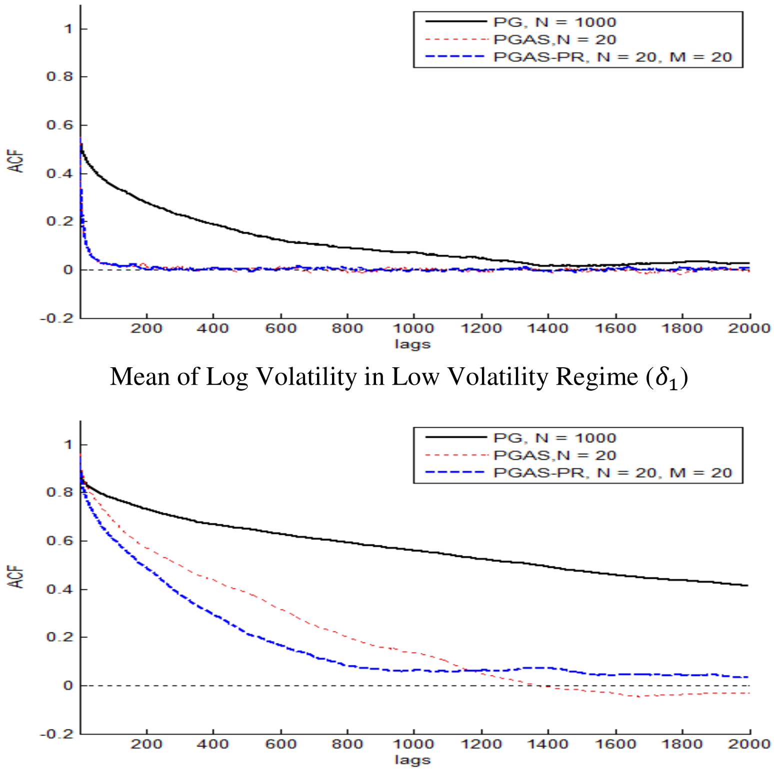
$$DIC = -2E_{\zeta|Y}[\ln f(Y|\zeta)] + 2\{\ln f(Y|\bar{\zeta}) - E_{\zeta|Y}[\ln f(Y|\zeta)]\}$$

	S&P 500		Nasdaq	
Model	DIC	Ranking	DIC	Ranking
1	26,088.95	4	27,015.57	4
2	26,078.41	3	26,992.56	3
3	26,054.90	2	26,977.52	2
4	26,044.53	1	26,947.22	1

Note: 1. Model 1: 2-state Regime Switching SV with Constant Leverage Effect
Model 2: 2-state Regime Switching SV with Regime-dependent Leverage Effect
Model 3: 3-state Regime Switching SV with Constant Leverage Effect
Model 4: 3-state Regime Switching SV with Regime-dependent Leverage Effect

2. Burn-in / Total iterations = 5,000 / 25,000

Figure 1.A. Autocorrelation Functions for Selected Model Parameters: Example 1 [T = 3000]



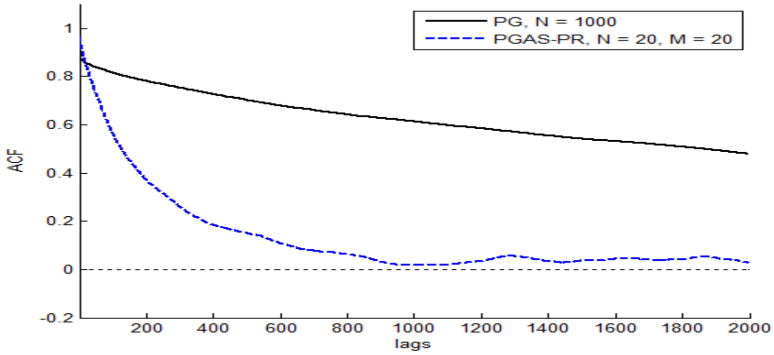
Mean of Log Volatility in Low Volatility Regime (δ_1)

AR Coefficient (ϕ)

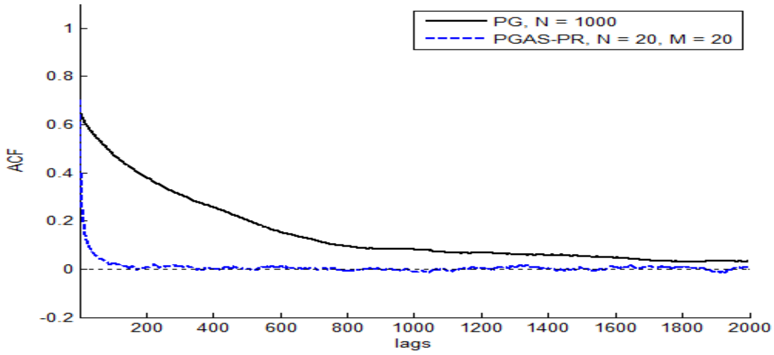
Note:

1. The empirical autocorrelation functions are obtained based on 40,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in.
2. The averages of the ACFs calculated from 5 simulations are reported.
3. PG refers to the benchmark PG sampler. ACF for PG is the black bold line.
4. PGAS refers to the proposed PG sampler with ancestor sampling. ACF for PGAS is the red dotted line.
5. PGAS-PR refers to the proposed PG sampler with ancestor sampling and particle rejuvenation. ACF for PGAS-PR is the blue dashed line.

Figure 1.B. Autocorrelation Functions for Selected Model Parameters: Example 2 [T = 500]



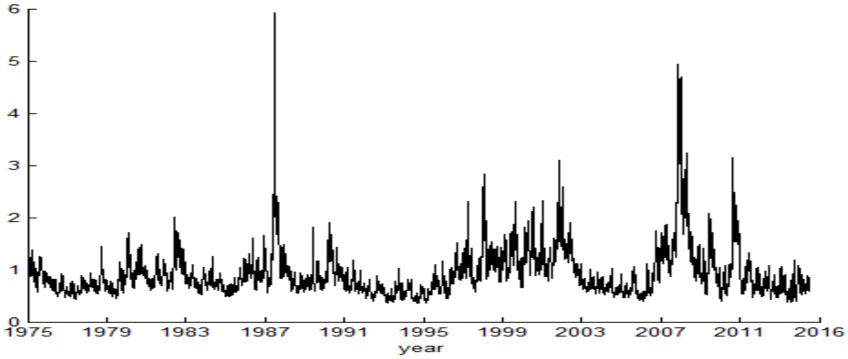
Coefficient for f_t (β_1)



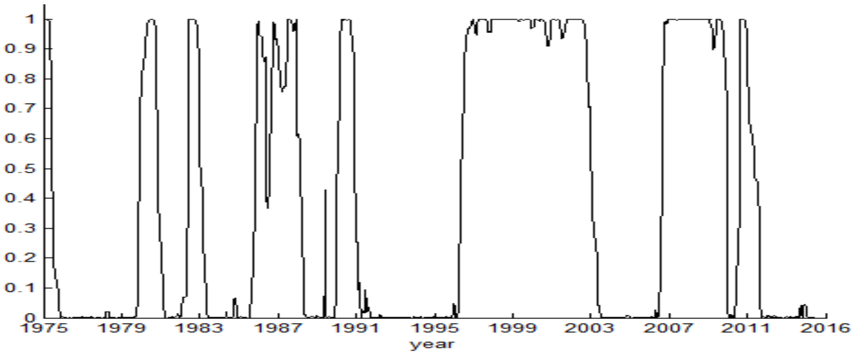
Coefficient for $f_t^{\ 2}$ (β_2)

Note: 1. The empirical autocorrelation functions are obtained based on 40,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in.
2. The averages of the ACFs calculated from 5 simulations are reported.
3. PG refers to the benchmark PG sampler. ACF for PG is the black bold line.
4. PGAS-PR refers to the proposed PG sampler with ancestor sampling and particle rejuvenation. ACF for PGAS-PR is the blue dashed line.

Table 2.A Posterior Estimates of Stochastic Volatility and Regime Probability: 2 State RS-SV with Leverage Effect [S&P 500: Jan/02/1975 ~ Aug/05/2015]



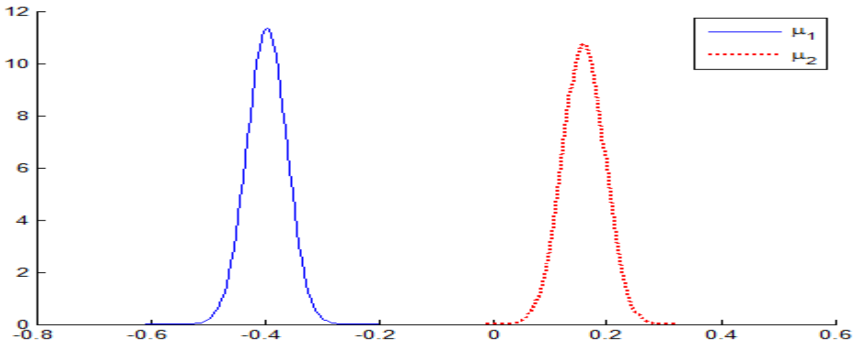
Posterior Mean of Stochastic Volatility



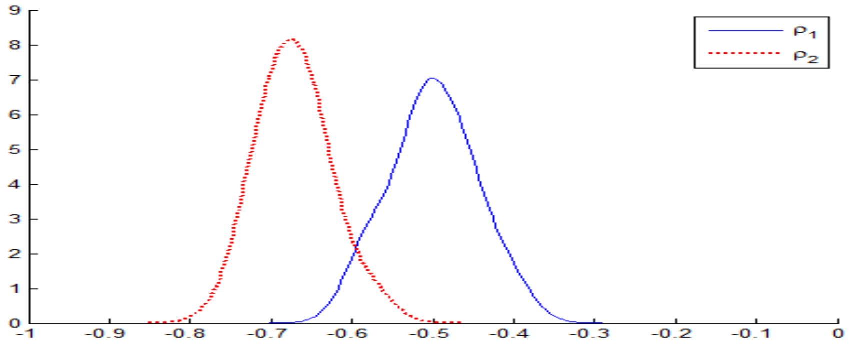
Posterior Probability of High Volatility Regime

Note: 1. The empirical autocorrelation functions are obtained based on 20,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in.
2. N = 20 particles are used in the proposed PGAS sampler.

Table 2.B Posterior Distributions of Regime Switching Parameters: 2 State RS-SV with Leverage Effect [S&P 500: Jan/02/1975 ~ Aug/05/2015]



Posterior Distributions of μ_{s_t}

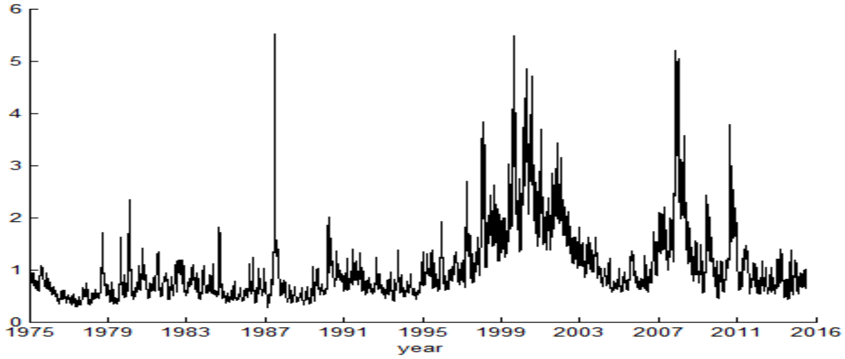


Posterior Distributions of ρ_{s_t}

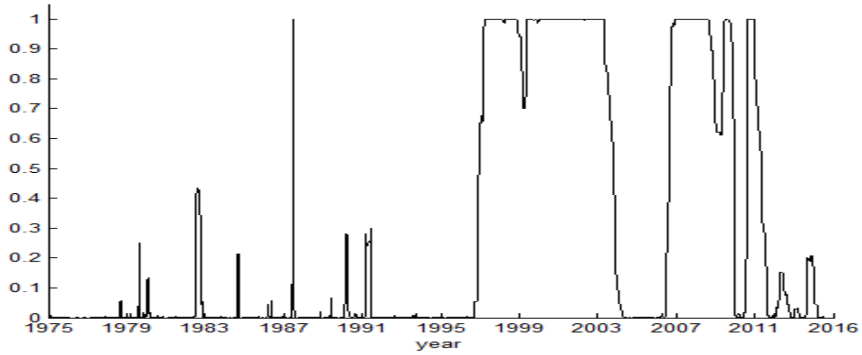
Note: 1. The empirical autocorrelation functions are obtained based on 20,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in.

2. N = 20 particles are used in the proposed PGAS sampler.

Table 3.A Posterior Estimates of Stochastic Volatility and Regime Probability: 2 State RS-SV with Leverage Effect [NASDAQ: Jan/02/1975 ~ Aug/05/2015]



Posterior Mean of Stochastic Volatility

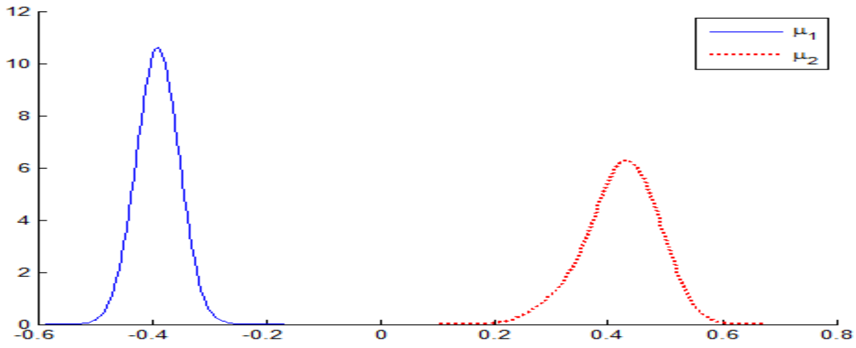


Posterior Probability of High Volatility Regime

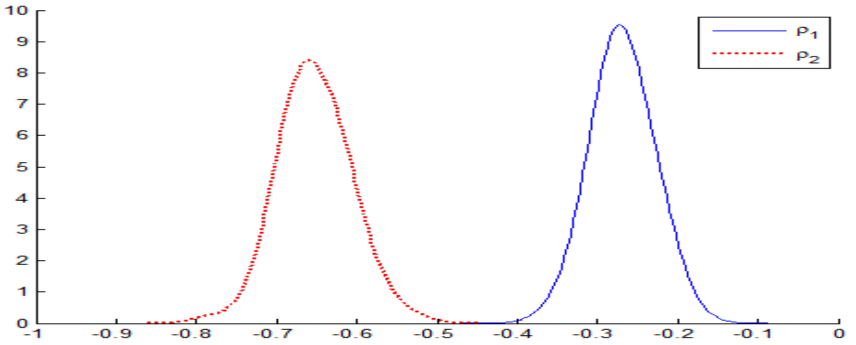
Note:

1. The empirical autocorrelation functions are obtained based on 20,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in.
2. N = 20 particles are used in the proposed PGAS sampler.

Table 3.B Posterior Distributions of Regime Switching Parameters: 2 State RS-SV with Leverage Effect [NASDAQ: Jan/02/1975 ~ Aug/05/2015]



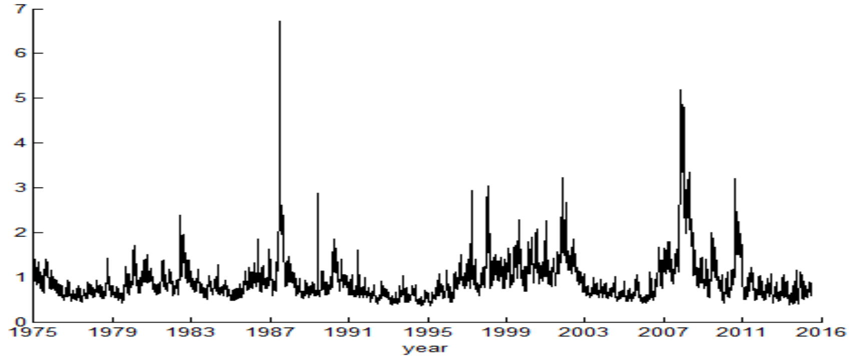
Posterior Distributions of μ_{s_t}



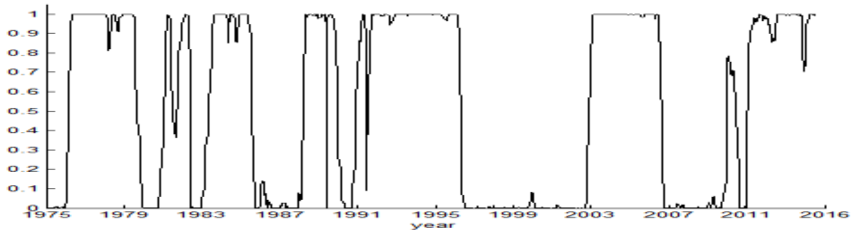
Posterior Distributions of ρ_{s_t}

Note: 1. The empirical autocorrelation functions are obtained based on 20,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in.
2. N = 20 particles are used in the proposed PGAS sampler.

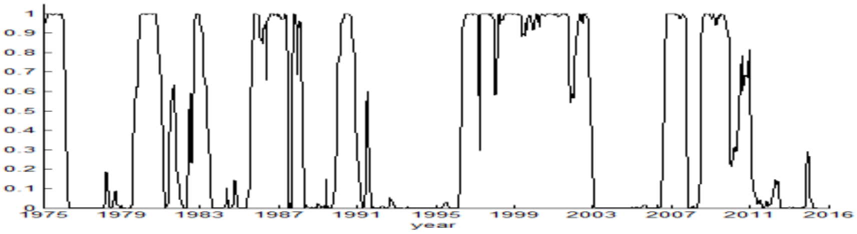
Table 4.A Posterior Estimates of Stochastic Volatility and Regime Probability: 3 State RS-SV with Leverage Effect [S&P 500: Jan/02/1975 ~ Aug/05/2015]



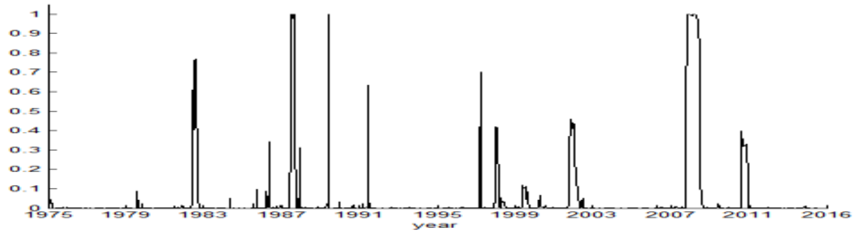
Posterior Mean of Stochastic Volatility



Posterior Probability of Low Volatility Regime



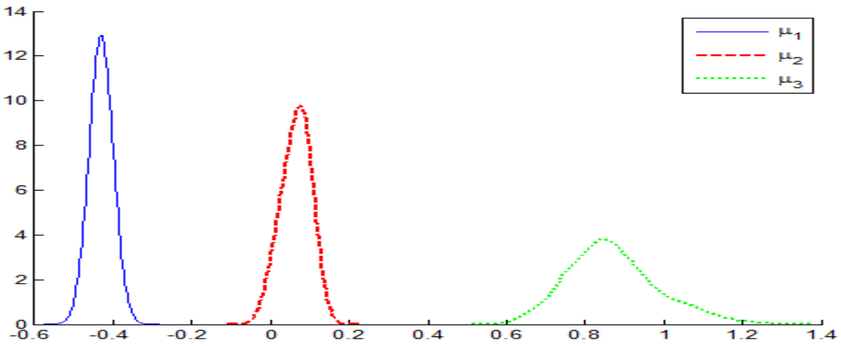
Posterior Probability of Medium Volatility Regime



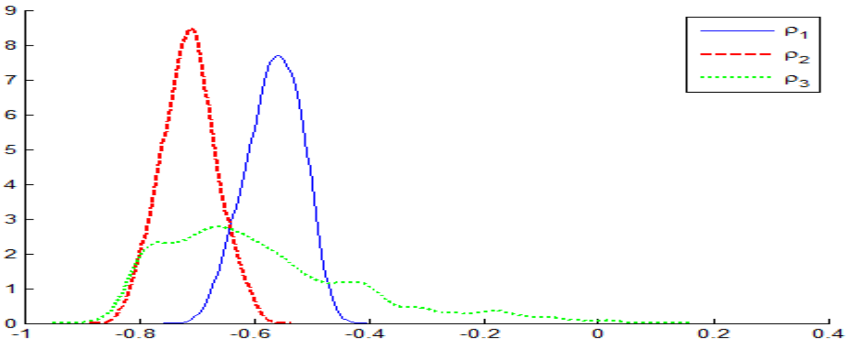
Posterior Probability of High Volatility Regime

Note: 1. The empirical autocorrelation functions are obtained based on 20,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in.
2. N = 20 particles are used in the proposed PGAS sampler.

Table 4.B Posterior Distributions of Regime Switching Parameters: 3 State RS-SV with Leverage Effect [S&P 500: Jan/02/1975 ~ Aug/05/2015]



Posterior Distributions of μ_{s_t}

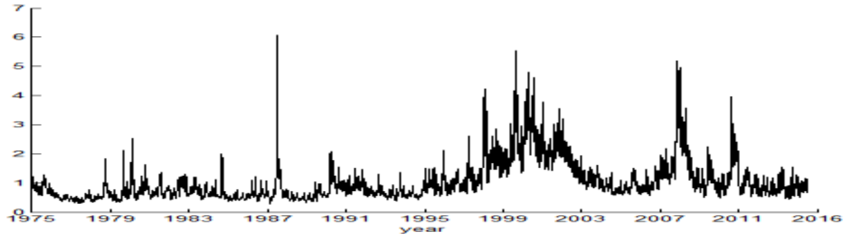


Posterior Distributions of ρ_{s_t}

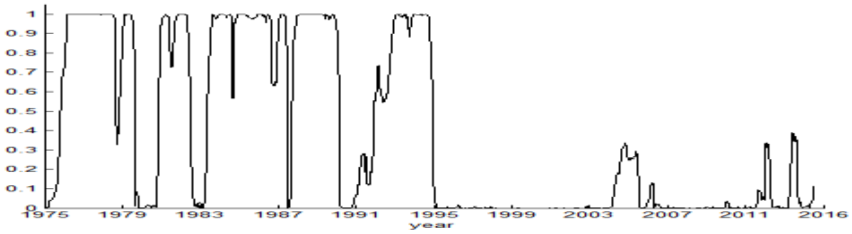
Note: 1. The empirical autocorrelation functions are obtained based on 20,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in.

2. N = 20 particles are used in the proposed PGAS sampler.

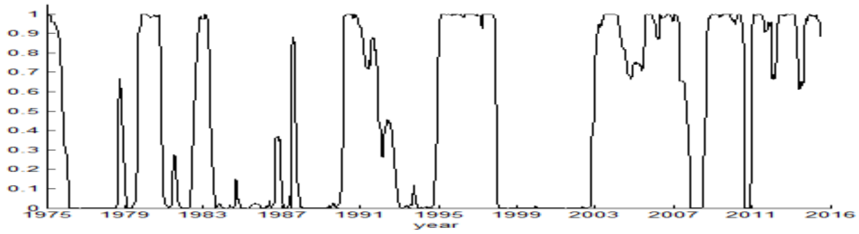
Table 5.A Posterior Estimates of Stochastic Volatility and Regime Probability: 3 State RS-SV with Leverage Effect [NASDAQ: Jan/02/1975 ~ Aug/05/2015]



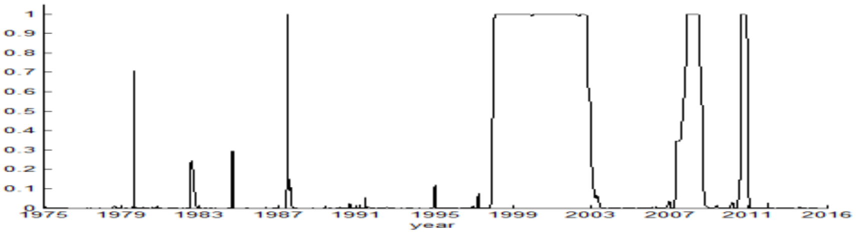
Posterior Mean of Stochastic Volatility



Posterior Probability of Low Volatility Regime



Posterior Probability of Medium Volatility Regime

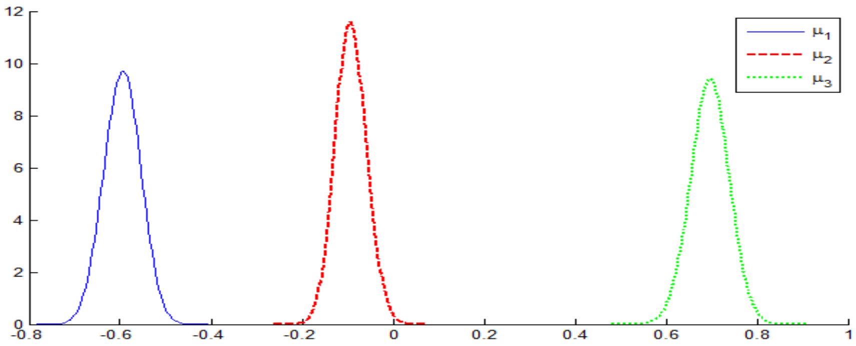


Posterior Probability of High Volatility Regime

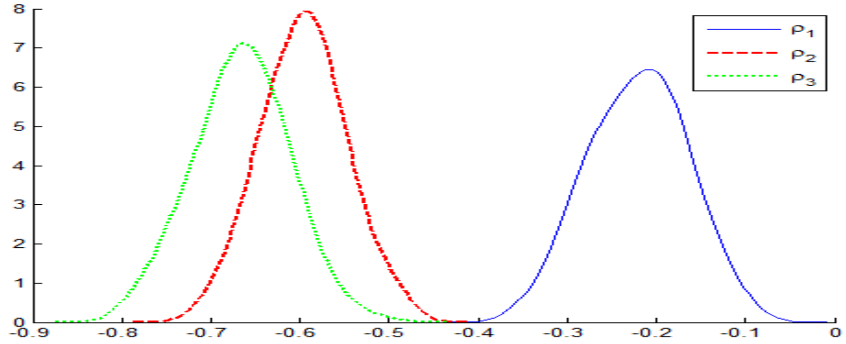
Note: 1. The empirical autocorrelation functions are obtained based on 20,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in.

2. N = 20 particles are used in the proposed PGAS sampler.

Table 5.B Posterior Distributions of Regime Switching Parameters: 3 State RS-SV with Leverage Effect [NASDAQ: Jan/02/1975 ~ Aug/05/2015]



Posterior Distributions of μ_{s_t}



Posterior Distributions of ρ_{s_t}

Note: 1. The empirical autocorrelation functions are obtained based on 20,000 MCMC iterations. The first 3,000 iterations are discarded as the burn-in.

2. N = 20 particles are used in the proposed PGAS sampler.