Bayesian Analysis in Non-linear Non-Gaussian State-Space Models using Particle Gibbs

Oliver Grothe
Institute of Operations Research, Karlsruhe Institute of Technology, Germany
Tore Selland Kleppe
Department of Mathematics and Natural Sciences, University of Stavanger, Norway
Roman Liesenfeld*
Institute of Econometrics and Statistics, University of Cologne, Germany

(January 7, 2016)

Abstract

We consider Particle Gibbs (PG) as a tool for Bayesian analysis of non-linear non-Gaussian state-space models. PG is a Monte Carlo (MC) approximation of the standard Gibbs procedure which uses sequential MC (SMC) importance sampling inside the Gibbs procedure to update the latent and potentially high-dimensional state trajectories. We propose to combine PG with a generic and easily implementable SMC approach known as Particle Efficient Importance Sampling (PEIS). By using SMC importance sampling densities which are closely globally adapted to the targeted density of the states, PEIS can substantially improve the mixing and the efficiency of the PG draws from the posterior of the states and the parameters relative to existing PG implementations. The efficiency gains achieved by PEIS are illustrated in PG applications to a stochastic volatility model for asset returns and a Gaussian nonlinear local level model for interest rates.

JEL classification: C11; C13; C15; C22.

Keywords: Ancestor sampling; Dynamic latent variable models; Efficient importance sampling; Markov chain Monte Carlo; Sequential importance sampling.

*Corresponding address: Institut für Ökonometrie und Statistik, Universität Köln, Universitätsstr. 22a, D-50937 Köln, Germany. Tel.: +49(0)221-470-2813; fax: +49(0)221-470-5074. *E-mail address:* liesenfeld@statistik.uni-koeln.de (R. Liesenfeld)

1. Introduction

In this paper we consider the particle Gibbs procedure (Holenstein, 2009, Andrieu et al., 2010) as a tool to perform a Bayesian analysis of non-linear, non-Gaussian state space models and discuss how to improve its efficiency by relying upon a sequential Monte Carlo procedure known as particle efficient importance sampling (Scharth and Kohn, 2013).

In the context of state space models (SSM), a latent Markov state variable x_t ($t = 1, \ldots, T$) is observed through a response variable y_t, where it is assumed that the y_t's are conditionally independent given the x_t's. The measurement density for y_t and the transition density for x_t, depending on a vector of parameters θ are written as

$$y_t|x_t \sim g_\theta(y_t|x_t) \qquad \text{and} \qquad x_t|x_{t-1} \sim f_\theta(x_t|x_{t-1}), \quad x_1 \sim f_\theta(x_1), \tag{1}$$

respectively.

Bayesian inference about the parameters θ and the states $x_{1:T}$ of the SSM in Equation (1) relies on their joint posterior denoted by $p(\theta, x_{1:T}|y_{1:T})$, where we have used the notation $z_{s:t}$ to denote $(z_s, z_{s+1}, \ldots, z_t)$. The corresponding marginal posterior for the parameters is $p(\theta|y_{1:T}) \propto p_\theta(y_{1:T})p(\theta)$, where $p(\theta)$ denotes the prior assigned to θ and $p_\theta(y_{1:T})$ the marginal likelihood. For non-linear, non-Gaussian models, both the joint posterior of θ and $x_{1:T}$ as well as the marginal posterior for θ are analytically intractable so that inference requires to resort to approximation techniques.

A well established class of approximation methods for Bayesian inference in non-linear, non-Gaussian SSMs are Markov Chain Monte Carlo (MCMC) procedures. A popular MCMC approach to approximate the joint posterior $p(\theta, x_{1:T}|y_{1:T})$ consists of using the Gibbs sampler, which alternately samples from the full conditional posterior of the parameters θ denoted by $p(\theta|x_{1:T}, y_{1:t})$ and the full conditional posterior of the states $x_{1:T}$ written as $p_\theta(x_{1:T}|y_{1:T})$. The problem with this method is that sampling from the density $p_\theta(x_{1:T}|y_{1:T})$ is typically difficult. In fact, for models of practical interest this Gibbs block is often a high-dimensional non-standard density so that sampling needs to rely on a Metropolis-Hastings (MH) algorithm based on a proposal density whose efficient design is a challenging task (see, e.g., Carter and Cohn, 1994, Shephard and Pitt, 1997 and Liesenfeld and

Richard, 2008).

A new and easy to implement tool for approximating the joint posterior $p(\theta, x_{1:T}|y_{1:T})$ in SSMs are the Particle MCMC (PMCMC) algorithms developed by Holenstein (2009) and Andrieu et al. (2010), which combine MCMC with sequential Monte Carlo (SMC) algorithms. The latter are simulation devices for forward-recursively approximating high-dimensional target densities and their integrating constants, such as the conditional posterior $p_\theta(x_{1:T}|y_{1:T})$ and the marginal likelihood $p_\theta(y_{1:T})$ in an SSM. More specifically, SMC methods generate a swarm of $x_{1:t}$-samples (particles), that evolve towards the target distribution according to a combination of sequentially importance sampling (IS) and resampling. Standard SMC implementations rely upon locally designed IS densities approximating the corresponding subcomponents of the full target density (see, e.g., Gordon et al., 1993, Pitt and Shephard, 1999, and Doucet and Johansen, 2009). Within the PMCMC approach such SMC algorithms are used in order to design high-dimensional proposal densities for MH updates producing MCMC draws from the respective target density.

For a direct application to a full Bayesian analysis in SSMs two PMCMC algorithms are available: The particle marginal MH (PMMH) and the particle Gibbs (PG). The PMMH algorithm represents an MC approximation of an 'ideal' (but infeasible) MH procedure targeting directly the marginal posterior density $p(\theta|y_{1:T})$ and marginalizes the states $x_{1:T}$ by using SMC to obtain an unbiased MC estimate of the marginal likelihood $p_\theta(y_{1:T})$. Applications of PMMH for Bayesian inference in SSMs are found in Fernandez-Villaverde and Rubio-Ramirez (2005), Flury and Shephard (2011), Scharth and Kohn (2013), and Pitt et al. (2012). However, a potential drawback of this approach in practical applications is, that the design of a proposal density for the MH updates of θ can be tedious, requiring a fair amount of fine tuning, especially, when the number of parameters in θ are large. Moreover, PMMH can be 'computationally brutal' if the SMC delivers, even with a large number of particles, noisy MC estimates for $p_\theta(y_{1:T})$, which is to be expected for standard locally designed SMCs in high-dimensional applications (Flury and Shephard, 2010). As a result, the MH updates for θ can get stuck for many iterations leading to very slow mixing. In order to address this problem of PMMH, Scharth and Kohn (2013) recently developed the Particle Efficient IS (PEIS) which combines SMC with the sequential Efficient IS (EIS) procedure of Richard and Zhang (2007).

This approach exploits that EIS produces by a sequence of auxiliary regressions a close global density approximation to a potentially high-dimensional target density and, thus, minimizes the noise in the corresponding estimate for the likelihood $p_\theta(y_{1:T})$.

Here we consider the PG approach as an alternative to PMMH to make inference in SSMs. The PG is an MC approximation of an 'ideal' Gibbs algorithm iterating between $p(\theta|x_{1:T}, y_{1:t})$ and $p_\theta(x_{1:T}|y_{1:T})$, where the output of an SMC algorithm targeting $p_\theta(x_{1:T}|y_{1:T})$ is used as a proposal distribution for MH updates of $x_{1:T}$. It can take advantage of the fact that in numerous application sampling from $p(\theta|x_{1:T}, y_{1:t})$ is easily feasible so that the tedious design of a proposal for θ as required by PMMH can be bypassed. A further potential advantage of the PG approach relative to PMMH is that it does not need to MH update $x_{1:T}$ in one block so that SMC sampling from $p_\theta(x_{1:T}|y_{1:t})$ can be partitioned into a sequence of smaller sampling problems. This can be a partitioning into blocks along the time dimension and/or into state components for a multivariate state vector x_t.

However, just as much as the PMMH, the PG can suffer from poor mixing, though for a different reason. Since the resampling of typical SMC procedures may lead to potentially identical genealogies of the $x_{1:T}$-particle paths, the exploration of the domain of $x_{1:s}$ under $p_\theta(x_{1:T}|y_{1:T})$ for $s \ll T$ may be very poor for the PG (Whiteley et al., 2010 and Lindsten and Schön, 2012).

Existing attempts to address this poor mixing problem of the PG are to either add a Backward Simulation step (PGBS) (Whiteley, 2010, Whiteley et al., 2010, Lindsten and Schön, 2012, and Carter et al., 2014) or an Ancestor Sampling step (PGAS) (Lindsten and Schön, 2014) to the SMC algorithm, or to introduce an additional MH move to update $x_{1:T}$ (Holenstein, 2009, p. 35). However, as we shall demonstrate, the efficacy of those extensions to improve the mixing of the baseline PG critically depends on how close the underlying SMC algorithm approximates the target $p_\theta(x_{1:T}|y_{1:T})$. As mentioned above, the globally designed PEIS of Scharth and Kohn (2013) provides an SMC which produces such very close approximations. Therefore, we should be able to improve the mixing of the baseline PG and its extensions by relying upon the PEIS for their applications. A striking illustration of how the PEIS improves this mixing is provided in Section 5, where we apply PG algorithms to a Bayesian analysis of a stochastic volatility model (SV) for asset returns and a time-discretized Constant Elasticity of Variance (CEV) diffusion with measurement errors for interest rates.

The rest of the paper is organized as follows: In Section 2 we briefly outline the SMC approach and in Section 3 the basline PG. Section 4 presents the PEIS (Section 4.1) and discusses potential efficiency improvements obtained by embedding the PEIS within the PGAS (Section 4.2) and the PGMH (Section 4.3). This is illustrated in Section 5 with a Bayesian PG analysis of the SV model and a CEV model. Section 6 concludes.

2. Sequential Monte Carlo (SMC)

2.1 Definition of SMC

Let $\pi(x_{1:T})$ denote the target density to be approximated/simulated with the following sequence of intermediate target densities:

$$\pi_t(x_{1:t}) = \frac{\gamma_t(x_{1:t})}{z_t}, \qquad z_t = \int \gamma_t(x_{1:t}) dx_{1:t}, \qquad t = 1, \ldots, T, \tag{2}$$

with $\pi_T(x_{1:T}) \equiv \pi(x_{1:T})$.

In an SSM of the form given by Equation (1) the full target is $\pi(x_{1:T}) = p_\theta(x_{1:T}|y_{1:T})$ and for standard SMC algorithms the intermediate targets are defined as $\pi_t(x_{1:t}) \equiv p_\theta(x_{1:t}|y_{1:t})$, so that we have

$$\gamma_t(x_{1:t}) = p_\theta(x_{1:t}, y_{1:t}) = \left[\prod_{\tau=2}^{t} g_\theta(y_\tau|x_\tau) f_\theta(x_\tau|x_{\tau-1})\right] g_\theta(y_1|x_1) f_\theta(x_1), \tag{3}$$

$$z_t = p_\theta(y_{1:t}) = \int p_\theta(x_{1:t}, y_{1:t}) dx_{1:t}, \tag{4}$$

where the sequence $z_t = p_\theta(y_{1:t})$ represent the marginal likelihoods.

SMC algorithms as discussed, e.g., in Cappé et al. (2007), Ristic et al. (2004) and Doucet and Johansen (2009), consist of recursively producing, for each period t, a weighted particle system $\{x^i_{1:t}, w^i_t\}_{i=1}^N$ with N particles $x^i_{1:t}$ and corresponding (non-normalized) IS weights w^i_t such that the intermediate target density $\pi_t(x_{1:t})$ in Equation (2) can be approximated by the point mass distri-

bution

$$\hat{\pi}_t(dx_{1:t}) = \sum_{i=1}^{N} W_t^i \, \delta_{x_{1:t}^i}(dx_{1:t}), \qquad W_t^i = \frac{w_t^i}{\sum_{l=1}^{N} w_t^l}, \tag{5}$$

where $\delta_x(\cdot)$ denotes the Dirac delta mass located at x. In period t, the weighted particle system $\{x_{1:t}^i, w_t^i\}_{i=1}^N$ is obtained from the period-$(t-1)$ system $\{x_{1:t-1}^i, w_{t-1}^i\}_{i=1}^N$ by drawing from an IS-density $q_t(x_t|x_{1:t-1}^i)$ to propagate the inherited particles $x_{1:t-1}^i$ to $x_{1:t}^i = (x_t^i, x_{1:t-1}^i)$ and updating the corresponding IS weights according to

$$w_t^i = W_{t-1}^i \frac{\gamma_t(x_{1:t}^i)}{\gamma_{t-1}(x_{1:t-1}^i) q_t(x_t^i|x_{1:t-1}^i)}. \tag{6}$$

In most applications, the variance of the IS weights w_t^i in Equation (6) increases exponentially with t reducing the effective sample size of the particle system (an effect known as 'weight degeneracy'). Hence, SMC algorithms include a resampling step before propagating the particles $x_{1:t-1}^i$ to $(x_t^i, x_{1:t-1}^i)$. It consists in sampling N 'ancestors particles' from $\{x_{1:t-1}^i\}_{i=1}^N$ according to their normalized IS weights $\{W_{t-1}^i\}$ and then setting in Equation (6) the IS weights W_{t-1}^i for the redrawn $x_{1:t-1}^i$-particles all equal to $1/N$. This resampling step amounts to sampling for $t = 2, ..., T$ the (auxiliary) indices of the ancestor particles $x_{1:t-1}^i$ denoted by a_t^i. For a discussion of popular resampling schemes including multinomial, residual and stratified resampling, see, e.g., Doucet and Johansen (2009).

At period T, this procedure provides us with an approximation of the full target density $\pi(x_{1:T})$ given by $\hat{\pi}_T(dx_{1:T})$ according to Equation (5). Approximate samples from $\pi(x_{1:T})$ can be obtained by sampling $x_{1:T}^i \sim \hat{\pi}_T(dx_{1:T})$, which is done by choosing particles $x_{1:T}^i$ according to their probabilities W_T^i. If required, the corresponding normalizing constant z_T of $\pi(x_{1:T})$ is estimated by

$$\hat{z}_T = \prod_{t=1}^{T} \left(\sum_{i=1}^{N} w_t^i \right). \tag{7}$$

In the SSM context, such an SMC produces an approximation of $\pi(x_{1:T}) = p_\theta(x_{1:T}|y_{1:T})$, denoted by $\hat{p}_\theta(x_{1:T}|y_{1:T})$, corresponding approximate samples $x_{1:T}^i \sim \hat{p}_\theta(x_{1:T}|y_{1:T})$, and an MC approximation to the full marginal likelihood $z_T = p_\theta(y_{1:T})$ written as $\hat{p}_\theta(y_{1:T})$. These are the main inputs of PG

algorithms implemented for Bayesian analyzes of SSMs.

2.2 SMC implementations in state space models

A critical issue in implementing an SMC is the choice of the IS densities $q_t(x_t|x^i_{1:t})$. The main recommendation is to design them *locally* so as to minimize the conditional variance of the IS weights in Equation (6) given $x^i_{1:t-1}$. This requires to select $q_t(x_t|x^i_{1:t-1})$ as a close approximation to the period-t conditional density $\pi_t(x_t|x^i_{1:t-1})$ (see Doucet and Johansen, 2009). For the SSM applications with $\pi_t(x_{1:t}) \propto p_\theta(x_{1:t}, y_{1:t})$ as given by Equation (3), those IS weights become

$$w^i_t = W^i_{t-1} \frac{g_\theta(y_t|x^i_t) f_\theta(x^i_t|x^i_{t-1})}{q_t(x^i_t|x^i_{1:t-1})}. \tag{8}$$

The most popular (but suboptimal) selection for the IS densities are the transition densities $f_\theta(x_t|x^i_{t-1})$ used by the Bootstrap Particle Filter (BPF) (Gordon et al., 1993). In scenarios where the measurement density g_θ is fairly flat in x_t, this selection typically leads to a satisfactory performance. A selection which sets the variance of the IS weights in Equation (8) *conditional on* x^i_{t-1} to zero is $p_\theta(x_t|y_t, x^i_{t-1}) \propto g_\theta(y_t|x_t) f_\theta(x_t|x^i_{t-1})$, leading to the conditionally Optimal Particle Filter (OPF) discussed, e.g., in Doucet and Johansen (2009). Further improvements can be achieved by replacing the standard resampling schemes based on the IS weights in Equation (8) by more sophisticated ones which favor ancestor particles which will be in regions with high probability mass after their propagation. This is implemented by the Auxiliary Particle Filter (APF) (Pitt and Shephard, 1999).

In contrast to those locally designed SMCs, the PEIS of Scharth and Kohn (2013) uses (nearly) *globally* optimal SMC-IS densities and resampling weights obtained from a close approximation to the full target $\pi(x_{1:T}) = p_\theta(x_{1:T}|y_{1:T})$. This will be explained in greater detail in Section 4.1 below.

Irrespectively of the particular IS density selected to implement an SMC, the resampling steps used to mitigate the weight degeneracy, typically lead to a loss of diversity among the particles as the resultant sample may contain many repeated points. Hence, in many SMC applications resampling is performed dynamically, i.e., only when the weight degeneracy exceeds a certain threshold (see, e.g., Doucet and Johansen, 2009).

3. Particle Gibbs (PG)

3.1 Baseline Particle Gibbs algorithm

For a Bayesian analysis in a non-linear, non-Gaussian SSM the 'ideal' Gibbs sampler targeting the joint posterior $p(\theta, x_{1:T}|y_{1:T})$ and alternately sampling from the full conditional posteriors $p_\theta(x_{1:T}|y_{1:T})$ and $p(\theta|x_{1:T}, y_{1:T})$ is typically unfeasible since exact sampling from $p_\theta(x_{1:T}|y_{1:T})$ is impossible. The PG approach of Holenstein (2009) and Andrieu et al. (2010) uses an SMC algorithm targeting $\pi(x_{1:T}) = p_\theta(x_{1:T}|y_{1:T})$ in order to propose approximate samples from this distribution in such a way that the ideal Gibbs sampler is 'exactly approximated'. This is achieved by augmenting the target density of the ideal Gibbs sampler $p(\theta, x_{1:T}|y_{1:T})$ to include *all* the random variables which are produced by the SMC in order to generate a proposal for $x_{1:T}$ (i.e., the set of all the particle paths $\{x^i_{1:T}\}$, the set of all ancestor indices for the resampling steps $\{a^i_t\}$, and the particle index to be drawn in order to select a particle path from $\{x^i_{1:T}\}$ as a proposal). The PG then obtains as a standard Gibbs sampler for this augmented target density, which is implicitly defined by the joint posterior $p(\theta, x_{1:T}|y_{1:T})$ together with the joint sampling density for all the SMC random variables.

The Gibbs sampler for this augmented target density requires a special type of SMC algorithm, referred to as *conditional* SMC, where one of the particles $\{x^i_{1:T}\}_{i=1}^N$ is specified a-priori. This pre-specified reference particle denoted by $x'_{1:T}$ is then retained throughout the entire SMC sampling process. To accomplish this, one can set $x^1_t \equiv x'_t$ and $a^1_t \equiv 1$ for all periods and use the SMC to sample the x^i_t's and a^i_t's only for $i = 2, ..., N$. This produces a set of N particles and IS weights $\{x^i_{1:T}, w^i_T\}_{i=1}^N$, where the first particle coincides with the pre-specified one, i.e., $x^1_{1:T} = x'_{1:T}$ (see, e.g., Lindsten et al., 2014, Chopin and Singh, 2013).

Based on such a conditional SMC the PG algorithm for an SSM is given by:

PG algorithm

(i) *Initialization* ($j = 0$)*: Set randomly* $\theta^{(0)}$*, run an SMC targeting* $p_{\theta^{(0)}}(x_{1:T}|y_{1:T})$*, and sample* $x^{(0)}_{1:T} \sim \hat{p}_{\theta^{(0)}}(x_{1:T}|y_{1:T})$.

(ii) *For iteration* $j \geq 1$*:*

- *sample* $\theta^{(j)} \sim p(\theta|x^{(j-1)}_{1:T}, y_{1:T})$,

- *run a conditional SMC targeting* $p_{\theta^{(j)}}(x_{1:T}|y_{1:T})$ *conditional on* $x_{1:T}^{(j-1)}$*, and sample* $x_{1:T}^{(j)} \sim \hat{p}_{\theta^{(j)}}(x_{1:T}|y_{1:T})$.

Under weak regularity conditions (Andrieu et al., 2010, Theorem 5) the Markov kernel defined by this PG algorithm leaves the exact target density $p(\theta, x_{1:T}|y_{1:T})$ invariant and delivers a sequence of Gibbs draws $\{\theta^{(j)}, x_{1:T}^{(j)}\}_j$ whose marginal distribution converges for any $N > 1$ to $p(\theta, x_{1:T}|y_{1:T})$ as $j \to \infty$.

Existing applications of the PG use locally designed SMC algorithms like the BPF with resampling steps which are performed at every time period t. Dynamic resampling, while in principle possible, is difficult to implement and computationally inefficient since the conditional SMC at the PG iteration step j requires simulating a set of $N-1$ particles not only consistent with the retained path $x'_{1:T} = x_{1:T}^{(j-1)}$ but also with the resampling times of the SMC pass which has produced the retained path (see Holenstein, 2009, Section, 3.4.1).

3.2 Particle Gibbs and the SMC-path degeneracy

The baseline PG will, if implemented using SMCs with resampling steps at every period t, have a very poor mixing, especially, when T is large (see, Whiteley et al., 2010 and Lindsten and Schön, 2012). The reason for this is that the SMC resampling, which is used to mitigate the weight degeneracy, inevitable leads to a path degeneracy of the SMC particle system (see, e.g., Doucet and Johansen, 2009). This means that every period-t resampling step will sequentially reduce for a fixed $s < t$ and increasing t the number of unique particle values representing $x_{1:s}$, which progressively reduces the quality of the SMC samples for the path $x_{1:t}$ under $\pi_t(x_{1:t}) = p_\theta(x_{1:t}|y_{1:t})$. The consequence of this SMC path degeneracy for the PG is that at iteration step j the new trajectory $x_{1:T}^{(j)}$ tend to coalesce (for $t : T \to 1$) with the previous one $x_{1:T}^{(j-1)}$ which is retained as the reference particle $x'_{1:T}$ throughout conditional SMC sampling. Thus, the resulting particle system degenerates towards this 'frozen' path, leading to a highly dependent Markov chain.

Before we discuss in the next section solutions to this problem of the baseline PG, we emphasize two important points. First, it is not the SMC path degeneracy *per se* which leads to the poor mixing of the PG, but the degeneration of the particle system towards the retained conditional SMC

reference particle $x'_{1:T}$. On the other hand, however, SMC implementations addressing successfully the path degeneracy problem can be used to fight the poor mixing of the PG. Second, by construction *any* SMC, whether implemented using locally or globally optimal IS densities, will lead to a fast degeneration of the SMC paths, when resampling is performed every period. This precludes that the mixing problem of the baseline PG resulting from the path degeneracy can be successfully addressed solely by the design of the SMC IS densities and resampling schemes.

4. Extensions of the baseline Particle Gibbs

In order to address the mixing problem of the baseline PG caused by the SMC path degeneracy the following strategies have been proposed: The first one is to augment the baseline PG by an additional particle MH update step (PGMH) proposing at each PG-iteration step j a completely new SMC path for $x_{1:T}$ (Holenstein, 2009, Section 3.2.3). The second alternative is to add additional Ancestor Sampling (AS) steps to the conditional SMC (PGAS), which assign at each time-period t a new artificial $x_{1:t-1}$-history to the partial frozen path $x'_{t:T}$ (Lindsten et al., 2014). A third strategy is to add to the conditional SMC a backward simulation step (PGBS) based on the output of the SMC forward filtering pass (Whiteley, 2010, Whiteley et al., 2010, and Lindsten and Schön, 2012). However, as discussed in Lindsten et al. (2014) this approach is in Markovian SSMs probabilistically equivalent to the PG with ancestor sampling[1].

As illustrated in our applications below the efficacy of the PGAS and PGMH to improve the mixing of the baseline PG critically depends on the SMC algorithm which is used for their implementation. In particular, an efficient PGMH implementation requires for the additional MH step numerically very precise SMC estimates of the marginal likelihood $p_\theta(y_{1:T})$, which can in high-dimensional applications be too much of a challenge for locally designed SMCs. On the other hand, the efficacy of the PGAS's ancestor sampling to improve the mixing can be seriously hampered by a large variance of the IS weights w_t^i, which is to be expected for local SMCs, especially, in SSM applications with a high signal to noise ratio, i.e., very informative observations coupled with a diffuse prior for the states. Since, as

[1]Recently, Carter et al. (2014) have extended the PGBS approach by adding in the backward simulation pass at each time period an extra MH step to generate new state values.

mentioned above and further detailed below, the PEIS of Scharth and Kohn (2013) uses IS densities which globally minimize across all periods the variance of the IS weights producing a very close SMC approximation to $p_\theta(x_{1:T}|y_{1:T})$ and $p_\theta(y_{1:T})$, we propose to use this PEIS in order to improve the efficiency of the PGAS and PGMH.

Moreover, the reduction of the SMC-weight degeneracy to a (close to) minimum level achieved by the PEIS, also offers the possibility to substantially reduce the SMC path degeneracy by performing the resampling step not at every but only at a few predetermined time periods (say every 500 periods). Hence, the baseline PG implemented by using the PEIS with such a sparse resampling frequency provides by itself a natural further alternative to the PGAS and PGMH in order to address the PG-mixing problem.

The extensions of the baseline PG outlined above are detailed in the next sections: In Section 4.1 we describe the PEIS. In Sections 4.2 and 4.3 we present the PGAS and PGMH, respectively, and discuss the potential efficiency improvements obtained if they are implemented with the PEIS.

4.1 Particle EIS (PEIS)

The PEIS as proposed by Scharth and Kohn (2013) is a 'forward-looking' SMC which uses the sequential EIS procedure of Richard and Zhang (2007) to design both IS densities and a resampling scheme. EIS is a generic algorithm which sequentially constructs a global IS density q for $x_{1:T}$ which provides a close approximation to $p_\theta(x_{1:T}|y_{1:T}) \propto p_\theta(x_{1:T}, y_{1:T})$. This global IS density is factorized conformably with $p_\theta(x_{1:T}, y_{1:T})$ in Equation (3) into

$$q(x_{1:T}; c) = \left[\prod_{t=2}^{T} q_t(x_t|x_{1:t-1}; c_t)\right] q_1(x_1; c_1), \tag{9}$$

with

$$q_t(x_t|x_{1:t-1}; c_t) = \frac{k_t(x_{1:t}; c_t)}{\chi_t(x_{1:t-1}; c_t)}, \qquad \chi_t(x_{1:t-1}; c_t) = \int k_t(x_{1:t}; c_t)dx_t, \tag{10}$$

where $\{k_t(\cdot; c_t), c_t \in \mathcal{C}_t\}$ represents a preselected class of parametric density kernels indexed by a vector of (auxiliary) parameters c_t and with known integrating factors given by χ_t. For any given

$c = (c_1, ..., c_T)$ the global IS ratio $p_\theta(x_{1:T}, y_{1:T})/q(x_{1:T}; c)$ can be factorized so as to obtain

$$\frac{p_\theta(x_{1:T}, y_{1:T})}{q(x_{1:T}; c)} = \chi_1(c_1) \prod_{t=1}^{T} \left[\frac{g_\theta(y_t|x_t) f_\theta(x_t|x_{t-1}) \chi_{t+1}(x_{1:t}; c_{t+1})}{k_t(x_{1:t}; c_t)} \right], \qquad \chi_{T+1}(\cdot) \equiv 1. \tag{11}$$

In order to construct IS densities which provide a close approximation to $p_\theta(x_{1:T}, y_{1:T})$, EIS aims at selecting a value of c that minimizes the variance of this global IS ratio by minimizing period by period the variance of the individual IS ratios given in Equation (11) by the terms in brackets.

A (near) optimal value $\hat{c}$ is obtained by solving the following back-recursive sequence of least squares (LS) approximation problems:

$$(\hat{c}_t, \hat{\alpha}_t) = \arg \min_{c_t \in \mathcal{C}_t, \alpha_t \in \mathbb{R}} \sum_{i=1}^{R} \left\{ \ln \left[g_\theta(y_t|x_t^i) f_\theta(x_t^i|x_{t-1}^i) \chi_{t+1}(x_{1:t}^i; \hat{c}_{t+1}) \right] \right. \tag{12}$$

$$\left. - \alpha_t - \ln k_t(x_{1:t}^i; c_t) \right\}^2, \qquad t = T, \ldots, 1,$$

where α_t represents an intercept, and $\{x_{1:T}^i\}_{i=1}^R$ denote R independent trajectories drawn from $q(x_{1:T}; c)$ itself. Thus, $\hat{c}$ results as a fixed point solution to the sequence $\{\hat{c}^{[0]}, \hat{c}^{[1]}, \ldots\}$ in which $\hat{c}^{[\ell]}$ is obtained from (12) under trajectories drawn from $q(\cdot; \hat{c}^{[\ell-1]})$. In order to ensure convergence to a fixed-point solution it is critical that all $x_{1:T}$ draws generated for the sequence $\{\hat{c}^{[\ell]}\}$ be produced by using a single set of canonical random numbers $\{u_{1:T}^i\}_{i=1}^R$. Note that the $\hat{c}_t$'s are implicit functions of θ, so that maximal efficiency requires complete reruns of the EIS regressions for any new value of θ.

The selection of the parametric class of kernels k_t is inherently problem-specific since these kernels are meant to provide a functional approximation to the product $g_\theta(y_t|x_t) f_\theta(x_t|x_{t-1}) \chi_{t+1}(x_{1:t}; c_{t+1})$. In the applications below, we consider SSMs with Gaussian transition densities f_θ, which suggest to select the k_t's as Gaussian kernels. In this case the EIS LS problems (12) take the form of simple *linear* LS problems. However, it is important to note that EIS is by no means restricted to the use of Gaussian IS samplers. The EIS LS problems become linear for all density kernels k_t chosen within the exponential family of densities and (P)EIS implementations for more flexible IS densities such as mixture of normal distributions are found in Kleppe and Liesenfeld (2014) and Scharth and Kohn

(2013).

The PEIS is an SMC, which is constructed from the output of this EIS algorithm as follows: Firstly, it makes use of the APF principle (Pitt and Shephard, 1999) and replaces the standard resampling scheme based upon the IS weights in Equation (8) by a scheme, which favors particles that are more likely to survive the next resampling steps. As discussed in Doucet and Johansen (2009, Section 4.2), this can be implemented within a standard SMC as outlined in Section 2.1, by replacing the natural intermediate targets $\pi_t(x_{1:t})$ in Equation (3) by auxiliary targets, which include information of future y_t-measurements. The particular auxiliary targets used by the PEIS are given by

$$\pi_t(x_{1:t}) \propto \gamma_t(x_{1:t}) \equiv p_\theta(x_{1:t}, y_{1:t})\chi_{t+1}(x_{1:t}; \hat{c}_{t+1}), \qquad \chi_{T+1}(\cdot) = 1. \tag{13}$$

Secondly, the SMC-IS densities used by the PEIS are the densities obtained from the EIS auxiliary regressions (12),

$$q_t(x_t|x_{1:t-1}) \equiv q_t(x_t|x_{1:t-1}; \hat{c}_t) = \frac{k_t(x_{1:t}; \hat{c}_t)}{\chi_t(x_{1:t-1}; \hat{c}_t)}. \tag{14}$$

The fundamental justification of this PEIS arises from the form of the resulting SMC-IS weights according to Equation (6) together with the specific interpretation of the EIS integrating factor $\chi_{t+1}(x_{1:t}; \hat{c}_{t+1})$ used to define the auxiliary intermediate SMC targets in Equation (13). Both are provided in the following lemma (for the proof see Appendix 1):

Lemma 1. *For the PEIS defined by Equations (13) and (14), the SMC-IS weights in Equation (6) are*

$$w_t^i = W_{t-1}^i \frac{g_\theta(y_t|x_t^i) f_\theta(x_t^i|x_{t-1}^i)\chi_{t+1}(x_{1:t}^i; \hat{c}_{t+1})}{k_t(x_{1:t}^i; \hat{c}_t)}, \tag{15}$$

where $\chi_{t+1}(\cdot; \hat{c}_{t+1})$ is close to be proportional to the multiperiod-a-head predictive density $p_\theta(y_{t+1:T}|x_t)$ for $y_{t+1:T}$ given x_t:

$$\chi_{t+1}(x_{1:t}; \hat{c}_{t+1}) \simeq \textit{constant} \cdot p_\theta(y_{t+1:T}|x_t). \tag{16}$$

Thus, according to Equation (15) the SMC-IS weights w_t^i of the PEIS are the weights, whose variance is minimized by the auxiliary EIS regressions (12), so that PEIS minimizes the SMC weight degeneracy across all periods. Moreover, Equation (16) implies that the intermediate SMC targets of

the PEIS in Equation (13) include a prediction about which particles will be for periods $t+1, ..., T$ in regions with high probability masses. Thus, the resulting resampling scheme based on the weights (15) favors ancestor particles with high weights in the subsequent periods. Both properties together ensure that the particle system obtained by sampling and resampling is (nearly) optimally adapted to the final target $p_\theta(x_{1:T}|y_{1:T})$.

This explains why the PEIS produces SMC estimates for the marginal likelihood $p_\theta(y_{1:T})$ (obtained according to Equation 7), which are numerically very accurate. In fact, as shown in Scharth and Kohn (2013), the PEIS is capable of producing dramatic improvements in numerical accuracy relative to local SMCs like the BPF and APF. This property is exploited by Scharth and Kohn (2013), when using the PEIS to obtain highly efficient implementations of the particle marginal MH (PMMH) procedure for SSMs.

Despite its nearly perfect global adaption, the PEIS when implemented with resampling steps in every period will suffer, as any SMC, from the SMC-path degeneracy phenomenon causing the poor mixing of the baseline PG. However, since the PEIS globally reduces the variance of the SMC-IS weights to a (close to) minimum level, it typically suffices to resample only at a few periods, which substantially reduces the path degeneracy. This motivates the implementation of the baseline PG using the PEIS with sparse resampling at a few predetermined time period (PG-PEIS-sparse).

Finally, it is important to note, that the PEIS implementation requires to run the sequence of T auxiliary regressions (12) *before* producing via the sequence of SMC steps a weighted particle system $\{x^i_{1:T}, w^i_T\}$. Hence, the global design of the SMC-IS densities used by PEIS comes at additional computational costs relative to the local design of the IS densities used by standard SMC procedures. However, as illustrated by Scharth and Kohn (2013) in the context of SMC approximations of the marginal likelihood as well as in our applications to PG algorithms below, the substantial improvements of the approximation to $p_\theta(x_{1:T}|y_{1:T})$ gained by the PEIS may outweigh its additional computational costs.

In conclusion of this generic presentation, we provide the full PEIS algorithm:

PEIS algorithm

(i) *Compute* $\hat{c} = (\hat{c}_1, \ldots, \hat{c}_T)$ *by iteratively drawing from* $q(x_{1:T}; \hat{c}^{[\ell]})$ *and producing* $c^{[\ell+1]}$ *via*

the T auxiliary EIS regressions in Equation (12), and store $\hat{c}$.

(ii) *For $t = 1$:*

- *Sample $x_1^i \sim q_1(x_1, \hat{c}_1)$, compute the IS weights*

$$w_1^i = \frac{g_\theta(y_1|x_1^i) f_\theta(x_1^i) \chi_2(x_1^i; \hat{c}_2)}{q_1(x_1^i; \hat{c}_1)}, \tag{17}$$

store $\bar{w}_1 = \sum_{i=1}^N w_1^i/N$, and compute normalized weights $W_1^i = w_1^i/(\sum_{l=1}^N w_1^l)$.

- *If resampling, sample $\bar{x}_1^i \sim \sum_{i=1}^N W_1^i \delta_{x_1^i}(dx_1)$ and set the IS weights to $W_1^i = 1/N$, otherwise set $\bar{x}_1^i = x_1^i$.*

For $t = 2, ..., T$:

- *Sample $x_t^i \sim q_t(x_t|\bar{x}_{1:t-1}^i, \hat{c}_t)$ and set $x_{1:t}^i = (x_t^i, \bar{x}_{1:t-1}^i)$;*
- *compute the IS weights*

$$w_t^i = W_{t-1}^i \frac{g_\theta(y_t|x_t^i) f_\theta(x_t^i|x_{t-1}^i) \chi_{t+1}(x_{1:t}^i, \hat{c}_{t+1})}{k_t(x_{1:t}^i; \hat{c}_t)}, \tag{18}$$

store $\bar{w}_t = \sum_{i=1}^N w_t^i$, and compute normalized weights $W_t^i = w_t^i/(\sum_{l=1}^N w_t^l)$.

- *If resampling, sample $\bar{x}_{1:t}^i \sim \sum_{i=1}^N W_t^i \delta_{x_{1:t}^i}(dx_{1:t})$ and set the IS weights to $W_t^i = 1/N$, otherwise set $\bar{x}_{1:t}^i = x_{1:t}^i$.*

(iii) *If required, compute the SMC likelihood estimate according to Equation (7):*

$$\hat{z}_T = \hat{p}_\theta(y_{1:T}) = \prod_{t=1}^T \bar{w}_t. \tag{19}$$

4.2 Particle Gibbs with ancestor sampling (PGAS)

In order to address the poor mixing of the baseline PG, Lindsten et al. (2014) developed the PGAS. It exploits the fact that it suffices to suppress the degeneration of the particle system towards the retained conditional SMC reference trajectory $x'_{1:T}$ and not the SMC-path degeneracy per se to improve the mixing. Based on this insight, the basic idea of the PGAS is to break this reference trajectory into pieces, so that the particle system tends to degenerate to something different than the reference trajectory.

In particular, the PGAS augments each period-t conditional-SMC resampling step by randomly

selecting from the set $\{x^i_{1:t-1}\}_{i=1}^N$ (including the reference particle $x'_{1:t-1}$) one ancestor particle which is used to assign a potentially new $x_{1:t-1}$-history to the partial frozen path $x'_{t:T}$. This produces a concatenated full path $[x^i_{1:t-1}, x'_{t:T}]$, and the corresponding (non-normalized) weight for selecting $x^i_{1:t-1}$ as the new ancestor for $x'_{t:T}$ is given by

$$\tilde{w}^i_{t-1|T} = w^i_{t-1} \frac{\gamma_T\left([x^i_{1:t-1}, x'_{t:T}]\right)}{\gamma_{t-1}(x^i_{1:t-1})}. \tag{20}$$

In Bayesian terms, the components of those anchestor sampling weights for the reference particle are the prior probability of the ancestor particle $x^i_{1:t-1}$ given by the 'standard' SMC-IS weights w^i_{t-1} and the likelihood that the partial reference path $x'_{t:T}$ originated from $x^i_{1:t-1}$ which is represented by the ratio of the targets $\gamma_T(\cdot)/\gamma_{t-1}(\cdot)$.

As shown by Lindsten et al. (2014, Theorem 1), the invariance property of the baseline PG is not violated by this additional AS step. However, since this AS step sequentially assigns in each period a potentially new ancestor to $x'_{t:T}$, it will produce a reference path $x'_{1:T}$ which tends to differ from the other (degenerated) conditional SMC paths $\{x^i_{1:T}\}_{i=2}^N$. Thus, while not preventing the particle system to degenerate, the PGAS typically improves the mixing of the baseline PG. Furthermore, if the variance of the AS weights $\tilde{w}^i_{t-1|T}$ in Equation (20) is minimized, the potential diversity of the resulting PGAS reference path $x'_{1:T}$ is maximized. Hence, by reducing the variance of $\tilde{w}^i_{t-1|T}$, we can improve the mixing of the PGAS trajectories $x^{(j)}_{1:T}$ under $p_\theta(x_{1:T}|y_{1:T})$.

In Lindsten et al. (2014), the PGAS is implemented by relying upon the BPF (PGAS-BPF), which uses $\pi_t(x_{1:t}) \propto p_\theta(x_{1:t}, y_{1:t})$, as given in Equation (3), together with $q_t(x_t|x_{1:t-1}) \equiv f_\theta(x_t|x_{t-1})$, so that according to Equation (6) the 'prior' weights are $w^i_{t-1} = W^i_{t-2} g_\theta(y_{t-1}|x^i_{t-1})$. The resulting AS weights are given by

$$\tilde{w}^i_{t-1|T} = W^i_{t-2} g_\theta(y_{t-1}|x^i_{t-1}) p_\theta(x'_{t:T}, y_{t:T}|x^i_{t-1}) \propto W^i_{t-2} g_\theta(y_{t-1}|x^i_{t-1}) f_\theta(x'_t|x^i_{t-1}). \tag{21}$$

This form of AS-weights obtained under the BPF, shows that in scenarios, where the measurement density g_θ is fairly flat in x_t (so that the y_t observations are not very informative about the states x_t) and the transition density f_θ exhibits a large conditional variance, the variation of $\tilde{w}^i_{t-1|T}$ can

be expected to be sufficiently small so as to obtain a sufficiently strong mixing of the PGAS-BPF. However, applications with highly informative observations and/or a state process with small noise produce a large variance of $\tilde{w}^i_{t-1|T}$, so that the efficacy of the PGAS-BPF to improve the mixing of the baseline PG can be expected to be limited.

Under the PEIS with $\pi_t(x_{1:t})$ and w^i_{t-1} as given by Equations (13) and (15) the PGAS ancestor weights become

$$\tilde{w}^i_{t-1|T} \propto \left[W^i_{t-2}\frac{g_\theta(y_{t-1}|x^i_{t-1})f_\theta(x^i_{t-1}|x^i_{t-2})\chi_t(x^i_{1:t-1};\hat{c}_t)}{k_{t-1}(x^i_{1:t-1};\hat{c}_{t-1})}\right]\left[\frac{p_\theta(y_{t:T}|x^i_{t-1})}{\chi_t(x^i_{1:t-1})}\right]. \tag{22}$$

Hence, according to Lemma 1 the PEIS produces PGAS ancestor weights with a (close to) minimal variation: Recall that the IS densities of the PEIS are designed so as to minimize the variance of the prior weights w^i_{t-1} (given by the term in the first bracket of Equation 22). Moreover, the predictive density $p_\theta(y_{t:T}|x_{t-1})$ as function in x_{t-1} is closely approximated by the EIS integrating factor $\chi_t(x_{1:t-1})$ so that the variance of the likelihood for the ancestor $x^i_{1:t-1}$ (term in the second bracket) is also close to a minimum level.

In conclusion, the PEIS not only produces a (conditional) SMC particle system which is nearly perfectly globally adapted to $p_\theta(x_{1:T}|y_{1:T})$, but also generates a very high potential diversity of the reference particle generated by the additional AS step. As a result, we expect to improve the mixing of the PGAS paths for $x_{1:T}$ obtained under local procedures like the BPF by relying upon the global PEIS (PGAS-PEIS).

4.3 Particle Gibbs with an additional MH step (PGMH)

The PGMH proposed by Holenstein (2009, Algorithm 3.6) in order to address the poor-mixing problem of the baseline PG bypasses the SMC-path degeneracy by using an additional particle-MH step proposing in each iteration step j a completely new SMC path denoted by $x^*_{1:T}$. This new path is MH-compared with the old path $x^{(j-1)}_{1:T}$ based upon the (conditional) SMC estimates of their respective marginal likelihood. The resulting PGMH algorithm is given by:

PGMH algorithm

(i) *Initialization* ($j = 0$)*: Set randomly* $\theta^{(0)}$*, run an SMC targeting* $p_{\theta^{(0)}}(x_{1:T}|y_{1:T})$*, and sample* $x_{1:T}^{(0)} \sim \hat{p}_{\theta^{(0)}}(x_{1:T}|y_{1:T})$.

(ii) *For iteration* $j \geq 1$*:*

- *sample* $\theta^{(j)} \sim p(\theta|x_{1:T}^{(j-1)}, y_{1:T})$,
- *run a conditional SMC targeting* $p_{\theta^{(j)}}(x_{1:T}|y_{1:T})$ *conditional on* $x_{1:T}^{(j-1)}$*, and compute the likelihood estimate* $\hat{p}_{\theta^{(j)}}(y_{1:T})$*,*
- *run an SMC targeting* $p_{\theta^{(j)}}(x_{1:T}|y_{1:T})$*, sample* $x_{1:T}^{*} \sim \hat{p}_{\theta^{(j)}}(x_{1:T}|y_{1:T})$*, and compute the likelihood estimate* $\hat{p}^{*}_{\theta^{(j)}}(y_{1:T})$*,*
- *with probability*

$$1 \wedge \frac{\hat{p}^{*}_{\theta^{(j)}}(y_{1:T})}{\hat{p}_{\theta^{(j)}}(y_{1:T})} \tag{23}$$

set $x_{1:T}^{(j)} = x_{1:T}^{*}$*, otherwise* $x_{1:T}^{(j)} = x_{1:T}^{(j-1)}$.

The efficacy of this PGMH algorithm to improve the mixing of the baseline PG critically depends on the numerical precision of the (conditional) SMC estimates for the marginal likelihood $p_\theta(y_{1:T})$ defining the acceptance rate of the additional particle MH-step as given in Equation (23). In particular, if the SMC delivers noisy estimates for $p_\theta(y_{1:T})$ the MH updates for $x_{1:T}$ can get stuck for many iterations leading to very poor mixing. Hence, efficient PGMH implementations are those for which the SMC marginal likelihood estimates have a small variance. Since, as discussed in Section 4.1, the PEIS produces very precise SMC estimates, we expect a high efficacy of the PGMH in improving the mixing of the baseline PG by relying upon PEIS estimates for $p_\theta(y_{1:T})$ as given by Equation (19) (PGMH-PEIS).

5. Applications

In this section we discuss two applications illustrating how the PEIS can be used to improve the mixing of the baseline PG and its extensions provided by the PGAS and PGMH: a stochastic volatility model and a Gaussian nonlinear local level model. The specific models are described in Section 5.1.

The PEIS implementation for both models is then outlined in Section 5.2 and the results are discussed in Section 5.3.

5.1 Example models and data

The first example is a standard stochastic volatility (SV) model for the volatility of financial returns (see, e.g., Ghysels et al., 1996). It has the form

$$y_t = \beta \exp\{x_t/2\}\eta_t, \qquad \eta_t \sim \text{i.i.d.}N(0,1), \tag{24}$$

$$x_t = \delta x_{t-1} + \nu\epsilon_t, \qquad \epsilon_t \sim \text{i.i.d.}N(0,1), \tag{25}$$

where y_t is the asset return observed at period t, x_t is the latent log volatility and $\theta = (\beta, \delta, \nu)'$. The innovations ϵ_t and η_t are mutually independent. Assuming $|\delta| < 1$, the distribution of the initial state is given by $x_1 \sim N(0, \nu^2/[1-\delta^2])$.

The second example is a time-discretized version of a constant elasticity of variance (CEV) diffusion model for daily short-term interest rates (Chan et al., 1992). In order to account for microstructure noise, which is to be expected for interest rate data at the daily frequency, the basic CEV specification is extended to include a noise component (Aït-Sahalia, 1999 and Kleppe and Skaug, 2015). The resulting model for the interest rate y_t observed at day t with x_t the latent interest-rate state, is described as

$$y_t = x_t + \sigma_y\eta_t, \qquad \eta_t \sim \text{i.i.d.}N(0,1), \tag{26}$$

$$x_t = x_{t-1} + \Delta(\alpha - \beta x_{t-1}) + \sigma_x x_{t-1}^{\gamma}\sqrt{\Delta}\epsilon_t, \qquad \epsilon_t \sim \text{i.i.d.}N(0,1), \tag{27}$$

where ϵ_t and η_t are independent and $\Delta = 1/252$. The parameters are $\theta = (\alpha, \beta, \sigma_x, \gamma, \sigma_y)'$. As the stationary distribution of x_t is not known analytically, we assume for the initial state x_1 a normal distribution with a mean set equal to the observed value of y_1 and a standard deviation of 100 basis points so that $x_1 \sim N(y_1, [0.01]^2)$.

The data we use for the SV model are daily log returns, multiplied by 100, on the S&P 500 stock index from October 1, 1999 to September 30, 2009, with a sample size of $T = 2515$. The data for

CEV model consists of daily 7-day Eurodollar deposit spot rates from January 2, 1983 to February 25, 1995, with $T = 3082$. (This data set is discussed in more detail in Aït Sahalia, 1996). See Figure 1 for time series plots of the SV and CEV data.

The two example models differ in their statistical structure and pose different challenges to PG algorithms. The SV model involves a linear Gaussian transition density and a measurement density which is non-Gaussian in the states. In the SV return data the parameter estimates imply a measurement density which is not very informative about the states and state innovations which are fairly volatile. This represents a scenario, where standard SMCs typically exhibit a satisfactory performance. However, in the SV data, the level of the return volatility features abrupt changes, like the dramatic increase associated with the last financial crisis in the second half of the 2000s. Such a burst of volatility poses a challenge for standard SMCs as it is difficult for their IS densities to properly adjust. In the CEV model we have a nonlinear Gaussian state transition density f_θ coupled with a measurement density g_θ which is Gaussian in the states. In the interest data for this model, the estimated standard deviation of the measurement error σ_y we obtain is small relative to the typical standard deviation of the state innovations $\sigma_x x_{t-1}^\gamma \sqrt{\Delta}$, so that the observations are much more informative about the states than in the SV model. This leads to a large sensitivity of SMC procedures to outliers (see, e.g., DeJong et al., 2013) with potential adverse effect on the efficiency of the PG. Such outliers are frequently observed in interest rate data.

5.2 PEIS implementation

As discussed in Section 4.1, the implementation of (P)EIS requires to select a parametric class for the EIS density kernel $k_t(x_{1:t}, c_t)$ capable of providing a good functional approximation to the period-t EIS target given by (see Equation 11)

$$g_\theta(y_t|x_t) f_\theta(x_t|x_{t-1}) \chi_{t+1}(x_{1:t}; c_{t+1}). \tag{28}$$

Both example models have in common that they involve a Gaussian transition density with a conditional mean μ_t and variance σ_t^2, written as

$$f_\theta(x_t|x_{t-1}) = f_N(x_t|\mu_t, \sigma_t^2), \tag{29}$$

where for the SV model we have $\mu_t = \delta x_{t-1}$ and $\sigma_t^2 = \nu^2$, while for the CEV model it is the case that $\mu_t = x_{t-1} + \Delta(\alpha - \beta x_{t-1})$ and $\sigma_t^2 = \sigma_x^2 x_{t-1}^{2\gamma}\Delta$.

In such Gaussian transition cases it is natural to select for k_t a Gaussian kernel in x_t which consists of the product of the Gaussian transition density f_θ already included in the EIS target in Equation (28) and a Gaussian kernel approximation in x_t to the remaining non-Gaussian product $g_\theta\,\chi_{t+1}$. The corresponding EIS kernel k_t can be parameterized as

$$k_t(x_{1:t}, c_t) = f_\theta(x_t|x_{t-1})\zeta_t(x_t; c_t), \quad \text{with} \quad \zeta_t(x_t; c_t) = \exp\{c_{1t}x_t + c_{2t}x_t^2\}, \tag{30}$$

where ζ_t is the Gaussian kernel designed to approximate $g_\theta\,\chi_{t+1}$ with auxiliary EIS parameters $c_t = (c_{1t}, c_{2t})$. Since f_θ in the kernel k_t as defined in Equation (30) is also a component of the EIS target, it cancels out in the EIS regressions (12). Hence, they simplify into simple linear LS regressions of $\ln[g_\theta(y_t|x_t^i)\chi_{t+1}(x_{1:t}^i; \hat{c}_{t+1})]$ on x_t^i and $(x_t^i)^2$ and a constant.

From Equation (30) it immediately follows that the Gaussian EIS density for $x_t|x_{t-1}$ has the form

$$q_t(x_t|x_{t-1}, c_t) = f_N\left(x_t|m_t, v_t^2\right), \tag{31}$$

with

$$v_t^2 = \frac{\sigma_t^2}{1 - 2c_{2t}\sigma_t^2}, \qquad m_t = v_t^2\left(\frac{\mu_t}{\sigma_t^2} + c_{1t}\right), \tag{32}$$

and integrating k_t w.r.t. x_t leads to an integrating factor of the form

$$\chi_t(x_{1:t-1}, c_t) = \frac{\sqrt{v_t^2}}{\sqrt{\sigma_t^2}}\exp\left\{\frac{1}{2}\left(\frac{m_t^2}{v_t^2} - \frac{\mu_t^2}{\sigma_t^2}\right)\right\}. \tag{33}$$

As mentioned in Section 4.1, the sequence of EIS regressions in Equation (12) producing near

optimal values for the EIS parameters c need to be iterated since the R trajectories $\{x_{1:T}^i\}_{i=1}^R$ used in the EIS regression are to be drawn from the joint IS density $q(x_{1:T};c)$ itself. This requires selecting an initial value $\hat{c}^{[0]} = (\hat{c}_1^{[0]}, ..., \hat{c}_T^{[0]})$ and then for iteration $\ell = 1, ..., L$ using trajectories from $q(x_{1:T}; \hat{c}^{[\ell-1]})$ to compute a new $c^{[\ell]}$. Actually, when using a number of EIS trajectories R of the order of 3 to 5 times the number of parameters in the period-t EIS regression, only the first 2 or 3 iterations produce significant improvements on the approximation of the EIS targets as measured by the R^2 of the EIS LS regressions. Thus we preset the number of EIS iterations at $L = 4$ and set $R = 15$. As for $\hat{c}^{[0]}$, we can exploit that in the case of the CEV model the measurement density $g_\theta(y_t|x_t)$ in the EIS targets itself is a Gaussian kernel in x_t so that we can select $\hat{c}_t^{[0]}$ as that value for c_t for which $\zeta_t(x_t; c_t) \propto g_\theta(y_t|x_t)$. It follows that the resulting initial EIS density $q_t(x_t|x_{t-1}, \hat{c}_t^{[0]})$ corresponds to the IS density of the conditional optimal particle filter. For the SV model, $g_\theta(y_t|x_t)$ is non-Gaussian in x_t so that we use for $\ln \zeta_t$ a second-order Taylor-series approximation in x_t to $\ln g_\theta$ to obtain an initial value $\hat{c}_t^{[0]}$. The R^2 we find in the final sequence of EIS regressions is typically larger than 0.99, which indicates that the resulting EIS densities are nearly perfectly globally adapted to the SMC target $p_\theta(x_{1:T}|y_{1:T})$.

The functional forms of the EIS densities given in Equations (30) to (33) together with the near optimal value $\hat{c} = \hat{c}^{[L]}$ are used to run the SMC steps (ii) and (iii) of the PEIS algorithm provided in Section 4.1. Note that this PEIS algorithm covers the standard BPF as a special case with $\hat{c} \equiv 0$, leading to $k_t(x_{1:t}; 0) = f_\theta(x_t|x_{t-1})$ with $\chi_t(x_{1:t-1}; 0) = 1$.

5.3 Results

Here we present simulation experiments using the SV and CEV model to compare the following 8 PG schemes:[2] The baseline PG based on the BPF (PG-BPF), PEIS (PG-PEIS) and PEIS with sparse resampling (PG-PEIS-sparse), then the PGAS combined with the BPF (PGAS-BPF) and PEIS (PGAS-PEIS) and, finally, the PGMH using the BPF (PGMH-BPF), PEIS (PGMH-PEIS)

[2]In addition, for the CEV model we considered the PGAS based on the fully adapted APF, which extends the intermediate target in Equation (3) by including the one-period ahead predictive density $p_\theta(y_{t+1}|x_t)$ and uses the conditional optimal IS density $p_\theta(x_t|y_t, x_{t-1})$ (see, Pitt et al., 2012, and Pitt et al., 2015). However, the results are not reported here as the predictive density and the conditional optimal IS density are analytically known only for the CEV model but not for the SV application. Moreover, the PGAS results for the CEV model show no improvements when replacing the BPF by the fully adapted APF.

and PEIS-sparse (PGMH-PEIS-sparse). We use multinomial resampling for the SMC resampling steps. For the PEIS-sparse the resampling is conducted only every 500 periods. The PG methods were all implemented in the interpreted language MATLAB, making computing times comparable.

For all the experiments we use the real data sets described in Section 5.1. The corresponding maximum likelihood (ML) estimates based on EIS evaluations of the likelihood are $(\beta, \delta, \nu) = (1.065, 0.992, 0.122)$ for the SV model and $(\alpha, \beta, \sigma_x, \gamma, \sigma_y) = (0.0097, 0.1656, 0.4250, 1.201, 0.0005)$ for the CEV model.

5.3.1 Mixing of Particle Gibbs for fixed parameters

The first experiment is designed to analyze the mixing of the PG algorithms w.r.t. the states under their joint posterior $p_\theta(x_{1:T}|y_{1:T})$ for a fixed value of the parameters θ. Throughout this experiment we set the parameters equal to their ML estimates and generate samples from this density using the PG algorithms, which are all implemented with two different numbers of particles, $N = 30$ and $N = 1000$. All methods are simulated for 1100 iterations, where the first 100 burn-in iterations are discarded.

In order to compare the mixing, we follow Lindsten et al. (2014) and compute the update rate for each x_t $(t = 1, ..., T)$ which is defined as the proportion of PG iterations where the value for x_t has changed. The update rates for the 8 PG algorithms plotted against time t are provided in Figure 2 for the SV model and in Figure 3 for the CEV model. They reveal that in both example models the update rate for the baseline PG-BPF for $N = 30$ as well as $N = 1000$ rapidly decreases for an increasing distance of t to the final period T. These poor update rates reflect the typical SMC path degeneracy causing, as discussed in Section 4.2, the state trajectory $x_{1:T}^{(j)}$ at PG iteration step j to coalesce with the previous trajectory $x_{1:T}^{(j-1)}$. That this poor mixing problem cannot be addressed satisfactorily by replacing the locally designed BPF by an SMC which is nearly perfectly globally adapted is evidenced by the update rates of the PG-PEIS: Even if they increase relative to the PG-BPF they fall in both models, even with $N = 1000$ particles, below 20% for the states of the first 500 periods. This is an illustration of the 'unavoidable' SMC path degeneracy which we would obtain under a fully optimal SMC when resampling is performed every period. The update

rates for the PG-PEIS-sparse remaining above 70% across all periods show that, as expected, sparse resampling greatly improves the mixing of the PG-PEIS by reducing the path degeneracy.

The comparison of the baseline PG-BPF with the PGAS-BPF shows that the additional AS step also increases significantly the average probability of updating x_t across all periods which is consistent with the results reported by Lindsten et al. (2014). However, for the CEV model, in particular, this probability drops dramatically in many periods, indicating that in these periods very few particles tend to keep all the weights across the PG iterations. As discussed in Section 4.2, this stems from the model's tight measurement distribution which makes the PGAS particularly vulnerable to outliers as they produce AS weights with a large variance (see Equation, 21). This effect appears to be less acute for the SV model reflecting the fact that its measurement distribution is not very sensitive to the state. When combined with PEIS, the PGAS with as little as $N = 30$ particles produce update rates which are uniformly above 95% for both, the CEV model and the SV model, indicating a close to perfect and robust mixing of the PGAS.

Turning to the PG augmented by an additional MH move, we also find in both example models a substantial improvement in the mixing when replacing the BPF by PEIS or PEIS-sparse. Those improvements reflect the fact that, as discussed in Section 4.3, PEIS(-sparse) produce numerically far more accurate SMC estimates of the marginal likelihood than the BPF.

For a further comparison of the PG methods, we compute the effective sample size (ESS) of the posterior samples for the state variable x_t at each time period t. The ESS is defined as

$$\text{ESS} = M \left[1 + 2 \textstyle\sum_{j=1}^{J} \gamma(j)\right]^{-1}, \tag{34}$$

where M is the size of the posterior sample, and $\sum_j \gamma(j)$ the sum of the J monotone sample autocorrelations as estimated by the initial monotone sequence estimator proposed by Geyer (1992). The interpretation is that the M PG draws lead to the same precision as a hypothetical i.i.d. sample from the posterior of size ESS, so that large values for ESS are preferable. We consider the minimum, median and maximum ESS over the T sampled state variables. These ESS values are computed for 10 independent complete PG runs from which we take the corresponding averages. In order to account for different computing times, we also compute the (average) minimum ESS standardized

by the Central Processor Unit (CPU) time required to run a PG algorithm. It measures the time it takes to obtain one i.i.d. draw of the *complete* $x_{1:T}$-trajectory from its posterior. The ESS results are reported in Table 1 for the SV model and in Table 2 for the CEV model.

The results for both models show that, for a given number of particles N, the PEIS(-sparse) substantially increases the median and the minimum ESS of the baseline PG, PGAS and PGMH relative to their corresponding BPF counterpart. The largest i.i.d. sample from $p_\theta(x_{1:T}|y_{1:T})$ per hour computing time is produced by the PG-PEIS-sparse with $N = 1000$ for the SV model, and by the PGAS-PEIS with $N = 30$ for the CEV model. This illustrates that the improvements of the PG approximations to $p_\theta(x_{1:T}|y_{1:T})$ gained by the global PEIS outweigh its additional computational costs relative to the locally designed BPF.

5.3.2 Full Bayesian analysis

Here, we compare the performance of the PG algorithms for a full Bayesian analysis of the two example models. For the parameters of both models we select fairly uninformative priors (for details of the prior selection, see Appendix 2). In light of the severe mixing problems of the PG-BPF, PG-PEIS and PGMH-BPF documented in the previous section, the remainder investigation focuses on the efficiency of the PG-PEIS-sparse, PGAS-BPF, PGAS-PEIS, PGMH-PEIS and PGMH-PEIS-sparse. For all of those five methods we use throughout 50,000 PG iterations where the first 10,000 burn-in iterations are discarded.

The Bayesian posterior results for the five PG procedures, each based on $N = 30$ particles, are summarized in Table 3 for the SV model and in Table 4 for the CEV model. Both tables report the following statistics for the model parameters (θ), the initial (x_1), middle ($x_{T/2}$) and last state (x_T): The PG posterior mean and standard deviation together with the ESS and ESS standardized by computing time. All statistics reported in Tables 3 and 4 are sample averages which are computed from 10 independent replications obtained by running each of the PG algorithms under 10 different seeds. The tables also provide the corresponding statistics for the 'ideal' Gibbs sampler, i.e., the sampler which simulates $x_{1:T}$ directly from the true posterior $p_\theta(x_{1:T}|y_{1:T})$. This fictitious Gibbs sampler is approximated by the PGAS-PEIS implemented with $N = 10,000$ particles. Since the

PG algorithms can be seen as MC approximations of the ideal Gibbs sampler, the latter provides a natural benchmark for the mixing performance of the former (see, e.g., Lindsten et al. 2014).

From the results for the SV model in Table 3 we see that with $N = 30$ all five PG algorithms produce MC estimates of the posterior means which are close to those of the ideal Gibbs sampler and the corresponding ML estimates. The ESS values indicate that replacing the BPF by PEIS improves, as expected from the results in Section 5.3.1, the mixing of the PGAS for the parameters and states, and shifts the ESS values closer to those of the ideal Gibbs sampler. The remaining PG-schemes based upon PEIS or PEIS-sparse also show a satisfactory mixing relative to the ideal Gibbs. Most critical for a posterior Gibbs analysis of the parameters θ appears to be the scaling parameter β, which has among all parameters and across all PG procedures the smallest ESS value. Hence, the mixing of the sampled β's sets the limit w.r.t. the amount of i.i.d. draws for θ which can be generated for a given number of Gibbs iterations or a fixed computing time. In terms of the largest minimum ESS of the sampled parameters per hour computing time, the PG-PEIS-sparse and PGAS-BPF show the best performance. For $N = 30$ particles both produce per hour 7 i.i.d. draws from the marginal posterior of the parameters $p(\theta|y_{1:T})$.

In order to analyze the robustness of the PG procedures w.r.t. the selected number of particles, we plot in Figure 4 the autocorrelation functions (ACF) of the sampled β-parameter for the PGAS-BPF and PGAS-PEIS for a range of different number of particles N. The ACF plots reveal that the PEIS version of the PGAS produce comparable mixing rates for any number of particles N larger than 30, suggesting that it does not require more than $N = 30$ particles to obtain a performance which comes close to that of the ideal Gibbs. In contrast, for the BPF counterpart to achieve this performance it needs more than $N = 100$ particles.

Turning to the PG posterior results for the CEV model in Table 4, we first note that the MC estimates for the posterior mean of the parameters θ associated with the PG procedures based on PEIS are all in close agreement with those of the ideal Gibbs and their ML counterparts. For the PGAS based on BPF, however, the posterior parameter estimates substantially differ from those benchmarks. These serious biases are consistent with the results of Section 5.3.1, showing that in situations involving tight measurement densities coupled with outliers the PGAS-BPF has severe

problems to fully explore the domain of the states under $p_\theta(x_{1:T}|y_{1:T})$. That the measurement density in this example is fairly tight is indicated by the tiny value of the estimates for the standard deviation σ_y. In contrast to the PGAS-BPF, the PG procedures based on PEIS ensure even in this challenging scenario a fast and reliable exploration of $p_\theta(x_{1:T}|y_{1:T})$ and lead to accurate posterior estimates for the parameters. Note also that the ESS values in Table 4 indicate that with $N = 30$ particles the mixing rate of all PG procedures using PEIS is very close to that of the ideal Gibbs sampler. (Since the PGAS-BPF parameter draws apparently fail to appropriately represent the posterior $p(\theta|y_{1:T})$ we refrain from reporting the corresponding values of the ESS statistic.)

The two parameters with the lowest ESS values are σ_x and γ. In Figure 4 we plot the ACF for those two parameters sampled by the PGAS-PEIS under different number of particles. The results reveal that the PGAS-PEIS achieves a performance close to that of an ideal Gibbs with as little as $N = 5$ particles.

6. Conclusions

The particle Gibbs (PG) is a flexible and easy to implement tool for conducting Bayesian analyses of state space models. It uses sequential Monte Carlo (SMC) inside the Gibbs procedure in order to update the latent state trajectories. However, in high-dimensional applications when there is path degeneracy in the underlying SMC sampler the baseline PG suffers from severe mixing problems. Refinements designed to improve the mixing of the baseline PG introduce an ancestor sampling step to the underlying SMC (PGAS) or an additional Metropolis-Hastings move for the update of the state trajectories (PGMH). However, such refinements when implemented using a standard locally designed SMC procedure such as the bootstrap particle filter of Gordon et al. (1993) can still be prone to mixing problems, particularly, in applications involving narrowly distributed measurement variables and given the presence of outliers.

Here, we have proposed to combine the PG and its refinements with Particle Efficient Importance Sampling (PEIS) to overcome the mixing problem of the PG. The PEIS is an SMC algorithm based on a recursive sequence of simple auxiliary regressions designed to construct highly efficient SMC importance sampling densities and resampling weights, which are globally adapted to the targeted

posterior density of the states. We have shown that the PG when combined with PEIS leads to significant improvements of the mixing w.r.t. the state trajectories relative to PG procedures based on standard locally designed SMC algorithms. By such improvements of the mixing, PG implementations based on PEIS allow for numerically accurate and reliable Bayesian parameter estimates in state space models as illustrated by the applications to a stochastic volatility model for asset returns and a constant elasticity of variance model for interest rates.

Acknowledgements

We thank participants of the 2015 Rhenisch Multivariate Time Series Econometrics (RMSE) Meeting (University of Cologne) and of the CMStatistics 2015 (ERCIM 2015) conference, London. We thank Yacine Aït-Sahalia for sharing the Eurodollar data used in Aït Sahalia (1996). We are grateful to the Regional Computing Center at the University of Cologne for providing parts of the computational resources required. R. Liesenfeld acknowledges support by the Deutsche Forschungsgemeinschaft (grant LI 901/3-1).

Appendix 1: Proof of Lemma 1

The SMC-IS weights of the PEIS in Equation (15) obtain immediately by replacing in the general form of the SMC-IS weights given by Equation (6) the kernel of the intermediate targets γ_t and γ_{t-1} and the IS densities q_t by the intermediate PEIS targets and PEIS densities defined, respectively, in Equations (13) and (14).

In order to obtain the relationship between the predictive density $p_\theta(y_{t+1:T}|x_t)$ and the EIS integrating factor $\chi_{t+1}(x_{1:t}; c_t)$ defined in Equation (10), we first write the predictive densities for the SSM given in Equation (1) as the following backward-recursive sequence of integrals:

$$p_\theta(y_T|x_{T-1}) = \int g_\theta(y_T|x_T) f_\theta(x_T|x_{T-1}) dx_T, \tag{35}$$

$$p_\theta(y_{t:T}|x_{t-1}) = \int p_\theta(y_{t+1:T}|x_t) g_\theta(y_t|x_t) f_\theta(x_t|x_{t-1}) dx_t, \qquad t = T-1, ..., 2, \tag{36}$$

$$p_\theta(y_{1:T}) = \int p_\theta(y_{2:T}|x_1) g_\theta(y_1|x_1) f_\theta(x_1) dx_1. \tag{37}$$

This sequence shows that if the period-T EIS-kernel $k_T(x_{1:T}; c_T)$ as a function in $x_{T-1:T}$ is close to be proportional to the integrand $g_\theta(y_T|x_T) f_\theta(x_T|x_{T-1})$ in Equation (35), then (i) its integrating factor $\chi_T(x_{1:T-1}; c_T)$ is close to be proportional to the density $p_\theta(y_T|x_{T-1})$ as a function in x_{T-1}, and (ii) we obtain the following close approximation to the period $T-1$ integrand as given by Equation (36)

$$p_\theta(y_T|x_{T-1}) g_\theta(y_{T-1}|x_{T-1}) f_\theta(x_{T-1}|x_{T-2}) \simeq \tag{38}$$

$$\text{constant} \cdot \chi_T(x_{1:T-1}, c_T) g_\theta(y_{T-1}|x_{T-1}) f_\theta(x_{T-1}|x_{T-2}),$$

where the r.h.s is approximated by the period-$(T-1)$ EIS density kernel $k_{T-1}(x_{1:T-1}; c_{T-1})$. The proof of Equation (16) follows by recursion.

Appendix 2: Prior assumptions

For the parameters of the SV model in Equations (24) and (25) we use the following priors: For $\ln \beta$ we assume a flat prior, and for $(\delta+1)/2$ a Beta prior with a prior mean for δ of 0.86 and a prior variance of 0.012. For ν^2 an inverted chi-squared prior with $\nu^2 \sim p_0 s_0 / \chi^2_{(p_0)}$ and $p_0 = 10$ and $s_0 = 0.01$ is used. The conditional posteriors for β and ν can be simulated directly. To sample from the conditional posterior of δ we use an independent MH sampler (for details, see Kim et al., 1998).

The prior assumptions on the parameters of the CEV model in Equation (26) and (27) are the following: for α and β we assume a Gaussian prior with $\alpha \sim N(0, 1000)$ and $\beta \sim N(0, 1000)$ and for γ a uniform pior on the interval $[0, 4]$. An uninformative inverted chi-squared prior is used for σ_x^2 and σ_y^2 with prior densities given by $p(\sigma_x^2) \propto 1/\sigma_x^2$ and $p(\sigma_y^2) \propto 1/\sigma_y^2$. All conditional posteriors in the CEV model are of known form, except for that of γ, which we sample using Griddy Gibbs.

References

Aït-Sahalia, Y., 1996. Testing continuous-time models of the spot interest rate. Review of Financial Studies 9, 385-426.

Aït-Sahalia, Y., 1999. Transition densities for interest rate and other nonlinear diffusions. Journal of Finance 54, 1361-1395.

Andrieu, C., Doucet, A., and Holenstein, R., 2010. Particle Markov Chain Monte Carlo methods. Journal of the Royal Statistical Society 72, Series B, 269-342.

Cappé, O., Godsill, S.J., and Moulines, E., 2007. An overview of exsting methods and recent advances in sequential Monte Carlo. Proceedings of the IEEE 95, 899-924.

Carter, C.K., and Kohn, R., 1994. On Gibbs sampling for state space models. Biometrika 81, 541-553.

Carter, C.K., Mendes E.F., and Kohn, R., 2014. An extended space approach for particle Markov chain Monte Carlo methods. Working paper, eprint arXiv:1406.5795.

Chan, K.C., Karolyi, G.A., Longstaff, F.A., and Sanders, A.B., 1992. An empirical comparison of alternative models of the short-term interest rate. Journal of Finance 47, 1209-1227.

Chopin, N., 2004. Central limit theorem for sequential Monte Carlo and its application to Bayesian inference. The Annals of Statistics 32, 2385-2411.

Chopin, N., and Singh, S.S., 2013. On the particle Gibbs sampler. Working paper, eprint arXiv:1304.1887.

DeJong, D.N., Liesenfeld, R., Moura, G.V., Richard, J.-F., and Dharmarajan, H., 2013. Efficient likelihood evaluation of state-space representations. The Review of Economic Sudies 80, 538-567

Doucet, A., and Johansen, A.M., 2009. A tutorial on particle filtering and smoothing: Fifteen years later. In: Crisan, D., Rozovskii, B. (eds), The Oxford Handbook of Nonlinear Filtering. Oxford University Press, 656-704.

Fernandez-Villaverde, J., and Rubio-Ramirez, J.F., 2005. Estimating dynamic equilibrium economies: Linear versus nonlinear likelihood. Journal of Applied Econometrics 20, 891-910.

Flury, T., and Shephard, N., 2010. Discussion on: Particle Markov Chain Monte Carlo methods. Journal of the Royal Statistical Society 72, Series B, 311.

Flury, T., and Shephard, N., 2011. Bayesian inference based only on simulated likelihood: Particle filter analysis of dynamic economic models. Econometric Theory 27, 933-956.

Geyer, C.J., 1992. Practical Markov Chain Monte Carlo. Statistical Science 7, 473-483.

Ghysels, E., Harvey, A., and Renault, E., 1996. Stochastic Volatility. In: Maddala, G., Rao, C.R. (eds), Handbook of Statistics, Vol 14. Elsevier Sciences, 119-191.

Gordon, N.J., Salmond, D.J., and Smith, A.F.M., 1993. A novel approach to non-linear and non-Gaussian Bayesian state estimation. IEEE Proceedings-F 140, 107-113.

Holenstein, R., 2009. Particle Markov Chain Monte Carlo. PhD-Thesis, University of British Colombia.

Kim, S., Shephard, N., and Chib, S., 1998. Stochastic volatility: Likelihood inference and comparison with ARCH models. Review of Economic Studies 65, 361-393.

Kleppe, T.S., and Liesenfeld, R., 2014. Efficient importance sampling in mixture frameworks. Computational Statistics and Data Analysis 76, 449-463.

Kleppe, T.S., and Skaug, H.J., 2015. Bandwidth selection in pre-smoothed particle filters. Statistics and Computing, in press (DOI 10.1007/s11222-015-9591-4).

Liesenfeld, R., and Richard, J.-F., 2008. Improving MCMC using efficient importance sampling. Computational Statistics and Data Analysis 53, 272-288.

Lindsten, F., Jordan, M.I., and Schön, T.B., 2014. Particle Gibbs with ancestor sampling. Journal of Machine Learning Research 15, 2145-2184.

Lindsten, F., and Schön, T.B., 2012. On the use of backward simulation in particle Markov chain Monte Carlo methods. Working paper, Linköping University, Sweden.

Pitt, M.K., Hall, J., and Kohn, R., 2015. Bayesian inference for latent factor GARCH models. Working paper, eprint arXiv:1507.01179.

Pitt, M.K., Silva, d.S.R., Giordani, P., Kohn, R., 2012. On some properties of Markov chain Monte Carlo simulation methods based on the particle filter. Journal of Econometrics 171, 134-151.

Pitt, M.K., Shephard, N., 1999. Filtering via simulation: Auxiliary particle filters. Journal of the American Statistical Association 94, 590-599.

Richard, J.-F., Zhang, W., 2007. Efficient high-dimensional importance sampling. Journal of Econometrics 141, 1385-1411.

Ristic, B., Arulampalam, S., Gordon, N., 2004. Beyond the Kalman Filter: Particle Filters for Tracking Applications. Artech House, Boston.

Scharth, M., and Kohn, R., 2013. Particle Efficient Importance Sampling. Journal of Econometrics, forthcoming.

Shephard, N., and Pitt, M.K., 1997. Likelihood analysis on non-Gaussian measurement time series. Biometrika 84, 653-667.

Whiteley, N., 2010. Discussion on: Particle Markov Chain Monte Carlo methods. Journal of the Royal Statistical Society 72, Series B, 306-307.

Whiteley, N., Andrieu, C., Doucet, A., 2010. Efficient Bayesian inference for switching state space models using discrete particle Markov chain Monte Carlo methods. Bristol Statistics Research Report 10:04, University of Bristol.

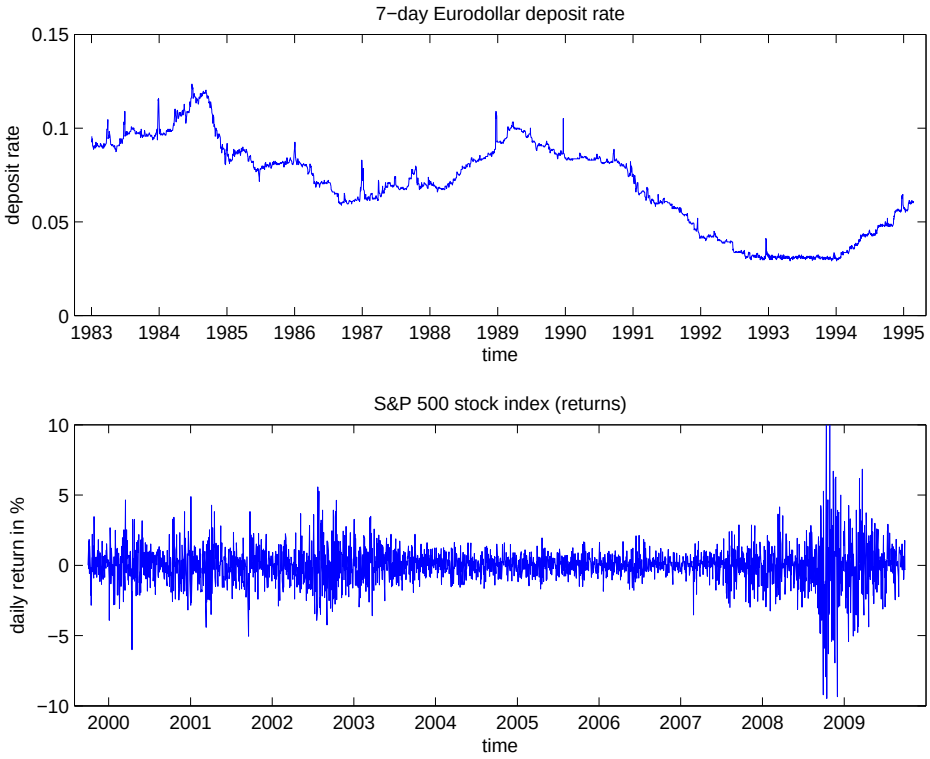

Figure 1. Top panel: The daily short-term Eurodollar interest rates from 1983 to 1995; Bottom panel: The daily returns on the S&P 500 stock index from 1999 to 2009.

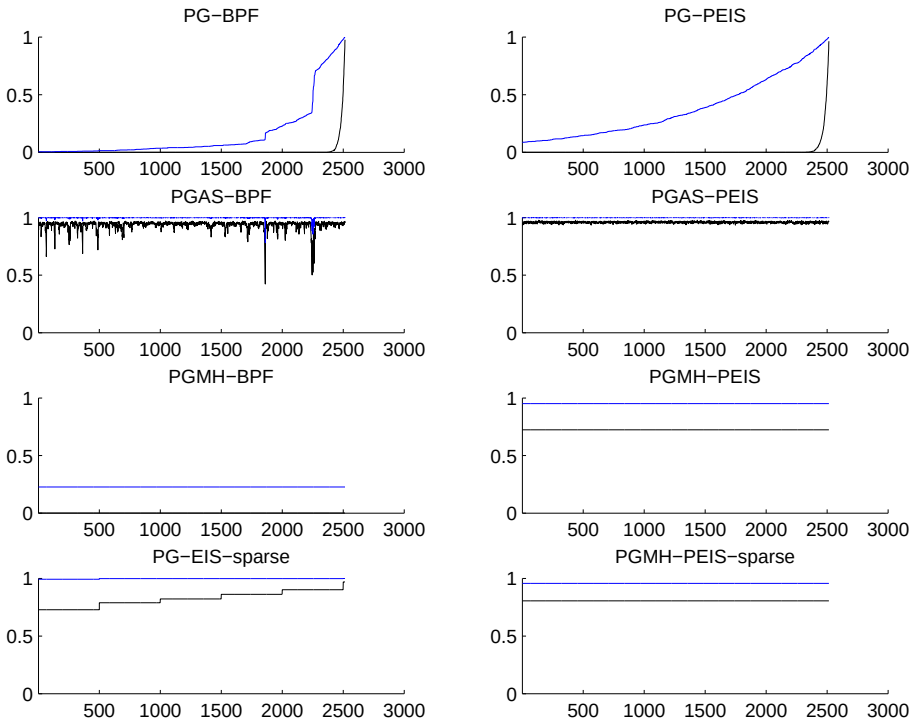

Figure 2. PG update rates for x_t versus $t = 1, ..., T$ for the SV model, using $N = 30$ particle (black line) and $N = 1000$ (blue line).

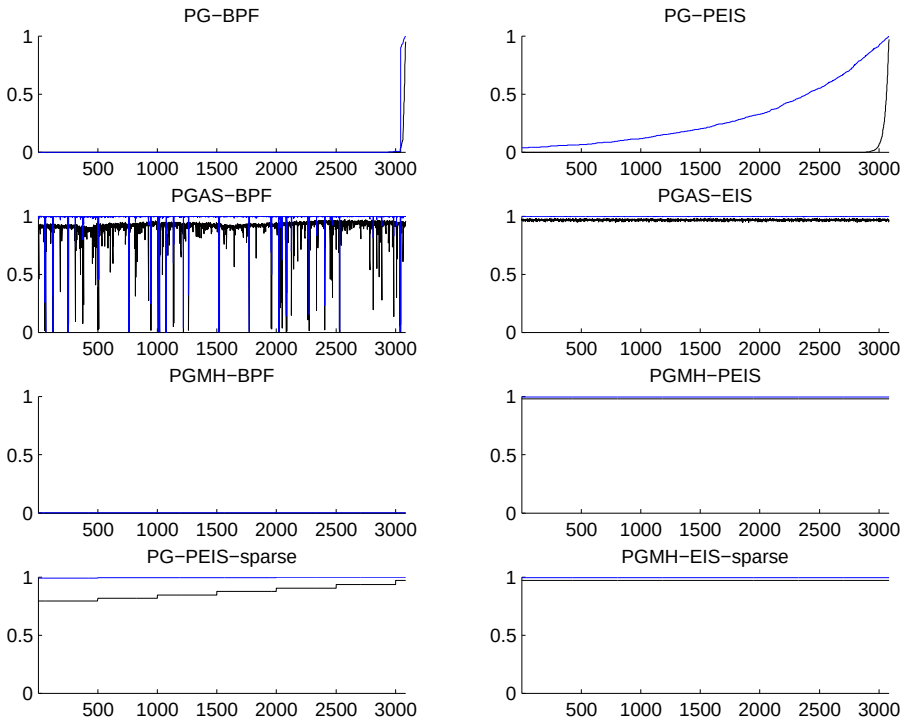

Figure 3. PG update rates for x_t versus $t = 1, ..., T$ for the CEV model, using $N = 30$ particle (black line) and $N = 1000$ (blue line).

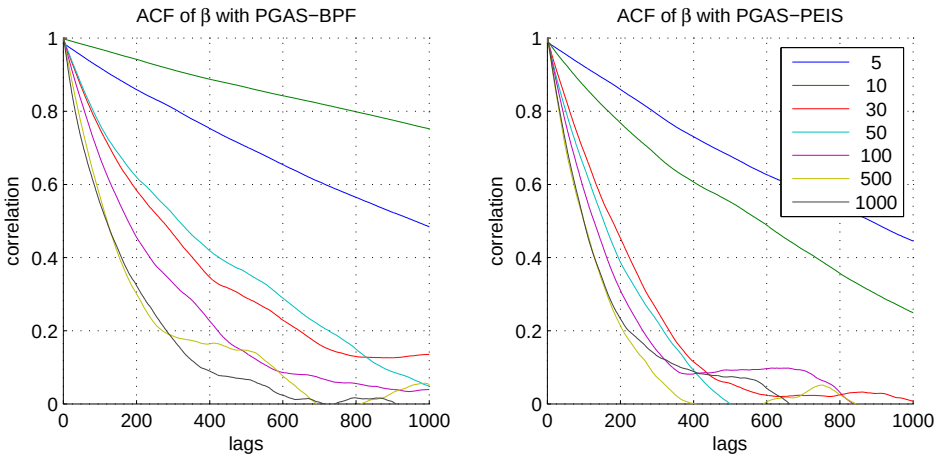

Figure 4. ACFs of the sampled SV parameter β for PGAS-BPF (left) and PGAS-PEIS (right) under different numbers of SMC particles, $N \in \{5, 10, 30, 50, 100, 500, 1000\}$.

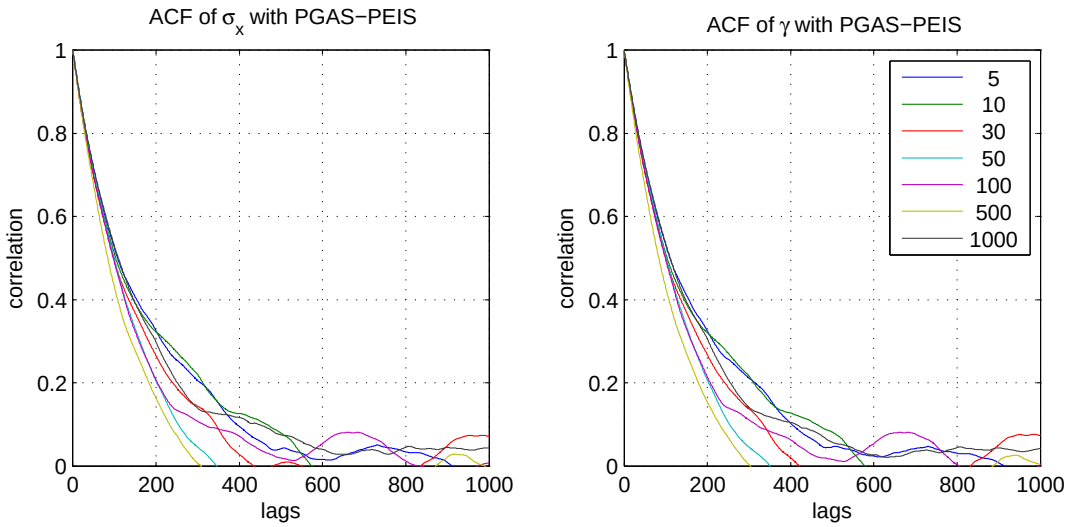

Figure 5. ACFs of the sampled CEV parameters σ_x (left) and γ (right) for the PGAS-PEIS under different numbers of SMC particles, $N \in \{5, 10, 30, 50, 100, 500, 1000\}$.

Table 1. Effective Sample Size for PG Samples from the Posterior of the States in the SV Model for Fixed Parameters

	Number of particles	CPU time in sec	Minimum ESS	Median ESS	Maximum ESS	Minimum ESS per hour CPU time
PG-BPF	30	283	1	1	621	13
PG-BPF	1000	942	3	30	1000	12
PG-PEIS	30	1646	1	1	566	2
PG-PEIS	1000	2397	21	173	999	31
PG-PEIS-sparse	30	1552	332	671	969	771
PG-PEIS-sparse	1000	2249	569	948	1000	912
PGAS-BPF	30	653	45	415	689	254
PGAS-BPF	1000	1499	297	934	1000	716
PGAS-PEIS	30	2042	240	475	707	423
PGAS-PEIS	1000	3038	573	949	1000	686
PGMH-BPF	30	881	1	1	1	4
PGMH-BPF	1000	2454	34	113	210	51
PGMH-PEIS	30	2313	284	538	756	442
PGMH-PEIS	1000	3798	532	876	1000	505
PGMH-PEIS-sparse	30	1906	355	662	860	674
PGMH-PEIS-sparse	1000	3345	552	894	1000	563

NOTE: Results from the PG algorithms are based on 1,100 PG iterations (discarding the first 100 draws). All reported statistics are sample averages computed from 10 independent replications of the PG algorithms under 10 different seeds.

Table 2. Effective Sample Size for PG Samples from the Posterior of the States in the CEV Model for Fixed Parameters

	Number of particles	CPU time in sec	Minimum ESS	Median ESS	Maximum ESS	Minimum ESS per hour CPU time
PG-BPF	30	159	1	1	926	23
PG-BPF	1000	1241	1	1	1000	3
PG-PEIS	30	1228	1	1	916	3
PG-PEIS	1000	3074	10	125	1000	11
PG-PEIS-sparse	30	1148	368	739	1000	1154
PG-PEIS-sparse	1000	2853	555	962	1000	701
PGAS-BPF	30	302	1	871	1000	12
PGAS-BPF	1000	2166	2	963	1000	3
PGAS-PEIS	30	1325	522	902	1000	1421
PGAS-PEIS	1000	3916	545	967	1000	501
PGMH-BPF	30	436	36	36	36	278
PGMH-BPF	1000	2723	9	9	11	12
PGMH-PEIS	30	1628	522	924	1000	1156
PGMH-PEIS	1000	5271	565	961	1000	386
PGMH-PEIS-sparse	30	1404	512	929	1000	1314
PGMH-PEIS-sparse	1000	4727	549	963	1000	418

NOTE: Results from the PG algorithms are based on 1,100 PG iterations (discarding the first 100 draws). All reported statistics are sample averages computed from 10 independent replications of the PG algorithms under 10 different seeds.

Table 3. PG Posterior Analysis of the SV model

		PG-PEIS-sparse	PGAS-BPF	PGAS-PEIS	PGMH-PEIS	PGMH-PEIS-sparse	Ideal Gibbs
	CPU time (hours)	14:30	5:49	18:45	21:26	17:15	
β	post. mean	1.0617	1.0645	1.0754	1.0580	1.0727	1.0708
	post. std.	0.1855	0.1943	0.1892	0.2214	0.1978	0.2003
	ESS	96	41	77	72	96	112
	ESS/hour CPU time	7	7	4	3	6	
δ	post. mean	0.9924	0.9924	0.9924	0.9926	0.9924	0.9924
	post. std.	0.0027	0.0027	0.0027	0.0028	0.0027	0.0028
	ESS	653	467	582	474	538	694
	ESS/hour CPU time	45	80	31	22	31	
ν	post. mean	0.1205	0.1204	0.1201	0.1203	0.1202	0.1206
	post. std.	0.0125	0.0125	0.0125	0.0123	0.0125	0.0128
	ESS	265	345	355	226	251	356
	ESS/hour CPU time	18	59	19	11	15	
x_1	post. mean	0.4246	0.4250	0.4032	0.4531	0.4101	0.4141
	post. std.	0.4997	0.5160	0.5092	0.5514	0.5109	0.5210
	ESS	263	108	204	198	251	261
	ESS/hour CPU time	18	18	11	9	15	
$x_{T/2}$	post. mean	-0.8114	-0.8136	-0.8347	-0.7794	-0.8281	-0.8229
	post. std.	0.4443	0.4600	0.4520	0.5054	0.4540	0.4678
	ESS	172	72	135	125	171	184
	ESS/hour CPU time	12	12	7	6	10	
x_T	post. mean	-0.2229	-0.2253	-0.2432	-0.1903	-0.2377	-0.2335
	post. std.	0.5091	0.5229	0.5154	0.5618	0.5208	0.5271
	ESS	273	113	211	202	252	277
	ESS/hour CPU time	19	19	11	9	15	

NOTE: Results from the PG algorithms for the stochastic volatility model based on 50,000 PG iterations (discarding the first 10,000 draws) and $N = 30$ SMC particles. All reported statistics are sample averages computed from 10 independent replications of the PG algorithms under 10 different seeds. The ML estimates for the parameters are $(\beta, \delta, \nu) = (1.065, 0.992, 0.122)$.

Table 4. PG Posterior Analysis of the CEV model

		PG-PEIS-sparse	PGAS-BPF	PGAS-PEIS	PGMH-PEIS	PGMH-PEIS-sparse	Ideal Gibbs
	CPU time (hours)	20:56	7:45	20:05	23:44	17:47	
α	post. mean	0.0099	0.0067	0.0099	0.0098	0.0098	0.0099
	post. std.	0.0090	0.0068	0.0090	0.0090	0.0090	0.0090
	ESS	35188	–	34896	35144	35544	34396
	ESS/hour CPU time	1682	–	1738	1481	1998	
β	post. mean	0.1685	0.1297	0.1682	0.1675	0.1682	0.1683
	post. std.	0.1727	0.1312	0.1726	0.1727	0.1728	0.1727
	ESS	37493	–	37689	37083	37407	37252
	ESS/hour CPU time	1792	–	1878	1563	2103	
σ_x	post. mean	0.4058	0.3046	0.4074	0.4080	0.4071	0.4074
	post. std.	0.0602	0.0570	0.0594	0.0634	0.0609	0.0616
	ESS	141	–	149	122	137	137
	ESS/hour CPU time	7	–	7	5	8	
γ	post. mean	1.1813	1.1744	1.1830	1.1831	1.1826	1.1828
	post. std.	0.0589	0.0711	0.0576	0.0609	0.0592	0.0593
	ESS	139	–	147	122	134	136
	ESS/hour CPU time	7	–	7	5	8	
σ_y	post. mean	0.0005	0.0009	0.0005	0.0005	0.0005	0.0005
	post. std.	2.3e-5	1.9e-5	2.2e-5	2.3e-5	2.3e-5	2.2e-5
	ESS	675	–	770	629	587	723
	ESS/hour CPU time	32	–	38	27	33	
x_1	post. mean	0.0954	0.0949	0.0954	0.0954	0.0954	0.0954
	post. std.	0.0005	0.0008	0.0005	0.0005	0.0005	0.0005
	ESS	36784	–	36900	37346	25507	39114
	ESS/hour CPU time	1759	–	1837	1574	1435	
$x_{T/2}$	post. mean	0.0925	0.0924	0.0925	0.0925	0.0925	0.0925
	post. std.	0.0005	0.0007	0.0005	0.0005	0.0005	0.0005
	ESS	37733	–	36712	37810	30781	39671
	ESS/hour CPU time	1804	–	1828	1594	1731	
x_T	post. mean	0.0608	0.0607	0.0608	0.0608	0.0608	0.0608
	post. std.	0.0005	0.0007	0.0005	0.0005	0.0005	0.0005
	ESS	37503	–	36689	37834	36953	39638
	ESS/hour CPU time	1792	–	1827	1595	2078	

NOTE: Results from the PG algorithms for the CEV interest rate model based on 50,000 PG iterations (discarding the first 10,000 draws) and $N = 30$ SMC particles. All reported statistics are sample averages computed from 10 independent replications of the PG algorithms under 10 different seeds. The ML estimates for the parameters are $(\alpha, \beta, \sigma_x, \gamma, \sigma_y)$ = (0.0097, 0.1656, 0.4250, 1.201, 0.0005).