Particle Gibbs for Likelihood-Free Inference of Stochastic Volatility Models

Zhaoran Hou[1] and Samuel W.K. Wong[1*]

[1*]Department of Statistics and Actuarial Science, University of Waterloo, 200 University Ave W, Waterloo, ON N2L 3G1, Canada.

*Corresponding author(s). E-mail(s): samuel.wong@uwaterloo.ca;
Contributing authors: zhaoran.hou@uwaterloo.ca;

Abstract

Stochastic volatility models (SVMs) are widely used in finance and econometrics for analyzing and interpreting volatility. Real financial data are often observed to have heavy tails, which violate a Gaussian assumption and may be better modeled using the stable distribution. However, the intractable density of the stable distribution hinders the use of common computational methods such as Markov chain Monte Carlo (MCMC) for parameter inference of SVMs. In this paper, we propose a new particle Gibbs sampler as a strategy to handle SVMs with intractable likelihoods in the approximate Bayesian computation (ABC) setting. The proposed sampler incorporates a conditional auxiliary particle filter, which can help mitigate the weight degeneracy often encountered when using ABC. Simulation studies demonstrate the efficacy of our sampler for inferring SVM parameters when compared to existing particle Gibbs samplers based on the conditional bootstrap filter, and for inferring both SVM and stable distribution parameters when compared to existing particle MCMC samplers. As a real data application, we apply the proposed sampler for fitting an SVM to S&P 500 Index time-series data during the 2008–2009 financial crisis.

Keywords: approximate Bayesian computation, financial time series, particle MCMC, sequential Monte Carlo

1 Introduction

In the analysis of financial time series, volatility is commonly used to quantify uncertainty or risk. Given a time series of observed prices $\{P_t\}_{t=0}^T$, for $t = 1, \ldots, T$ the *return* is defined as $r_t \equiv \log(P_t) - \log(P_{t-1})$ and the *volatility* (which is unobserved) is defined as $h_t \equiv Var(r_t|r_{1:t-1})$. In this paper, we consider the classic stochastic volatility model (SVM) (Jacquier et al., 1994, 2004; Taylor, 2008) that has been widely used in option pricing and portfolio management: for $t = 1, \ldots, T$,

$$\log(h_t) = \tau + \phi \log(h_{t-1}) + \sigma_h \epsilon_t, \quad r_t = \sqrt{h_t} \times Z_t \tag{1}$$

where $\log(h_t)$ is the log-volatility, Z_t and ϵ_t are independent noise terms, $\theta \equiv (\tau, \phi, \sigma_h^2)$ denotes the model parameters, and an initial distribution is assigned to h_0. Given the distributions of Z_t and ϵ_t, the goal is to infer θ from the time series of returns $r_{1:T}$.

The methods of inference available for Model (1) depend on the distributions chosen for Z_t and ϵ_t. When Z_t and ϵ_t are assumed to follow standard Gaussian distributions, the unobserved volatilities $h_{0:T}$ can be analytically integrated out to directly obtain the marginal likelihood of θ, and standard techniques can be applied, e.g., maximum likelihood estimation (MLE) in a frequentist approach (Fridman and Harris, 1998), and Markov chain Monte Carlo (MCMC) in a Bayesian approach (Jacquier et al., 1994). However, empirical studies suggest that a Gaussian assumption for Z_t may not adequately capture the heavy tails and skewness of financial time series (Engle and Patton, 2001). Figure 1 illustrates this phenomenon, which plots the daily returns from the Standard & Poor 500 (S&P 500) index for the period January 2008 to March 2009; the corresponding normal Q-Q plot indicates that the distribution of these returns has much heavier tails than a Gaussian. As a more flexible alternative, the so-called *stable distribution* (Mandelbrot, 1963; Nolan, 1997) can be adopted for Z_t instead (Lombardi and Calzolari, 2009; Vankov et al., 2019). Stable distributions can capture a wide range of heavy-tailedness, and also have appealing theoretical properties due to their role in the generalized central limit theorem (Nolan, 2020); further details are reviewed in Section 2.1. However, the stable distribution does not have a closed-form density function; hence, the resulting likelihood of r_t is analytically intractable and more specialized methodology is needed. Moreover, the parameters governing the heavy-tailedness and skewness of the stable distribution may also need to be estimated. This is our setting of interest, and this paper develops a more effective method to handle the SVM in Model (1) when Z_t follows a stable distribution.

The SVM in Model (1) takes the form of a state space model (SSM). Briefly, an SSM describes a dynamic system via a time series of unobserved variables (or *hidden states*) and observations generated conditional on those variables (Kitagawa, 1998). When the hidden states are a discrete-time Markov process (e.g., $h_{0:T}$ in the SVM), the SSM is specified via the transition probability of the hidden states and the likelihood of the observations. For Model (1), at time t we let $l_t(r_t \mid h_t)$ denote the likelihood density and $g_t(h_t \mid h_{t-1}, \theta)$ denote the transition density. When Z_t follows a stable distribution, the likelihood $l_t(r_t \mid h_t)$ is analytically intractable and standard MLE or MCMC methods cannot be applied. Instead, particle Markov chain Monte Carlo (PMCMC, Andrieu et al., 2010) is a general sampling approach that can be adapted for inference in this setting under a Bayesian framework. Originally proposed for inference of SSMs, PMCMC provides a class of algorithms that combine the features of MCMC and sequential Monte Carlo (SMC, Liu and West, 2001; Storvik, 2002; Carvalho et al.,

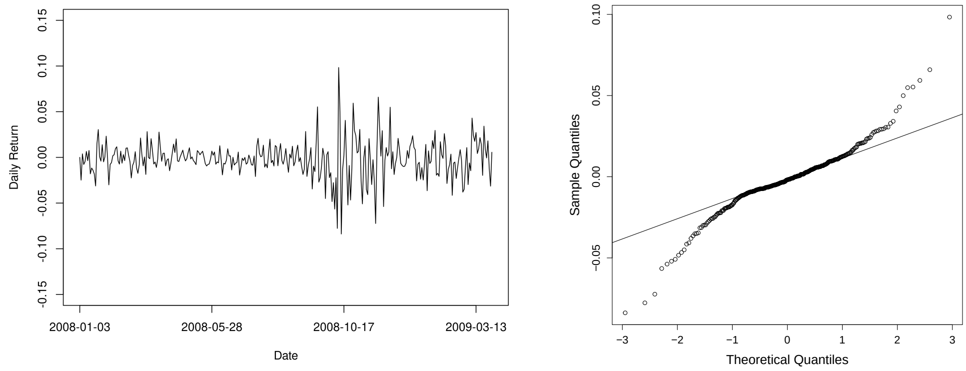

Fig. 1 The left panel presents the daily returns of the S&P 500 index from January 2008 to March 2009. The large fluctuations around October 2008 indicate the climax of the global financial crisis. The right panel presents a normal Q-Q plot of these daily returns, which indicates that the distribution of returns has much heavier tails than a Gaussian.

2010); in this context, SMC methods (also known as particle filters) are well-suited for drawing posterior samples of an SSM's hidden states.

To our best knowledge, PMCMC is the main type of approach for SVM parameter inference when it is not possible to analytically integrate out $h_{0:T}$ (Vankov et al., 2019), which includes our setting of interest. Let $p(\theta)$ denote the joint prior for θ. PMCMC then targets the joint posterior of θ and $h_{0:T}$ in Model (1), namely

$$p(\theta, h_{0:T} \mid r_{1:T}) \propto p(\theta) g_0(h_0 \mid \theta) \prod_{t=1}^{T} g_t(h_t \mid h_{t-1}, \theta) l_t(r_t \mid h_t),$$

if $l_t(r_t \mid h_t)$ can be evaluated. (The corresponding version of PMCMC that bypasses this likelihood evaluation is discussed in Section 2.) When closed-form conditional distributions of the parameters are available, a special case of PMCMC, known as *particle Gibbs*, can be implemented. As applied here, the basic strategy of particle Gibbs alternates between sampling from $p(\theta \mid h_{0:T}, r_{1:T})$ and $p(h_{0:T} \mid \theta, r_{1:T})$ at each iteration. With the help of conjugate priors, sampling from $p(\theta \mid h_{0:T}, r_{1:T})$ can be straightforward. Sampling from $p(h_{0:T} \mid \theta, r_{1:T})$ can potentially be handled by SMC as for an SSM; e.g., one might apply SMC and obtain a particle approximation of $p(h_{0:T} \mid \theta, r_{1:T})$, denoted by $\widehat{p}_{SMC}(h_{0:T} \mid \theta, r_{1:T})$. However, it is not valid to simply substitute sampling from $p(h_{0:T} \mid \theta, r_{1:T})$ with sampling from $\widehat{p}_{SMC}(h_{0:T} \mid \theta, r_{1:T})$ in particle Gibbs, because doing so does not admit the target distribution as invariant (Andrieu et al., 2010). To correctly embed SMC within a particle Gibbs sampler, the form of SMC known as *conditional SMC* (cSMC) should be implemented instead; we review cSMC algorithms in Section 2.2.

Approximate Bayesian computation (ABC) is a general technique that can be used to bypass the evaluation of an intractable or expensive likelihood, if one can directly simulate observations from the likelihood (Marin et al., 2012). Thus, to perform inference on Model (1) with an analytically intractable $l_t(r_t \mid h_t)$, ABC may be combined with cSMC. The basic idea is to introduce a sequence of auxiliary observations (that are sampled from the likelihood) and then assign weights according to the "distances" between the auxiliary observations and the actual observations; ABC-based methods are further reviewed in Section 2.3. It is straightforward to construct ABC versions of existing cSMC algorithms based on the bootstrap filter (BF, Gordon et al., 1993); however, in practice the particles they generate can tend to have many near-zero weights due to large "distances", i.e., the algorithms suffer from weight degeneracy. In the SMC literature, specific techniques to mitigate weight degeneracy include the SMC sampler with annealed importance sampling (Del Moral et al., 2006), weight tempering via lookahead strategies (Lin et al., 2013), and drawing multiple descendants per particle (Hou and Wong, 2024); however, as cSMC must be embedded within every iteration of particle Gibbs, these techniques would be computationally expensive to apply. In contrast, the auxiliary particle filter (APF, Pitt and Shephard, 1999) is a common alternative to the BF that can help reduce weight degeneracy, since the resampling step of the APF accounts for the one-step-ahead observation.

In this paper, we use the APF as a strategy to reduce weight degeneracy and improve parameter estimation in the particle Gibbs and ABC setting. In related work, Vankov et al. (2019) proposed an ABC-based APF that uses auxiliary observations to assign importance weights for PMCMC. However, that ABC-based APF has only been embedded within a particle marginal Metropolis-Hastings (PMMH) algorithm (Andrieu et al., 2010); a PMMH algorithm entails a relatively large particle size for reliable marginal likelihood estimation, which is computationally expensive. Therefore, as the main contribution of this paper, we propose to embed an ABC-based APF within a particle Gibbs sampler. We show that our proposed sampler satisfies the form of cSMC, and thus admits the target posterior distribution as invariant. We then perform inference on Model (1) when Z_t is assumed to follow the stable distribution. Simulation results indicate that our particle Gibbs sampler often outperforms existing ones based on the BF algorithm, given known stable distribution parameters. To handle the more general inference question where the stable distribution parameters are also unknown, we show how our proposed particle Gibbs sampler can be extended to incorporate a Metropolis-Hastings step. Simulation results also support the efficacy of our extended particle Gibbs sampler compared to the PMMH (Andrieu et al., 2010) and the single filter particle Metropolis-within-Gibbs (SF-PMwG) algorithms (Vankov et al., 2019) when estimating the SVM and stable distribution parameters together.

The paper is organized as follows: in Section 2, we review ABC and particle Gibbs methods, and present the proposed ABC-based particle Gibbs sampler with a conditional auxiliary particle filter (ABC-PG-cAPF). In Section 3, we illustrate the effectiveness of the proposed sampler for inference of Model (1) with the stable distribution. In Section 4, we present a real data application by fitting an SVM to S&P 500 daily returns during the 2008–2009 financial crisis. In Section 5, we briefly summarize the paper and its contributions and discuss some potential future directions.

2 Methodology

2.1 Model setup

Returning to the SVM in Model (1), we begin by reviewing relevant properties of the *stable distribution* (Nolan, 2020). We denote $Z \sim SD(\alpha, \beta, \gamma, \delta)$ if Z follows a stable distribution with parameters $\alpha \in (0, 2]$ for heavy-tailedness, $\beta \in [-1, 1]$ for skewness, $\gamma \in (0, \infty)$ for scale and $\delta \in (-\infty, \infty)$ for location. Z is defined via its characteristic function

$$\psi_Z(t) = \begin{cases} \exp\left(-\gamma^\alpha |t|^\alpha \left[1 + i\beta \operatorname{sign}(t) \tan\left(\frac{\pi\alpha}{2}\right) \left\{(\gamma|t|)^{1-\alpha} - 1\right\}\right] + i\delta t\right) & \text{if } \alpha \neq 1 \\ \exp\left(-\gamma|t| \left\{1 + \frac{2}{\pi} i\beta \operatorname{sign}(t) \log(\gamma|t|)\right\} + i\delta t\right) & \text{if } \alpha = 1. \end{cases}$$

Its corresponding density function, $f_Z(z) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \psi_Z(t)\exp(-izt)dt$, does not have an analytical form in general; some basic facts are that if $\alpha > 1$, $E[Z] = \delta - \beta\gamma\tan(\pi\alpha/2)$ (and undefined otherwise); if $\alpha < 2$, $Var(Z) = \infty$ (Mandelbrot, 1963; Nolan, 1997, 2020). Evaluation of $f_Z(z)$ is analytically impossible in general, and only expensive approximations are available via the fast Fourier transform (Mittnik et al., 1999), numerical integration (Nolan, 1997, 1999) and MCMC (Lemke et al., 2015); however, it is feasible to simulate realizations of Z according to $\psi_Z(t)$ based on the work of Kanter (1975) and Chambers et al. (1976). In this paper, we fix $\gamma = 1$ and $\delta = 0$ following Vankov et al. (2019) and use the R package *stabledist* (Wuertz et al., 2016) for generating random draws of Z.

Our model of interest is thus the SVM specified in (1) with $\epsilon_t \sim N(0, 1)$ and $Z_t \sim SD(\alpha, \beta, 1, 0)$ for $t = 1, \ldots, T$, all independent. We also set $h_0 \sim LN(\tau/(1-\phi), \sigma_h^2/(1-\phi^2))$ according to the stationarity of the log-volatility process, with LN denoting the log-normal distribution. The goal is to infer the parameters $\theta = (\tau, \phi, \sigma_h^2)$ and $\zeta = (\alpha, \beta)$ given a time series of returns $r_{1:T}$. To simplify exposition of the algorithms, we first focus on θ (treating ζ is known) in what follows; then, we return to the problem of jointly estimating (θ, ζ) in Section 2.5.

We take a Bayesian approach to inference and assign conjugate inverse-gamma (IG) and normal priors for θ: $\sigma_h^2 \sim IG(a_0, b_0)$ where a_0 and b_0 respectively denote the shape and rate of the inverse-gamma distribution, and $(\tau, \phi) \mid \sigma_h^2 \sim N(\boldsymbol{\mu}_0, \sigma_h^2 \boldsymbol{\Lambda}_0^{-1})$ with $|\phi| < 1$, i.e., the joint prior for θ is a truncated normal–inverse Gamma with hyperparameters $a_0, b_0, \boldsymbol{\mu}_0, \boldsymbol{\Lambda}_0$, which we denote as $NIG(a_0, b_0, \boldsymbol{\mu}_0, \boldsymbol{\Lambda}_0)$; the truncated support for ϕ ensures the required second-order stationarity of the log-volatility process (Jacquier et al., 2004). The conjugacy of these priors follows from the results of Bayesian linear regression, i.e., by defining

$$\mathbf{X} = \begin{bmatrix} 1 & \log(h_0) \\ \vdots & \vdots \\ 1 & \log(h_{T-1}) \end{bmatrix} \text{ and } \boldsymbol{y} = \begin{bmatrix} \log(h_1) \\ \vdots \\ \log(h_T) \end{bmatrix},$$

then $p(\theta \mid h_{0:T}, r_{1:T}) = p(\theta \mid h_{0:T})$ is a truncated $NIG(a_T, b_T, \boldsymbol{\mu}_T, \boldsymbol{\Lambda}_T)$ with $|\phi| < 1$, where

$$\boldsymbol{\Lambda}_T = \left(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \boldsymbol{\Lambda}_0\right), \quad \boldsymbol{\mu}_T = \boldsymbol{\Lambda}_T^{-1}\left(\boldsymbol{\Lambda}_0\boldsymbol{\mu}_0 + \mathbf{X}^{\mathrm{T}}\boldsymbol{y}\right),$$
$$a_T = a_0 + \frac{T}{2}, \quad b_T = b_0 + \frac{1}{2}\left(\mathbf{y}^{\mathrm{T}}\mathbf{y} + \boldsymbol{\mu}_0^{\mathrm{T}}\boldsymbol{\Lambda}_0\boldsymbol{\mu}_0 - \boldsymbol{\mu}_T^{\mathrm{T}}\boldsymbol{\Lambda}_T\boldsymbol{\mu}_T\right).$$

Thus, a Gibbs sampler update of θ only depends on these sufficient statistics. The choice of hyperparameters could be informed by previous empirical studies (Kim et al., 1998; Yang et al., 2018; Vankov et al., 2019), or set to resemble a flat (Gelman et al., 2014) or weakly informative prior (Jacquier et al., 1994, 2004).

2.2 Particle Gibbs for the SVM with tractable likelihoods

In the SVM context, a particle Gibbs sampler alternates between sampling from $p(\theta \mid h_{0:T}, r_{1:T})$ and $p(h_{0:T} \mid \theta, r_{1:T})$. As described in the Introduction, sampling from $p(h_{0:T} \mid \theta, r_{1:T})$ requires the use of conditional SMC algorithms. The basic idea of cSMC is to take an input trajectory (i.e., a sample for $h_{0:T}$) as a reference and produce an output trajectory via SMC-style propagation; furthermore, cSMC accounts for all random variables generated during propagation via an extended target distribution of higher dimension (Chopin and Singh, 2015). The steps for a sweep of particle Gibbs then consist of (i) updating the parameters based on the input trajectory, (ii) generating new trajectories based on the updated parameters and input trajectory, and (iii) selecting an output trajectory as input for the next iteration (Andrieu et al., 2010). The use of cSMC in particle Gibbs guarantees the target distribution is admitted as the invariant density. Here, we briefly review two existing cSMC algorithms that are applicable within particle Gibbs when the likelihood $l_t(r_t \mid h_t)$ can be computed.

The first is the conditional bootstrap filter (cBF, Andrieu et al., 2010) as summarized in Algorithm 1. A key feature is that it preserves the input trajectory $h^*_{0:T}$ throughout propagation and resampling; holding $h^*_{0:T}$ intact, $N-1$ new trajectories are generated "conditional on" $h^*_{0:T}$ in SMC fashion; finally one of the N trajectories (i.e., among the input trajectory and the $N-1$ generated trajectories) is selected to be the new input trajectory for the next iteration. As shown in Algorithm 1, for any particle $h_t^{(n)}$ with $n \in \{1, \ldots, N\}$ simulated at step t, we denote the index of the ancestor particle of $h_t^{(n)}$ by $a_{t-1}^{(n)}$, i.e., $h_t^{(n)}$ is propagated from $h_{t-1}^{(a_{t-1}^{(n)})}$. For simplicity, let $A_{t,t}^{(n)} = n$, $A_{t-1,t}^{(n)} = a_{t-1}^{(n)}$ and $A_{t-l,t}^{(n)} = a_{t-l}^{(A_{t-l+1,t}^{(n)})}$ for $2 \leq l \leq t$; then the n-th trajectory can be written as $h_{0:T}^{(n)} = (h_0^{(A_{0,T}^{(n)})}, h_1^{(A_{1,T}^{(n)})}, \ldots, h_T^{(A_{T,T}^{(n)})})$ after $t = T$ propagation and resampling steps.

The second is the conditional bootstrap filter with ancestor sampling (cBFAS, Lindsten et al., 2014) as summarized in Algorithm 2. The cBF keeps the input trajectory $h^*_{0:T}$ intact, i.e., $A_{t,T}^{(N)} = N$, and preserved throughout for all $t \in \{0, \ldots, T-1\}$, so the early parts of generated trajectories may tend to closely resemble (or be identical to) the input trajectory, which when too extreme is known as path degeneracy. The key idea of cBFAS is to stochastically perturb the input trajectory by breaking it into pieces via ancestor sampling. Ancestor sampling, as presented in Algorithm 2, takes

Algorithm 1: Conditional Bootstrap Filter

input: observations $r_{1:T}$, particle size N, input trajectory $h^*_{0:T}$, transition density g_t, likelihood l_t;
draw $h_0^{(n)} \sim g_0(h_0)$ for $n = 1, \ldots, N-1$ and set $h_0^{(N)} = h_0^*$;
$w_0^{(n)} = \frac{1}{N}$ for $n = 1, \ldots, N$;
for t *in 1:T* **do**
  draw index $a_{t-1}^{(n)}$ from $\{(n, w_{t-1}^{(n)})\}_{n=1}^N$ for $n = 1, \ldots, N-1$;
  set index $a_{t-1}^{(N)} = N$;
  draw $h_t^{(n)} \sim g_t(h_t \mid h_{t-1}^{(a_{t-1}^{(n)})})$ for $n = 1, \ldots, N-1$ and set $h_t^{(N)} = h_t^*$;
  $w_t^{(n)} = l_t(r_t \mid h_t^{(n)})$ for $n = 1, \ldots, N$;
end
draw index b from $\{(n, w_T^{(n)})\}_{n=1}^N$;
return $h_{0:T}^{(b)}$;

Algorithm 2: Conditional Bootstrap Filter with Ancestor Sampling

input: observations $r_{1:T}$, particle size N, input trajectory $h^*_{0:T}$, transition density g_t, likelihood l_t;
draw $h_0^{(n)} \sim g_0(h_0)$ for $n = 1, \ldots, N-1$ and set $h_0^{(N)} = h_0^*$;
$w_0^{(n)} = \frac{1}{N}$ for $n = 1, \ldots, N$;
for t *in 1:T* **do**
  draw index $a_{t-1}^{(n)}$ from $\{(n, w_{t-1}^{(n)})\}_{n=1}^N$ for $n = 1, \ldots, N-1$;
  draw index $a_{t-1}^{(N)}$ from $\{(n, w_{t-1}^{(n)} g_t(h_t^* \mid h_{t-1}^{(n)}))\}_{n=1}^N$;
  draw $h_t^{(n)} \sim g_t(h_t \mid h_{t-1}^{(a_{t-1}^{(n)})})$ for $n = 1, \ldots, N-1$ and set $h_t^{(N)} = h_t^*$;
  $w_t^{(n)} = l_t(r_t \mid h_t^{(n)})$ for $n = 1, \ldots, N$;
end
draw index b from $\{(n, w_T^{(n)})\}_{n=1}^N$;
return $h_{0:T}^{(b)}$;

$A_{t,T}^{(N)}$ to be stochastic, so the input trajectory can be partially replaced by other generated trajectories at each step t. Consequently, the input trajectory interacts much more with the other trajectories (Svensson et al., 2015), which can help cBFAS mitigate path degeneracy while maintaining the target distribution as invariant (Lindsten et al., 2014; Svensson et al., 2015).

2.3 Review of ABC methods and ABC-SMC methods

To briefly review ABC methods, consider observations r and parameters θ where the likelihood $l(r \mid \theta)$ is intractable and expensive to approximate. Consequently, the posterior $p(\theta \mid r) \propto p(\theta) l(r \mid \theta)$ is also intractable. In this setting, ABC can be used for inference of θ if sampling from $l(\cdot \mid \theta)$ is straightforward. The basic idea of ABC methods is to construct an approximation to the posterior with the help of auxiliary observations (denoted by u) and an ABC kernel (denoted by $K_\epsilon(r \mid u)$), which may be interpreted from the perspective of model error (Wilkinson, 2013). The simplest example of an ABC method is a likelihood-free rejection sampling algorithm with a uniform kernel (i.e., $K_\epsilon(r \mid u) \propto \mathbb{1}_{u-\epsilon<r<u+\epsilon}$ for a chosen $\epsilon > 0$) that implements the following steps: (1) sample a candidate θ^* from the prior $p(\theta)$; (2) sample a realization u from $l(u \mid \theta^*)$; (3) accept (θ^*, u) if $|r - u| < \epsilon$. Then accepted samples of (θ^*, u) follow the density defined by $p_\epsilon(\theta, u \mid r) \propto p(\theta)l(u \mid \theta)K_\epsilon(r \mid u)$, and the marginal density $p_\epsilon(\theta \mid r)$ can be viewed as the ABC approximation of $p(\theta \mid r)$. In practice, Gaussian kernels (i.e., $K_\epsilon(r \mid u) \propto \exp[-(r-u)^2/(2\epsilon^2)]$) are more commonly used than uniform ones (Nakagome et al., 2013; Park et al., 2016; Beaumont, 2019). It is clear that $p_\epsilon(\theta \mid r) \xrightarrow{\text{d}} p(\theta \mid r)$ as $\epsilon \to 0$ and $p_\epsilon(\theta \mid r) \xrightarrow{\text{d}} p(\theta)$ as $\epsilon \to \infty$. Therefore, an ABC kernel with a small ϵ can lead to many near-zero weights of generated candidates (θ^*, u) (or more rejected samples), while a larger ϵ can lead to more uniform weights (or more accepted samples); however, the accuracy of the ABC approximation will decrease as a tradeoff.

For the SVM with intractable likelihoods, we can similarly construct the ABC approximation $p_\epsilon(h_{0:T}, u_{1:T} \mid r_{1:T}, \theta)$ of the corresponding extended distribution and sample from it using an SMC algorithm; this is known as ABC-SMC (Peters et al., 2012). Thus, the target distribution of ABC-SMC involves the auxiliary observations $u_{1:T}$ for evaluating importance weights and has the form

$$p_\epsilon(h_{0:T}, u_{1:T} \mid r_{1:T}, \theta) \propto g_0(h_0 \mid \theta) \prod_{t=1}^{T} g_t(h_t \mid h_{t-1}, \theta) K_\epsilon(r_t \mid u_t) l_t(u_t \mid h_t). \tag{2}$$

To sample from (2), Vankov et al. (2019) proposed an ABC-based APF, which is summarized in Algorithm 3 and applicable within a particle Metropolis-Hastings algorithm. The computation of the tempered weights $\{\widetilde{w}_{t-1}\}_{n=1}^{N}$, i.e., the adjusted importance weights that incorporate the one-step-ahead observation r_t, requires an intractable integration, namely

$$\widetilde{w}_{t-1}^{(n)} = w_{t-1}^{(n)} p(r_t \mid h_{t-1}^{(n)}) = w_{t-1}^{(n)} \int\int K_\epsilon(r_t \mid u_t) l_t(u_t \mid h_t) g_t(h_t \mid h_{t-1}^{(n)}) dh_t du_t \tag{3}$$

where the double integral does not have a closed form; Vankov et al. (2019) suggest approximating it with a more heavy-tailed distribution such as a t-class distribution.

Within a particle Gibbs sampler, however, Algorithm 3 cannot be directly used (recall that particle Gibbs requires a cSMC setup to admit the target distribution as invariant). Thus, in the following we propose a new ABC-based APF, which we call the ABC-based conditional auxiliary particle filter (ABC-cAPF), that can be embedded within a particle Gibbs sampler.

Algorithm 3: ABC-based Auxiliary Particle Filter

input: observations $r_{1:T}$, particle size N, transition density g_t, likelihood l_t;
draw $h_0^{(n)} \sim g_0(h_0)$ for $n = 1, \dots, N$;
set $w_0^{(n)} = \frac{1}{N}$ for $n = 1, \dots, N$;
for t *in 1:T* **do**
  $\widetilde{w}_{t-1}^{(n)} = w_{t-1}^{(n)} \int\int K_\epsilon(r_t \mid u_t) l_t(u_t \mid h_t) g_t(h_t \mid h_{t-1}^{(n)}) dh_t du_t$ for $n = 1, \dots, N$;
  draw index $a_{t-1}^{(n)}$ from $\{(n, \widetilde{w}_{t-1}^{(n)})\}_{n=1}^N$;
  draw $h_t^{(n)} \sim g_t(h_t \mid h_{t-1}^{(a_{t-1}^{(n)})})$ for $n = 1, \dots, N$;
  draw $u_t^{(n)} \sim l_t(u_t \mid h_t^{(n)})$ for $n = 1, \dots, N$;
  $w_t^{(n)} = \frac{w_{t-1}^{(a_{t-1}^{(n)})}}{\widetilde{w}_{t-1}^{(a_{t-1}^{(n)})}} K_\epsilon(r_t \mid u_t^{(n)}) \Big/ \sum_{n=1}^N \frac{w_{t-1}^{(a_{t-1}^{(n)})}}{\widetilde{w}_{t-1}^{(a_{t-1}^{(n)})}} K_\epsilon(r_t \mid u_t^{(n)})$ for $n = 1, \dots, N$;
end
draw index b from $\{(n, w_T^{(n)})\}_{n=1}^N$;
return $h_{0:T}^{(b)}$;

2.4 Likelihood-free ABC-based cSMC for the SVM

Now we consider the SVM with stable distribution as presented in Section 2.1. In the ABC setting with a chosen kernel K_ϵ, Model (1) can be re-expressed as

$$\log(h_t) = \tau + \phi \log(h_{t-1}) + \sigma_h \epsilon_t, \quad u_t = \sqrt{h_t} \times Z_t, \quad r_t \sim K_\epsilon(\cdot \mid u_t) \tag{4}$$

with $\epsilon_t \sim N(0,1)$ and $Z_t \sim SD(\alpha, \beta, 1, 0)$ all independent for $t = 1 \dots, T$, and $h_0 \sim LN(\tau/(1-\phi), \sigma_h^2/(1-\phi^2))$. Here, each h_t, $t = 1 \dots, T$ corresponds to an auxiliary observation u_t for importance weight calculation according to K_ϵ. The extended posterior distribution of interest is then

$$p_\epsilon(\theta, h_{0:T}, u_{1:T} \mid r_{1:T}) \propto p(\theta) g_0(h_0 \mid \theta) \prod_{t=1}^T g_t(h_t \mid h_{t-1}, \theta) K_\epsilon(r_t \mid u_t) l_t(u_t \mid h_t), \tag{5}$$

and the goal is to sample from (5) in a likelihood-free manner, i.e., without computing l_t. We shall focus on the particle Gibbs case, and develop ABC-based algorithms to sample from (5) that alternate between sampling from $p(\theta \mid h_{0:T}, u_{1:T}, r_{1:T}) = p(\theta \mid h_{0:T})$ and $p_\epsilon(h_{0:T}, u_{1:T} \mid r_{1:T}, \theta)$.

The cBF and cBFAS algorithms can be applied in the ABC setting with slight modifications: in the step where we draw each $h_t^{(n)} \sim g_t(h_t \mid h_{t-1}^{(a_{t-1}^{(n)})})$, we also draw an auxiliary observation $u_t^{(n)}$ from $l_t(u_t \mid h_t^{(n)})$ and then assign the importance weight $w_t^{(n)} = K_\epsilon(r_t \mid u_t^{(n)})$ for the particle $(h_t^{(n)}, u_t^{(n)})$. We shall call these algorithms the ABC-based conditional bootstrap filter (ABC-cBF) and ABC-based conditional bootstrap filter with ancestor sampling (ABC-cBFAS), respectively. However, these two algorithms can encounter severe weight degeneracy in practice, if we choose a small ϵ in the ABC kernel to obtain an accurate approximation of the true target distribution. Thus, in the following we also propose a novel ABC-based cSMC algorithm using the auxiliary particle filter as the building block.

We shall call this third algorithm the ABC-based conditional auxiliary particle filter (ABC-cAPF), as presented in Algorithm 4. To initialize the ABC-cAPF at $t = 0$, $N-1$ particles, $\{h_0^{(n)}\}_{n=1}^{N-1}$, are sampled from the log-normal density $g_0(h_0|\theta)$ and $h_0^{(N)}$ is assigned the initial log-volatility of the input trajectory. Then after the $(t-1)$-th propagation step ($t = 1, \ldots, T$), the N-th particle is set to be the input trajectory $h^*_{0:t-1}$ and the remaining $N-1$ particles $\{(h_{0:t-1}^{(n)}, u_{1:t-1}^{(n)})\}_{n=1}^{N-1}$ will have been generated with weights that satisfy

$$w_{t-1}^{(n)} \propto g_0(h_0^{(A_{0,t-1}^{(n)})}) \prod_{s=1}^{t-1} K_\epsilon(r_s \mid u_s^{(A_{s,t-1}^{(n)})}) l_s(u_s^{(A_{s,t-1}^{(n)})} \mid h_s^{(A_{s,t-1}^{(n)})}) g_s(h_s^{(A_{s,t-1}^{(n)})} \mid h_{s-1}^{(A_{s-1,t-1}^{(n)})})$$

for $n = 1, \ldots, N-1$. Following the concept of properly weighted particles in SMC (Liu, 2001), this weight ensures that the set of $N-1$ generated particles is properly weighted with respect to $p_\epsilon(h_{0:t-1}, u_{1:t-1} \mid r_{1:t-1}, \theta)$. As an important feature of the APF (Pitt and Shephard, 1999) when propagating the particles from $t-1$ to t, the tempered weights $\widetilde{w}_{t-1}^{(n)}$ defined in (3) are computed and utilized. In the following, we propose a different way of approximating the double integral in (3) that more directly incorporates h_{t-1} into the variability of r_t.

Specifically, since $r_t = \sqrt{h_t} \times Z_t$, using a standard Cauchy random variable Z_{Cauchy} to approximate Z_t leads to the relation $\log(r_t^2) = \log(h_t) + \log(Z^2_{\text{Cauchy}})$, where $E\{\log(r_t^2) \mid h_{t-1}\} = \tau + \phi \log(h_{t-1})$ and $Var\{\log(r_t^2) \mid h_{t-1}\} = \sigma_h^2 + \pi^2$. Then by approximating $\log(r_t^2)$ as a linear combination of $\log(h_{t-1})$ and $\log(Z^2_{\text{Cauchy}})$ to match this conditional mean and variance, we obtain $\log(Z^2_{\text{Cauchy}}) = \sqrt{\frac{\pi^2}{\sigma_h^2+\pi^2}}\{\log(r_t^2) - \tau - \phi \log(h_{t-1})\}$; in practice, $\frac{\pi^2}{\sigma_h^2+\pi^2} \approx 1$ which we set as 1 for simplicity. This leads to an approximation $\tilde{p}(r_t \mid h_{t-1})$ of $p(r_t \mid h_{t-1})$ via a scaled Cauchy distribution with density

$$\tilde{p}(r_t \mid h_{t-1}) = \frac{\exp\{-0.5(\tau + \phi \log(h_{t-1}))\}}{\pi\left[1 + r_t^2 \exp\{-(\tau + \phi \log(h_{t-1}))\}\right]}.$$

Note that the quality of the integral approximation in (3) does not influence the validity of ABC-cAPF; however, the tempered weights should cover the high-density regions of the true importance weights for a more efficient algorithm.

For $t = 1, \ldots, T$, resampling and propagation are implemented with the tempered weights, but otherwise similar to the cBF: (i) resample the first $N-1$ particles from $\{(h_{0:t-1}^{(n)}, u_{1:t-1}^{(n)})\}_{n=1}^{N}$ proportional to the tempered weights $\{\widetilde{w}_{t-1}^{(n)}\}_{n=1}^{N}$ while leaving the N-th particle intact as the input trajectory; (ii) propagate the resampled particles as in the cBF; (iii) compute the importance weights of the propagated particles so that cAPF targets the same density as cBF; specifically, we set the weight of the N-th

Algorithm 4: ABC-based Conditional Auxiliary Particle Filter

input: observations $r_{1:T}$, particle size N, input trajectory $h^*_{0:T}$, transition density g_t, likelihood l_t;
draw $h_0^{(n)} \sim g_0(h_0)$ for $n = 1, \ldots, N-1$ and set $h_0^{(N)} = h_0^*$;
$w_0^{(n)} = \frac{1}{N}$ for $n = 1, \ldots, N$;
for *t in 1:T* **do**
  $\widetilde{w}_{t-1}^{(n)} = w_{t-1}^{(n)} \tilde{p}(r_t \mid h_{t-1}^{(n)})$ for $n = 1, \ldots, N$;
  draw index $a_{t-1}^{(n)}$ from $\{(n, \widetilde{w}_{t-1}^{(n)})\}_{n=1}^N$ for $n = 1, \ldots, N-1$ and set index $a_{t-1}^{(N)} = N$;
  draw $h_t^{(n)} \sim g_t(h_t \mid h_{t-1}^{(a_{t-1}^{(n)})})$ for $n = 1, \ldots, N-1$ and set $h_t^{(N)} = h_t^*$;
  draw $u_t^{(n)} \sim l_t(u_t \mid h_t^{(n)})$ for $n = 1, \ldots, N$;
  $w_t^{(n)} = (N-1) \times \frac{w_{t-1}^{(a_{t-1}^{(n)})}}{\widetilde{w}_{t-1}^{(a_{t-1}^{(n)})}} \Big/ \sum_{n=1}^{N-1} \frac{w_{t-1}^{(a_{t-1}^{(n)})}}{\widetilde{w}_{t-1}^{(a_{t-1}^{(n)})}} \times K_\epsilon(r_t \mid u_t^{(n)})$ for $n = 1, \ldots, N-1$ and $w_t^{(N)} = K_\epsilon(r_t \mid u_t^{(n)})$;
  normalize weights $\{w_t^{(n)}\}_{n=1}^N$;
end
draw index b from $\{(n, w_T^{(n)})\}_{n=1}^N$;
return $h_{0:T}^{(b)}$;

Algorithm 5: ABC-based Particle Gibbs with the Conditional Auxiliary Particle Filter (ABC-PG-cAPF)

input: observations $r_{1:T}$, particle size N, burn-in size I, sample size J;
sample $\theta[0] = (\tau[0], \phi[0], \sigma_h^2[0])$ from $NIG(a_0, b_0, \boldsymbol{\mu}_0, \boldsymbol{\Lambda}_0)$;
sample $h_t^*[0]$ from $LN(\tau[0]/(1-\phi[0]), \sigma_h^2[0]/(1-\phi[0]^2))$ for all t;
for *k in 1:(I + J)* **do**
  run Algorithm 4 with $\theta[k-1]$ and $h^*_{0:T}[k-1]$ and output $h^*_{0:T}[k]$;
  update the sufficient statistics with $h^*_{0:T}[k]$;
  sample $\theta[k]$ from the posterior $NIG(a_T, b_T, \boldsymbol{\mu}_T, \boldsymbol{\Lambda}_T)$;
end
return $\theta[(I+1):J]$;

particle to be the same as that in cBF, and then re-scale and normalize the weights of the other $N-1$ particles accordingly.

Each of the three algorithms, i.e., ABC-cBF, ABC-cBFAS, and ABC-cAPF, can be combined with a Gibbs sampler for the parameters $p(\theta \mid h_{0:T}, u_{1:T}, r_{1:T})$ to construct particle Gibbs samplers for (5). We call these samplers the ABC-based particle Gibbs with ABC-cBF (ABC-PG-cBF), ABC-based particle Gibbs with ABC-cBFAS (ABC-PG-cBFAS), and ABC-based particle Gibbs with the conditional auxiliary particle filter (ABC-PG-cAPF), respectively. Algorithm 5 shows how the ABC-cAPF is embedded within the particle Gibbs sampler: each Gibbs sweep uses the ABC-cAPF to draw an output trajectory for $h_{0:T}$ (given the current draw of θ and the input trajectory), and then draws an updated θ from its closed-form NIG conditional posterior (given the output trajectory, which becomes the input trajectory for the next iteration). The ABC-PG-cBF and ABC-PG-cBFAS samplers are constructed analogously. To initialize the particle Gibbs sampler, we draw parameters from $p(\theta)$ and an input trajectory $h_{0:T}^*$ from the corresponding log-normal distribution given those parameters.

Proposition 1. *The proposed particle Gibbs sampler (ABC-PG-cAPF in Algorithm 5) admits the target distribution, i.e., the posterior distribution $p_\epsilon(h_{0:T}, u_{1:T}, \theta \mid r_{1:T})$ in (5), as the invariant density under some mild assumptions.*

Proof. See Section S1 of the Supplementary Material. □

2.5 Extended ABC-PG-cAPF for estimating SVM and stable distribution parameters

We now turn to the more general inference problem of estimating both the SVM parameters $\theta = (\tau, \phi, \sigma_h^2)$ and stable distribution parameters $\zeta = (\alpha, \beta)$, as considered in Vankov et al. (2019). The proposed ABC-PG-cAPF can be extended for this purpose; such an extension is intuitively straightforward by appending ζ so that the parameter vector becomes (θ, ζ). However, as ζ is associated with the intractable stable distribution, no conjugacy is available for ζ and its Gibbs update needs to be handled via an ABC-based Metropolis-Hastings (MH) kernel. To achieve this, each sweep of particle Gibbs now consists of three steps: (i) update $h_{0:T}$ via ABC-cAPF conditional on θ and ζ (via Algorithm 4); (ii) update θ conditional on $h_{0:T}$ (via the NIG posterior, as shown in Algorithm 5); (iii) update ζ conditional on $h_{0:T}$ using an ABC-based MH kernel. The overall steps of this extended ABC-PG-cAPF are summarized in Algorithm 7; next, we consider the details of step (iii).

Let $\pi_\zeta(\cdot)$ denote the prior for ζ, then the Gibbs update for ζ targets $p(\zeta \mid r_{1:T}, h_{0:T}^*, \theta^*)$, where $h_{0:T}^*$ and θ^* denote the current draws of $h_{0:T}$ and θ, respectively. Since ζ and θ are conditionally independent given $h_{0:T}$, we have

$$\begin{aligned} p(\zeta \mid r_{1:T}, h_{0:T}^*, \theta^*) &= p(\zeta \mid r_{1:T}, h_{0:T}^*) \\ &\propto p(r_{1:T} \mid \zeta, h_{0:T}^*) p(\zeta \mid h_{0:T}^*) \\ &= p(r_{1:T} \mid \zeta, h_{0:T}^*) \pi_\zeta(\zeta) \end{aligned}$$

where the last equality follows from the prior independence of ζ and $h_{0:T}$. Using the conditional independence of the observations, we obtain

$$p(\zeta \mid r_{1:T}, h_{0:T}^*, \theta^*) \propto \pi_\zeta(\zeta) \prod_{t=1}^{T} l_t(r_t \mid \zeta, h_t^*) = \pi_\zeta(\zeta) \prod_{t=1}^{T} l_t^*(r_t^* \mid \zeta), \tag{6}$$

where $r_t^* = r_t / \sqrt{h_t^*}$ and $l_t^*(r_t^* \mid \zeta)$ is the density of the stable distribution with parameter ζ. As seen from Model (1), $\{r_t^*\}_{t=1}^T$ are T independent and identically distributed (IID) stable variables with parameter ζ. Since $l_t^*(r_t^* \mid \zeta)$ is also intractable, we employ an ABC approximation to (6) via

$$p_{\epsilon_\zeta}(\zeta, u_{1:T}^* \mid r_{1:T}, h_{0:T}^*, \theta^*) \propto \pi_\zeta(\zeta) K_{\epsilon_\zeta} \left\{\mathcal{H}(r_{1:T}^*) \mid \mathcal{H}(u_{1:T}^*)\right\} \prod_{t=1}^{T} l_t^*(u_t^* \mid \zeta),$$

where the auxiliary variables $u_{1:T}^*$ are generated from $\prod_{t=1}^{T} l_t^*(u_t^* \mid \zeta)$, K_{ϵ_ζ} is a chosen ABC kernel with tolerance ϵ_ζ, and $\mathcal{H}$ is a set of summary statistics. A MH kernel that targets this ABC posterior is therefore given by Algorithm 6, where $\mathcal{Q}(\cdot \mid \cdot)$ is a proposal distribution for updating ζ.

The summary statistics $\mathcal{H}$ should be chosen to distinguish between stable distribution samples drawn with different ζ. Following the work of McCulloch (1986), we use the set of three statistics defined by $\mathcal{H}(u_{1:T}^*) = \left(\frac{Q_{95\%} - Q_{5\%}}{Q_{75\%} - Q_{25\%}}, \frac{Q_{95\%} + Q_{5\%} - 2Q_{50\%}}{Q_{95\%} - Q_{5\%}}, Q_{50\%}\right)$, where Q_q is the empirical q-th quantile of $u_{1:T}^*$. The first two are location-invariant statistics proposed by McCulloch (1986) in developing simple estimators for the stable distribution parameters (α, β) that are valid for $0.5 < \alpha \leq 2$; since α is much larger than 0.5 for financial data in practice (Kabašinskas et al., 2009), we may impose $\alpha > 0.5$ as a reasonable restriction via the prior. We include the median as an additional statistic that further informs the skewness of the distribution. As in Algorithm 4, we adopt a Gaussian ABC kernel for K_{ϵ_ζ}, i.e., $K_{\epsilon_\zeta}\left\{\mathcal{H}(r_{1:T}^*) \mid \mathcal{H}(u_{1:T}^*)\right\} \propto \exp\left\{-\left\|\mathcal{H}(r_{1:T}^*) - \mathcal{H}(u_{1:T}^*)\right\|_2^2 / (2\epsilon_\zeta^2)\right\}$, where $\|\cdot\|_2$ denotes the l^2-norm. Note that the ABC tolerance ϵ_ζ in Algorithm 6 is distinct from ϵ in Algorithm 4; in Algorithm 6, ϵ_ζ should be tailored to the scale of the summary statistics $\mathcal{H}$ to yield an acceptable approximation. In our experience, a reasonable choice is $\epsilon_\zeta = 0.05$ for the $\mathcal{H}$ selected above.

This extended ABC-PG-cAPF algorithm has notable differences compared to the SF-PMwG algorithm proposed in Vankov et al. (2019). SF-PMwG provides a solution to the sampling problem by jointly updating $(\zeta, \sigma_h^2, h_{0:T})$ via a MH step, followed by Gibbs steps to update τ and ϕ. When carrying out the MH step, SF-PMwG proposes a trajectory $h_{0:T}$ and estimates the marginal likelihood of the observations via ABC-APF (Algorithm 3). An estimate of the marginal likelihood is required for computing the MH acceptance probability in SF-PMwG, and a relatively large particle size N is needed for a reliable estimate, which is computationally expensive. In contrast, the extended ABC-PG-cAPF algorithm handles the same model in a different manner: it updates θ via a closed-form Gibbs step, $h_{0:T}$ via the ABC-cAPF, and ζ via a MH step. The MH step for ζ within our algorithm does not require running SMC; only a simple ABC approximation is needed, because our MH update targets the distribution of ζ conditional on θ and $h_{0:T}$. Therefore, for each complete Gibbs sweep of the posterior in the extended ABC-PG-cAPF, only one pass of the ABC-cAPF is needed, which is significantly cheaper. The ABC-cAPF can be run with a much smaller particle size than ABC-APF, since ABC-cAPF only updates the trajectory $h_{0:T}$ and does not estimate the marginal likelihood.

Algorithm 6: Metropolis-Hastings Kernel for Updating ζ

input: observations $r_{1:T}$, ζ^*, auxiliary variables $u^*_{1:T}$, input trajectory $h^*_{0:T}$, prior π_ζ, summary statistics $\mathcal{H}$, proposal kernel $\mathcal{Q}$, ABC Kernel K_{ϵ_ζ};
initialization: $r^*_{1:T} = (r_1/\sqrt{h^*_1}, \ldots, r_T/\sqrt{h^*_T})$;
sample a proposal ζ^{**} from $\mathcal{Q}(\zeta \mid \zeta^*)$;
draw $u^{**}_{1:T} \sim \prod_{t=1}^T l^*_t(u^*_t \mid \zeta^{**})$;
compute $R = \min\left\{1, \frac{K_{\epsilon_\zeta}\{\mathcal{H}(r^*_{1:T})|\mathcal{H}(u^{**}_{1:T})\}\pi_\zeta(\zeta^{**})Q(\zeta^*|\zeta^{**})}{K_{\epsilon_\zeta}\{\mathcal{H}(r^*_{1:T})|\mathcal{H}(u^*_{1:T})\}\pi_\zeta(\zeta^*)Q(\zeta^{**}|\zeta^*)}\right\}$;
if $U \sim Unif[0,1] < R$ **then**
  return $(\zeta^{**}, u^{**}_{1:T})$;
end
else
  return $(\zeta^*, u^*_{1:T})$;
end

Algorithm 7: Extended ABC-PG-cAPF for (θ, ζ)

input: observations $r_{1:T}$, particle size N, burn-in size I, sample size J;
sample $\theta[0] = (\tau[0], \phi[0], \sigma^2_h[0])$ from $NIG(a_0, b_0, \boldsymbol{\mu}_0, \boldsymbol{\Lambda}_0)$;
sample $h^*_t[0]$ from $LN(\tau[0]/(1-\phi[0]), \sigma^2_h[0]/(1-\phi[0]^2))$ for all t;
sample $\zeta[0]$ from $\pi_\zeta(\zeta)$ and draw $u^*_{1:T}[0] \sim \prod_{t=1}^T l^*_t(u^*_t \mid \zeta[0])$;
for *k in 1:(I + J)* **do**
  run Algorithm 4 with $\theta[k-1]$, $\zeta[k-1]$ and $h^*_{0:T}[k-1]$ and output $h^*_{0:T}[k]$;
  update the sufficient statistics for θ with $h^*_{0:T}[k]$ and sample $\theta[k]$ from the posterior $NIG(a_T, b_T, \boldsymbol{\mu}_T, \boldsymbol{\Lambda}_T)$;
  run Algorithm 6 with $\zeta[k-1]$, $u^*_{1:T}[k-1]$ and $h^*_{0:T}[k]$ and output $\zeta[k]$, $u^*_{1:T}[k]$;
end
return $\theta[(I+1):J], \zeta[(I+1):J]$;

3 Simulation study

In this section, we implement two experiments. In Section 3.1, we assess the efficacy of the proposed ABC-cAPF sampler by comparing it with the other ABC-based cSMC methods for estimating θ only (with ζ known), following the settings of Jacquier et al. (1994). In Section 3.2, we compare the extended ABC-PG-cAPF (Algorithm 7) with the SF-PMwG (Vankov et al., 2019) and PMMH (Andrieu et al., 2010) algorithms for estimating the SVM and stable distribution parameters (θ, ζ) together.

3.1 Comparison of cSMC particle Gibbs samplers with a known stable distribution

We consider the model described in Equation (4) with a second-order stationary log-volatility process. Since h_t conditional on h_{t-1} follows a log-normal distribution, we have $E(h_t) = \exp\left\{\frac{\tau}{1-\phi} + \frac{\sigma_h^2}{2(1-\phi^2)}\right\}$ and $Var(h_t) = E^2(h_t)\left\{\exp\left(\frac{\sigma_h^2}{1-\phi^2}\right) - 1\right\}$. Therefore, the (squared) coefficient of variation, CV $= Var(h_t)/E^2(h_t)$, can be written as CV $= \exp\left(\frac{\sigma_h^2}{1-\phi^2}\right) - 1$. Practical ranges of ϕ that have been suggested from empirical analyses are $\phi \in [0.9, 0.98]$, or more loosely, $\phi \in [0.8, 0.995]$ (Jacquier et al., 1994); following their simulation study we take $\phi \in \{0.9, 0.95, 0.98\}$, CV $\in \{0.1, 1, 10\}$, and set $E(h_t) = 0.0009$. Given the values of ϕ, CV, and $E(h_t)$, the corresponding values of τ and σ_h^2 can be easily computed. Following the setup for the stable distribution $SD(\alpha, \beta, \gamma, \delta)$ in Vankov et al. (2019), we hold $\gamma = 1$ and $\delta = 0$ fixed throughout. We consider $(\alpha, \beta) = (1.75, 0.1), (1.7, 0.3), (1.5, -0.3)$ respectively as three experimental settings: (1.75, 0.1) gives the least heavy-tailed stable distribution and the least (right) skewness and is expected to be the easiest to handle; (1.5, -0.3) gives the most heavy-tailed stable distribution and the most (left) skewness and is expected to be the hardest to handle; (1.7, 0.3) has heavy-tailedness between that of the above two cases and the most (right) skewness. Following Kim et al. (1998), we assign a NIG conjugate prior for θ with $a_0 = 2.5$, $b_0 = 0.025$, $\boldsymbol{\mu}_0 = \begin{bmatrix} 0 \\ 0.9 \end{bmatrix}$ and $\boldsymbol{\Lambda}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$; this choice allows the prior density of θ to cover a fairly wide range, with the prior mean of ϕ centered at an empirically feasible value of 0.9.

For a given combination of values for ϕ, CV, and (α, β), we generated 100 simulated datasets, where each dataset is a time series of returns with length $T = 350$ observations. For each dataset, we ran the three particle Gibbs samplers (ABC-PG-cBF, ABC-PG-cBFAS, and ABC-PG-cAPF) using $N = 250$ particles for the embedded cSMC algorithm and a Gaussian ABC kernel with $\epsilon = 0.001$. For all sampling algorithms, we discarded the first 2000 iterations as burn-in and took the 5000 subsequent iterations as the posterior sample. The posterior means are treated as the Bayes' estimates for the parameters. The simulation results, summarized by computing the root-mean-squared errors (RMSEs) of the parameter estimates over the 100 simulated datasets, are presented in Tables 1, 2, and 3 for $(\alpha, \beta) = (1.75, 0.1), (1.7, 0.3), (1.5, -0.3)$, respectively.

The numerical results indicate that the proposed ABC-PG-cAPF sampler outperforms the corresponding cBF-based ones for most of the scenarios considered. The Gaussian ABC kernel tends to produce very uneven particle weights and SMC samplers can become hindered by weight degeneracy, especially for more heavy-tailed Z_t: Table 3 for $(\alpha, \beta) = (1.5, -0.3)$ has the highest RMSEs overall. While ancestor sampling (in cBFAS) is designed to tackle path degeneracy, doing so appears to have limited effectiveness for mitigating weight degeneracy; in fact, this extra sampling step may degrade the overall performance of cBFAS compared to cBF in the scenarios considered. With only a moderately sized N, the candidates available for cBFAS to partially replace the input trajectory at each step may not be of sufficient quality,

CV	ϕ	Algorithm	RMSE τ	RMSE ϕ	RMSE σ_h^2
10	0.9	ABC-PG-cBF	0.541	0.066	0.328
		ABC-PG-cBFAS	0.697	0.084	0.384
		ABC-PG-cAPF	0.377	0.047	0.228
10	0.95	ABC-PG-cBF	0.425	0.052	0.209
		ABC-PG-cBFAS	0.632	0.078	0.295
		ABC-PG-cAPF	0.309	0.038	0.152
10	0.98	ABC-PG-cBF	0.225	0.028	0.095
		ABC-PG-cBFAS	0.611	0.077	0.231
		ABC-PG-cAPF	0.200	0.024	0.066
1	0.9	ABC-PG-cBF	0.477	0.065	0.128
		ABC-PG-cBFAS	0.624	0.086	0.217
		ABC-PG-cAPF	0.441	0.060	0.101
1	0.95	ABC-PG-cBF	0.152	0.021	0.045
		ABC-PG-cBFAS	0.723	0.100	0.199
		ABC-PG-cAPF	0.158	0.021	0.043
1	0.98	ABC-PG-cBF	0.090	0.012	0.017
		ABC-PG-cBFAS	0.786	0.109	0.180
		ABC-PG-cAPF	0.080	0.011	0.012
0.1	0.9	ABC-PG-cBF	0.499	0.070	0.013
		ABC-PG-cBFAS	0.453	0.066	0.165
		ABC-PG-cAPF	0.498	0.070	0.008
0.1	0.95	ABC-PG-cBF	0.164	0.023	0.011
		ABC-PG-cBFAS	0.702	0.101	0.168
		ABC-PG-cAPF	0.159	0.022	0.014
0.1	0.98	ABC-PG-cBF	0.073	0.011	0.016
		ABC-PG-cBFAS	0.871	0.125	0.165
		ABC-PG-cAPF	0.086	0.013	0.018

Table 1 RMSEs of the SVM parameter estimates using the three different ABC-based particle Gibbs samplers, based on 100 simulated datasets with $T = 350$, $N = 250$, $Z_t \sim SD(1.75, 0.1, 1, 0)$ and $\epsilon = 0.001$.

such that the stochastic perturbations may introduce extra noise rather than improving the sampled trajectories. In contrast, the weight tempering strategy of cAPF helps reduce the variability in the weights, which can lead to more plausible trajectories being sampled, and in turn improve the accuracy of the parameter estimates. It can be noted that the advantages of cAPF over cBF may be more pronounced when the observations are more heavy-tailed ($\alpha = 1.5$, where weight degeneracy may be most severe), or when CV is larger (e.g., 10) and ϕ is smaller (e.g., 0.9). This is intuitively sensible, since cAPF considers the likelihood of the 'one-step-ahead' observation; when the log-volatility process is quite autocorrelated (i.e., CV is small or ϕ is large), this likelihood tends to be less informative (Johansen and Doucet, 2008). On a single CPU core, posterior sampling for one simulated dataset requires approximately 6.6 minutes for ABC-PG-cBF, 7.4 minutes for ABC-PG-cBFAS, and 7.1 minutes for ABC-PG-cAPF. Hence, the particle Gibbs samplers all have similar computational cost; cBFAS is slightly more expensive due to its extra ancestor sampling step, and cAPF is slightly more expensive due to its weight tempering.

Figure 2 shows the estimated posterior densities of each parameter obtained from each sampling algorithm for each of the 100 simulated datasets, taking the CV = 10, $\phi = 0.95$, and $(\alpha, \beta) = (1.7, 0.3)$ scenario as an example. In each panel, the average

CV	ϕ	Algorithm	RMSE τ	RMSE ϕ	RMSE σ_h^2
10	0.9	ABC-PG-cBF	0.576	0.070	0.347
		ABC-PG-cBFAS	0.745	0.090	0.415
		ABC-PG-cAPF	0.388	0.048	0.237
10	0.95	ABC-PG-cBF	0.426	0.052	0.215
		ABC-PG-cBFAS	0.693	0.085	0.327
		ABC-PG-cAPF	0.328	0.041	0.157
10	0.98	ABC-PG-cBF	0.250	0.030	0.100
		ABC-PG-cBFAS	0.666	0.084	0.252
		ABC-PG-cAPF	0.222	0.027	0.073
1	0.9	ABC-PG-cBF	0.515	0.070	0.143
		ABC-PG-cBFAS	0.703	0.097	0.248
		ABC-PG-cAPF	0.448	0.061	0.100
1	0.95	ABC-PG-cBF	0.197	0.027	0.053
		ABC-PG-cBFAS	0.783	0.108	0.219
		ABC-PG-cAPF	0.173	0.024	0.051
1	0.98	ABC-PG-cBF	0.134	0.020	0.033
		ABC-PG-cBFAS	0.847	0.118	0.198
		ABC-PG-cAPF	0.110	0.016	0.019
0.1	0.9	ABC-PG-cBF	0.497	0.070	0.031
		ABC-PG-cBFAS	0.564	0.082	0.199
		ABC-PG-cAPF	0.494	0.069	0.008
0.1	0.95	ABC-PG-cBF	0.168	0.023	0.023
		ABC-PG-cBFAS	0.789	0.114	0.194
		ABC-PG-cAPF	0.167	0.023	0.018
0.1	0.98	ABC-PG-cBF	0.117	0.017	0.026
		ABC-PG-cBFAS	0.945	0.136	0.189
		ABC-PG-cAPF	0.086	0.013	0.018

Table 2 RMSEs of the SVM parameter estimates using the three different ABC-based particle Gibbs samplers, based on 100 simulated datasets with $T = 350$, $N = 250$, $Z_t \sim SD(1.7, 0.3, 1, 0)$, $\epsilon = 0.001$.

posterior density over the 100 datasets is superimposed by the thick solid line. These plots further indicate that ABC-PG-cAPF provides more reliable posterior densities that tend to cluster closer to the truth (red dashed lines) across the simulated datasets; in contrast, the RMSEs of the other two sampling algorithms are higher due to less accurate posterior density estimates for some datasets.

Finally in the ABC setting, a smaller ϵ is generally preferred for a better ABC approximation of the target distribution; however, there is a trade-off as the sampling algorithms can suffer from increased weight degeneracy when ϵ is very small. For example, when we decreased ϵ to 0.0005, the RMSEs of the parameter estimates increased for all the cSMC algorithms. With the smaller ϵ, the RMSEs of ABC-PG-cBF and ABC-PG-cBFAS increased notably more than those of ABC-PG-cAPF, which indicates that our proposed sampler is more robust to such weight degeneracy compared to existing particle Gibbs samplers; see Section S2 in the Supplementary Material for the detailed results.

CV	ϕ	Algorithm	RMSE τ	RMSE ϕ	RMSE σ_h^2
10	0.9	ABC-PG-cBF	0.732	0.090	0.474
		ABC-PG-cBFAS	1.008	0.123	0.601
		ABC-PG-cAPF	0.445	0.056	0.308
10	0.95	ABC-PG-cBF	0.565	0.070	0.332
		ABC-PG-cBFAS	0.957	0.118	0.506
		ABC-PG-cAPF	0.390	0.049	0.209
10	0.98	ABC-PG-cBF	0.278	0.036	0.169
		ABC-PG-cBFAS	0.930	0.118	0.417
		ABC-PG-cAPF	0.257	0.032	0.103
1	0.9	ABC-PG-cBF	0.563	0.076	0.181
		ABC-PG-cBFAS	0.976	0.135	0.385
		ABC-PG-cAPF	0.480	0.065	0.117
1	0.95	ABC-PG-cBF	0.247	0.034	0.082
		ABC-PG-cBFAS	1.118	0.155	0.372
		ABC-PG-cAPF	0.174	0.024	0.049
1	0.98	ABC-PG-cBF	0.156	0.021	0.049
		ABC-PG-cBFAS	1.203	0.168	0.353
		ABC-PG-cAPF	0.087	0.013	0.017
0.1	0.9	ABC-PG-cBF	0.543	0.077	0.081
		ABC-PG-cBFAS	0.897	0.130	0.336
		ABC-PG-cAPF	0.495	0.069	0.012
0.1	0.95	ABC-PG-cBF	0.181	0.026	0.032
		ABC-PG-cBFAS	1.090	0.157	0.315
		ABC-PG-cAPF	0.165	0.023	0.025
0.1	0.98	ABC-PG-cBF	0.179	0.026	0.051
		ABC-PG-cBFAS	1.269	0.183	0.317
		ABC-PG-cAPF	0.087	0.013	0.019

Table 3 RMSEs of the SVM parameter estimates using the three different ABC-based particle Gibbs samplers, based on 100 simulated datasets with $T = 350$, $N = 250$, $Z_t \sim SD(1.5, -0.3, 1, 0)$, $\epsilon = 0.001$.

3.2 Comparison of methods for estimating both SVM and stable distribution parameters

We now compare our extended ABC-PG-cAPF with the PMMH and SF-PMwG algorithms for estimating both θ and ζ. We take the CV $= 1$, $\phi = 0.95$ scenario for further investigation following Jacquier et al. (1994) and consider the same three choices of $\zeta = (\alpha, \beta) = (1.75, 0.1), (1.7, 0.3), (1.5, -0.3)$ as in Section 3.1. For all the algorithms, we assign the independent uniform priors $\alpha \sim Unif(0.5, 2)$ and $\beta \sim Unif(-1, 1)$ for the stable distribution parameters ζ, i.e., $\pi_\zeta(\zeta) \propto 1$ on $(0.5, 2) \times (-1, 1)$. Other settings, namely the prior for θ, length of observations ($T = 350$), Gaussian ABC kernel (with $\epsilon = 0.001$), and 2000 burn-in and 5000 sampling iterations, are chosen to be the same as in Section 3.1.

To complete the specification of Algorithm 6 in our extended ABC-PG-cAPF, the proposal kernel $\mathcal{Q}$ needs to be chosen. Similar to the way the proposal kernel is constructed in Vankov et al. (2019), our implementation uses an adaptive Metropolis-Hastings proposal (Haario et al., 2001) for ζ based on the empirical covariance: we let $\mathcal{Q}$ be a Gaussian random walk kernel, so that at iteration k, the proposal ζ^{**} is drawn from

$$\mathcal{Q}(\zeta^{**} \mid \zeta[k-1]) = N\left(\zeta[k-1], \mathbf{V}_k\right),$$

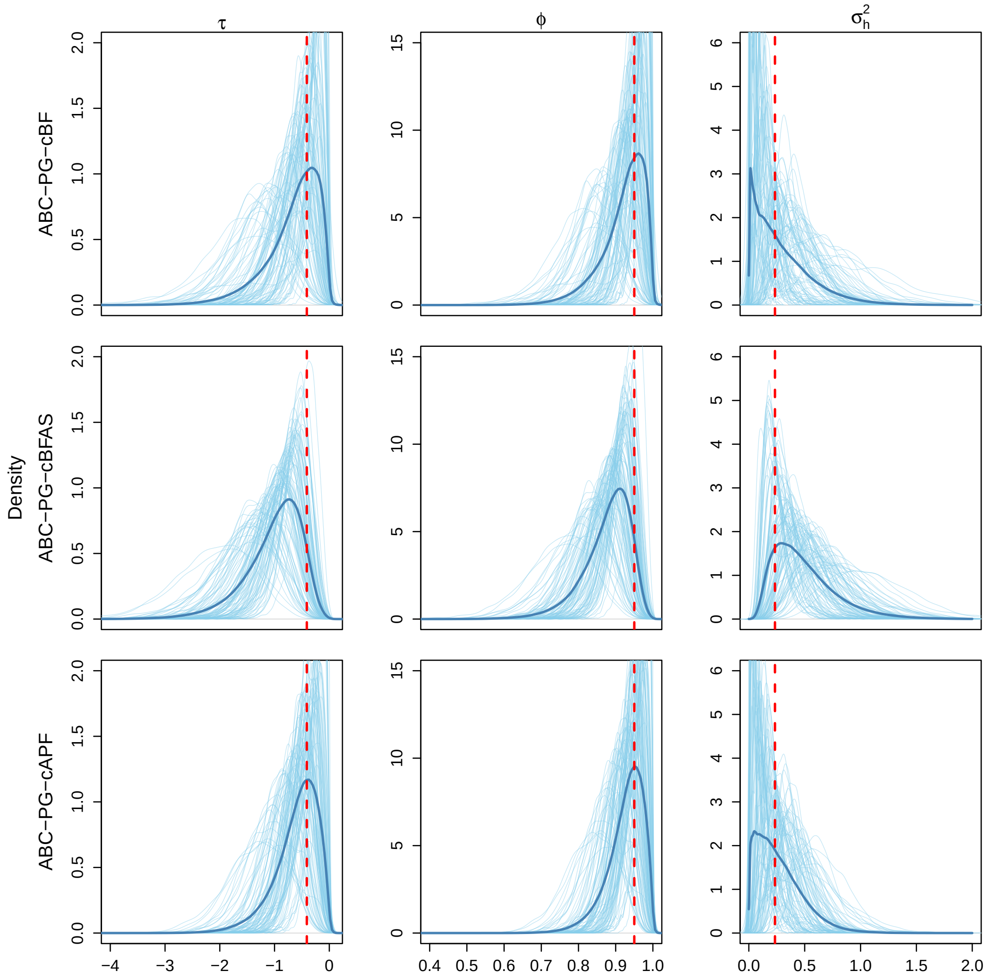

Fig. 2 Estimated posterior densities of the parameters τ, ϕ, and σ_h^2 for each of the 100 simulated datasets, based on the samples obtained from the three sampling algorithms in the scenario with CV = 10, $\phi = 0.95$, and $(\alpha, \beta) = (1.7, 0.3)$. In each panel, the thick solid line represents average posterior density over the 100 datasets. The true values of the parameters are indicated by the red dashed lines.

where $\zeta[k-1]$ is the draw of ζ from the previous iteration and

$$\mathbf{V}_k = \begin{cases} \begin{bmatrix} 0.2 & 0 \\ 0 & 0.2 \end{bmatrix} & \text{if } k \leq I \\ Cov(\zeta[1:(k-1)]) + \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix} & \text{if } k > I \end{cases}$$

where I is the number of burn-in iterations and $Cov(\zeta[1:(k-1)])$ is the empirical covariance matrix of all previous draws of ζ. We run the extended ABC-PG-cAPF with $N = 500$ particles.

For comparison, we implemented an ABC variant of the PMMH algorithm introduced in Andrieu et al. (2010), by incorporating an ABC-based bootstrap filter (ABC-BF) to handle the intractable stable distributions. Our implementation of PMMH closely follows that described by Vankov et al. (2019), which uses a single MH step with an adaptive Gaussian kernel to propose (θ, ζ), then approximates the marginal likelihood via the ABC-BF. In each iteration of SF-PMwG (Vankov et al., 2019), the parameters are updated in two blocks: $(\zeta, \sigma_h^2, h_{0:T})$ are updated together via a MH step, while τ and ϕ are updated via Gibbs steps with the help of closed-form conditional distributions. The MH update for $(\zeta, \sigma_h^2, h_{0:T})$ in SF-PMwG uses an adaptive Gaussian kernel to propose ζ, σ_h^2, then runs the ABC-APF (Algorithm 3) to approximate the marginal likelihood (in the same manner as PMMH) and samples one particle as the proposed trajectory for $h_{0:T}$. For τ and ϕ, which have conditional distributions available, the Gibbs step in SF-PMwG conditions on the $(\zeta, \sigma_h^2, h_{0:T})$ sampled from the MH step. Further details and algorithmic descriptions of PMMH and SF-PMwG are provided in Section S3 of the Supplementary Material. We run PMMH and SF-PMwG with $N = 10,000$ particles as suggested in Vankov et al. (2019).

Analogously to Section 3.1, for a given combination of values for (θ, ζ), we generated 100 simulated datasets. For each dataset, we ran the three samplers (ABC-PG-cAPF, PMMH, and SF-PMwG) as described above. The simulation results are summarized in Table 4, in terms of the average estimate and RMSE of each parameter over the 100 datasets. Generally, the RMSEs are larger for smaller values of α, especially when $(\alpha, \beta) = (1.5, -0.3)$. This is consistent with the results in Section 3.1, since a smaller α indicates heavier tails and greater variability from the stable distribution, which makes weight degeneracy and inference more challenging. The results support the overall efficacy of the extended ABC-PG-cAPF, which has RMSEs smaller than (or comparable to, in the case of $(\alpha, \beta) = (1.75, 0.1)$) those obtained by PMMH and SF-PMwG, especially for estimating the parameters (α, β) of the stable distribution. SF-PMwG generally outperforms PMMH (more notably for smaller values of α) with the help of its ABC-APF, which may handle weight degeneracy better than the ABC-BF in PMMH. Nevertheless, the performance of SF-PMwG is still dependent on the quality of its particle marginal likelihood approximation, such that the extended ABC-PG-cAPF handles these cases more effectively overall.

Furthermore, the time required to run the extended ABC-PG-cAPF on one dataset using a single CPU core is approximately 14.3 minutes, which is about 20 times faster than both PMMH and SF-PMwG. The overall computational cost can be mainly attributed to the embedded SMC algorithm, which is approximately linear in the particle size N. PMMH and SF-PMwG use a much larger particle size of $N = 10,000$, i.e., 20 times larger than the $N = 500$ in our extended ABC-PG-cAPF. Therefore, we also assessed whether PMMH and SF-PMwG can produce reasonable results with $N = 500$ instead, i.e., using a similar computational budget as the extended ABC-PG-cAPF. We found that even for the easiest case of $(\alpha, \beta) = (1.75, 0.1)$, PMMH and SF-PMwG failed to produce reasonable estimates with this reduced particle size, as presented in Table S4 of the Supplementary Material. This indicates that PMMH and SF-PMwG require a large N for their marginal likelihood approximations to be adequate; in contrast, the extended ABC-PG-cAPF can work well with a much smaller N since its embedded ABC-cAPF only updates the trajectory $h_{0:T}$ and does not estimate the marginal likelihood.

4 Application: the S&P 500 Index during the Financial Crisis in 2008

We now apply the extended ABC-PG-cAPF sampler to fit an SVM to the S&P 500 time-series data first introduced in Figure 1. Given the time series of the daily price (i.e., the average of the open and close price on each day), these daily returns are computed for the period January 2008 to March 2009. The large fluctuations around October 2008 indicate the climax of the well-known global financial crisis.

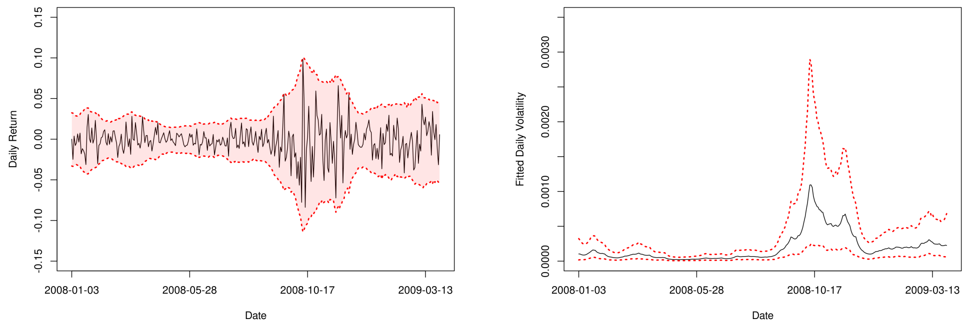

Fig. 3 The left panel presents the daily returns of the S&P 500 index from January 2008 to March 2009. The 2.5% and 97.5% quantiles of the sampled returns are shown with red dashed lines and the corresponding 95% credible interval for the returns is highlighted in red. The right panel presents the fitted daily volatility (black line) with a 95% credible interval (red dashed lines); the high volatility around the climax is well-captured.

We estimated the SVM and stable distribution parameters (θ, ζ) based on these data, running 2000 burn-in iterations of the extended ABC-PG-cAPF followed by 10000 sampling iterations, using $N = 1000$ particles and a Gaussian ABC kernel with $\epsilon = 0.001$, $\epsilon_\zeta = 0.05$. We adopt an NIG conjugate prior for θ with $a_0 = 2$, $b_0 = 0.5$, $\boldsymbol{\mu}_0 = \begin{bmatrix} 0 \\ 0.9 \end{bmatrix}$ and $\boldsymbol{\Lambda}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ following the work of Yang et al. (2018) on the same S&P 500 Index data, and a uniform prior for ζ as in Section 3.2. The Bayes' estimates (posterior means) and central 95% credible intervals of the parameters are reported in Table 5.

The Bayes' estimate of $\alpha = 1.813$ is consistent with the heavy-tailedness of the returns apparent in the Q-Q plot (Figure 1), and similar to the practical range of $\alpha \in (1.65, 1.8)$ for financial data indicated by previous studies (Kabašinskas et al.,

Parameter	Method	Estimate	RMSE
$\tau = -0.368$	ABC-PG-cAPF	-0.341	0.216
	PMMH	-0.339	0.235
	SF-PMwG	-0.295	0.205
$\phi = 0.95$	ABC-PG-cAPF	0.953	0.030
	PMMH	0.954	0.032
	SF-PMwG	0.942	0.055
$\sigma_h^2 = 0.068$	ABC-PG-cAPF	0.064	0.066
	PMMH	0.058	0.071
	SF-PMwG	0.078	0.089
$\alpha = 1.75$	ABC-PG-cAPF	1.679	0.131
	PMMH	1.636	0.236
	SF-PMwG	1.704	0.174
$\beta = 0.1$	ABC-PG-cAPF	0.070	0.189
	PMMH	-0.011	0.308
	SF-PMwG	0.122	0.276
$\tau = -0.368$	ABC-PG-cAPF	-0.352	0.201
	PMMH	-0.380	0.433
	SF-PMwG	-0.282	0.256
$\phi = 0.95$	ABC-PG-cAPF	0.951	0.028
	PMMH	0.948	0.057
	SF-PMwG	0.947	0.047
$\sigma_h^2 = 0.068$	ABC-PG-cAPF	0.070	0.068
	PMMH	0.076	0.151
	SF-PMwG	0.075	0.092
$\alpha = 1.7$	ABC-PG-cAPF	1.637	0.134
	PMMH	1.556	0.246
	SF-PMwG	1.619	0.200
$\beta = 0.3$	ABC-PG-cAPF	0.227	0.177
	PMMH	0.186	0.311
	SF-PMwG	0.247	0.286
$\tau = -0.368$	ABC-PG-cAPF	-0.382	0.303
	PMMH	-0.505	0.500
	SF-PMwG	-0.357	0.396
$\phi = 0.95$	ABC-PG-cAPF	0.947	0.042
	PMMH	0.926	0.079
	SF-PMwG	0.927	0.099
$\sigma_h^2 = 0.068$	ABC-PG-cAPF	0.088	0.122
	PMMH	0.303	1.128
	SF-PMwG	0.131	0.460
$\alpha = 1.5$	ABC-PG-cAPF	1.461	0.123
	PMMH	1.350	0.265
	SF-PMwG	1.370	0.223
$\beta = -0.3$	ABC-PG-cAPF	-0.249	0.145
	PMMH	-0.176	0.298
	SF-PMwG	-0.238	0.233

Table 4 Summary of parameter estimates for (θ, ζ) where CV=1, $\phi = 0.95$, and (α, β) are one of $(1.75, 0.1), (1.7, 0.3), (1.5, -0.3)$. Three samplers (ABC-PG-cAPF, PMMH, SF-PMwG) are compared, based on $T = 350$, $\epsilon = 0.001$ and 5000 posterior samples. ABC-PG-cAPF is run with $N = 500$ particles, while PMMH and SF-PMwG are run with $N = 10,000$ particles.

2009). While α has a relatively small 95% credible interval, in contrast the inference for β is highly uncertain; intuitively, the skewness of the stable distribution may contribute little to the variation of the observed returns compared to the heavy-tailedness. Hence, it may be difficult to generate precise estimates of β with limited observations.

In the right panel of Figure 3, we plot the fitted daily volatility along with a central 95% credible interval based on the posterior samples $\{h_t^*\}_{t=1}^T$; the estimated volatility peaks at the climax of the financial crisis. Lastly, the estimated 95% credible intervals for the returns are also superimposed on the left panel, which were computed via the samples generated from $l_t(r_t \mid h_t^*)$ for each draw of h_t^* at each time $t = 1, \dots, T$ and indicate a good fit to the data.

Parameter	Estimate	95% credible interval
τ	-0.307	(-0.680, -0.044)
ϕ	0.966	(0.926, 0.995)
σ_h^2	0.098	(0.050, 0.184)
α	1.813	(1.477, 1.987)
β	-0.195	(-0.929, 0.328)

Table 5 Estimates and credible intervals of τ, ϕ, σ_h^2, α and β based on fitting an SVM to S&P 500 Index data from January 2008 to March 2009. The numerical results were obtained from the ABC-PG-cAPF sampler with $N = 1000$ and 10000 posterior samples.

5 Conclusion and Discussion

In this paper, we proposed an ABC-based cAPF embedded within a particle Gibbs sampler for likelihood-free inference of the SVM. Our proposed sampler builds upon a rich SMC and SVM literature, e.g., the idea of using MCMC for parameter estimation in SVMs (Jacquier et al., 1994), particle MCMC for state space models (Andrieu et al., 2010), and ABC-based PMCMC for SVM inference with intractable likelihoods (Vankov et al., 2019). Compared to existing particle Gibbs samplers, the PMMH algorithm, and the SF-PMwG algorithm, the proposed ABC-PG-cAPF sampler produces more accurate parameter estimates with the help of its weight tempering strategy, as demonstrated in the simulation study.

Our sampler can be adapted for broader use with different models and setups. First, if there are any model parameters without closed-form conditional distributions (e.g., ϵ_t chosen to be a t-distribution and thus no conjugacy available for σ_h^2), this can be handled by incorporating an additional block of Metropolis-Hastings updates within the particle Gibbs sampler. Second, the computation of the tempered weights for cAPF can be adapted as appropriate to cover the high-density regions of the true importance weights. For example, if the likelihood involves a stable distribution with $0.5 < \alpha < 1$, the Lévy distribution (which corresponds to a stable distribution with $\alpha = 0.5$) can be a better choice of approximating distribution than a Cauchy. However, for a stable distribution with $\alpha < 0.5$ or other kinds of intractable distributions, alternative approximation schemes or numerical methods might be necessary to perform weight tempering efficiently. While ABC-PG-cAPF is proposed with Model (4) as the focus in this paper, the computational strategy is not limited to this specific problem and could be adapted to other SSMs with intractable likelihoods in future work, e.g., stochastic kinetic models (Owen et al., 2015; Lowe et al., 2023) and models with likelihoods that follow g-and-k distributions (Rayner and MacGillivray, 2002; Drovandi and Pettitt, 2011).

Supplementary information. The Supplementary Material contains the proof of Proposition 1 in Section 2.4 and the additional experiments and algorithms (PMMH and SF-PMwG) described in Section 3. The code and data to replicate the results of this study are provided in a supplementary .zip file.

Acknowledgements. This work was partially supported by Discovery Grant RGPIN-2019-04771 from the Natural Sciences and Engineering Research Council of Canada.

References

Andrieu, C., Doucet, A., Holenstein, R.: Particle Markov chain Monte Carlo methods. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **72**(3), 269–342 (2010)

Beaumont, M.A.: Approximate Bayesian computation. Annual Review of Statistics and Its Application **6**, 379–403 (2019)

Carvalho, C.M., Johannes, M.S., Lopes, H.F., Polson, N.G.: Particle learning and smoothing. Statistical Science **25**(1), 88–106 (2010)

Chambers, J.M., Mallows, C.L., Stuck, B.: A method for simulating stable random variables. Journal of the American Statistical Association **71**(354), 340–344 (1976)

Chopin, N., Singh, S.S.: On particle Gibbs sampling. Bernoulli **21**(3), 1855 (2015)

Del Moral, P., Doucet, A., Jasra, A.: Sequential Monte Carlo samplers. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **68**(3), 411–436 (2006)

Drovandi, C.C., Pettitt, A.N.: Likelihood-free Bayesian estimation of multivariate quantile distributions. Computational Statistics & Data Analysis **55**(9), 2541–2556 (2011)

Engle, R.F., Patton, A.J.: What good is a volatility model? Quantitative Finance **1**, 237–245 (2001)

Fridman, M., Harris, L.: A maximum likelihood approach for non-Gaussian stochastic volatility models. Journal of Business & Economic Statistics **16**(3), 284–291 (1998)

Gelman, A., Carlin, J.B., Stern, H.S., Rubin, D.B.: Bayesian Data Analysis. CRC Press, Boca Raton, FL (2014)

Gordon, N., Salmond, D., Smith, A.: Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proceedings F (Radar and Signal Processing) **140**(2), 107–113 (1993)

Haario, H., Saksman, E., Tamminen, J.: An adaptive Metropolis algorithm. Bernoulli, 223–242 (2001)

Hou, Z., Wong, S.W.: Estimating Boltzmann averages for protein structural quantities using sequential Monte Carlo. Statistica Sinica **34**, 1263–1280 (2024)

Johansen, A.M., Doucet, A.: A note on auxiliary particle filters. Statistics & Probability Letters **78**(12), 1498–1504 (2008)

Jacquier, E., Polson, N.G., Rossi, P.E.: Bayesian analysis of stochastic volatility models. Journal of Business & Economic Statistics **12**(4), 371–389 (1994)

Jacquier, E., Polson, N.G., Rossi, P.E.: Bayesian analysis of stochastic volatility models with fat-tails and correlated errors. Journal of Econometrics **122**(1), 185–212 (2004)

Kanter, M.: Stable densities under change of scale and total variation inequalities. The Annals of Probability **3**(4), 697–707 (1975)

Kitagawa, G.: A self-organizing state-space mode. Journal of the American Statistical Association **93**(443), 1203–1215 (1998)

Kabašinskas, A., Rachev, S., Sakalauskas, L., Sun, W., Belovas, I.: Alpha-stable paradigm in financial markets. Journal of Computational Analysis and Applications **11**(4), 641–668 (2009)

Kim, S., Shephard, N., Chib, S.: Stochastic volatility: likelihood inference and comparison with arch models. The review of economic studies **65**(3), 361–393 (1998)

Lombardi, M.J., Calzolari, G.: Indirect estimation of α-stable stochastic volatility models. Computational Statistics & Data Analysis **53**(6), 2298–2308 (2009)

Lin, M., Chen, R., Liu, J.S.: Lookahead strategies for sequential Monte Carlo. Statistical Science **28**(1), 69–94 (2013)

Lowe, T.E., Golightly, A., Sherlock, C.: Accelerating inference for stochastic kinetic models. Computational Statistics & Data Analysis **185**, 107760 (2023)

Liu, J.S.: Monte Carlo Strategies in Scientific Computing. Springer, New York (2001)

Lindsten, F., Jordan, M.I., Schon, T.B.: Particle Gibbs with ancestor sampling. Journal of Machine Learning Research **15**(1), 2145–2184 (2014)

Lemke, T., Riabiz, M., Godsill, S.J.: Fully Bayesian inference for α-stable distributions using a Poisson series representation. Digital Signal Processing **47**, 96–115 (2015)

Liu, J., West, M.: Combined Parameter and State Estimation in Simulation-Based Filtering, pp. 197–223. Springer, New York (2001)

Mandelbrot, B.: The variation of certain speculative prices. The Journal of Business **36**(4), 394–419 (1963)

McCulloch, J.H.: Simple consistent estimators of stable distribution parameters. Communications in Statistics - Simulation and Computation **15**(4), 1109–1136 (1986)

Mittnik, S., Doganoglu, T., Chenyao, D.: Computing the probability density function of the stable paretian distribution. Mathematical and Computer Modelling **29**(10-12), 235–240 (1999)

Marin, J.-M., Pudlo, P., Robert, C.P., Ryder, R.J.: Approximate Bayesian computational methods. Statistics and Computing **22**(6), 1167–1180 (2012)

Nakagome, S., Fukumizu, K., Mano, S.: Kernel approximate Bayesian computation in population genetic inferences. Statistical Applications in Genetics and Molecular Biology **12**(6), 667–678 (2013)

Nolan, J.P.: Numerical calculation of stable densities and distribution functions. Communications in Statistics. Stochastic Models **13**(4), 759–774 (1997)

Nolan, J.P.: An algorithm for evaluating stable densities in Zolotarev's (M) parameterization. Mathematical and Computer Modelling **29**(10-12), 229–233 (1999)

Nolan, J.P.: Univariate Stable Distributions. Springer, Switzerland AG (2020)

Owen, J., Wilkinson, D.J., Gillespie, C.S.: Scalable inference for Markov processes with intractable likelihoods. Statistics and Computing **25**, 145–156 (2015)

Peters, G.W., Fan, Y., Sisson, S.A.: On sequential Monte Carlo, partial rejection control and approximate Bayesian computation. Statistics and Computing **22**, 1209–1222 (2012)

Park, M., Jitkrittum, W., Sejdinovic, D.: K2-ABC: approximate Bayesian computation with kernel embeddings. In: Artificial Intelligence and Statistics, pp. 398–407 (2016). PMLR

Pitt, M.K., Shephard, N.: Filtering via simulation: auxiliary particle filters. Journal of the American Statistical Association **94**(446), 590–599 (1999)

Rayner, G.D., MacGillivray, H.L.: Numerical maximum likelihood estimation for the g-and-k and generalized g-and-h distributions. Statistics and Computing **12**(1), 57–75 (2002)

Svensson, A., Schön, T.B., Kok, M.: Nonlinear state space smoothing using the conditional particle filter. IFAC-PapersOnLine **48**(28), 975–980 (2015)

Storvik, G.: Particle filters for state-space models with the presence of unknown static parameters. IEEE Transactions on Signal Processing **50**(2), 281–289 (2002)

Taylor, S.J.: Modelling Financial Time Series. World scientific, Singapore (2008)

Vankov, E.R., Guindani, M., Ensor, K.B.: Filtering and estimation for a class of stochastic volatility models with intractable likelihoods. Bayesian Analysis **14**(1), 29–52 (2019)

Wilkinson, R.D.: Approximate Bayesian computation (ABC) gives exact results under the assumption of model error. Statistical Applications in Genetics and Molecular Biology **12**(2), 129–141 (2013)

Wuertz, D., Maechler, M., Rmetrics core team members.: Stabledist: Stable Distribution Functions. (2016). R package version 0.7-1. https://CRAN.R-project.org/package=stabledist

Yang, B., Stroud, J.R., Huerta, G.: Sequential Monte Carlo smoothing with parameter estimation. Bayesian Analysis **13**(4), 1137–1161 (2018)