Bayesian estimation of general state space models by applying adaptive sampling and particle filter methods

Ralph S. Silva
(Joint work with Michael K. Pitt, Paolo Giordani and Robert Kohn)

Departamento de Métodos Estatísticos
Instituto de Matemática
Universidade Federal de Rio de Janeiro

Univ. of St. Gallen, 27/09/2011

Summary

Introduction

- **Bayesian inference in general state space models**
- **Standard particle filter (SPF) by Gordon et al. (1993)**
- **Fully adapted particle filter (FAPF) by Pitt and Shephard (1999); Pitt (2002)**
- **Partially adapted particle filter (PAPF)**
- **PAPF or FAPF can be much more efficient than the SPF**
- **Adaptive random walk Metropolis (Roberts and Rosenthal, 2009)**
- **Adaptive independent Metropolis-Hastings (Giordani and Kohn, 2010)**
- **Efficient and unbiased estimation of the marginal likelihood**

State space models

- y_t and x_t are the observation and the state at time t;
- Observation equation: $p(y_t|x_t;\theta)$;
- Transition equation: $p(x_t|x_{t-1};\theta)$;
- θ is a vector of unknown parameters; and
- The initial state is $p(x_0|\theta)$.

The filtering equations for the state space model (for $t \geq 1$) are

$$p(x_t|y_{1:t-1};\theta) = \int p(x_t|x_{t-1};\theta)p(x_{t-1}|y_{1:t-1};\theta)dx_{t-1}, \tag{1a}$$

$$p(x_t|y_{1:t};\theta) = \frac{p(y_t|x_t;\theta)p(x_t|y_{1:t-1};\theta)}{p(y_t|y_{1:t-1};\theta)}, \tag{1b}$$

$$p(y_t|y_{1:t-1};\theta) = \int p(y_t|x_t;\theta)p(x_t|y_{1:t-1};\theta)dx_t. \tag{1c}$$

where $y_{1:t} = \{y_1, \ldots, y_t\}$.

Equations (1a)–(1c) allow for filtering for a given θ and for evaluating the likelihood of the observations $y = y_{1:T}$,

$$p(y|\theta) = p(y_1|\theta) \prod_{t=1}^{T-1} p(y_{t+1}|y_{1:t};\theta) \ . \tag{2}$$

- If the likelihood $p(y|\theta)$ can be computed, maximum likelihood and MCMC methods can be used;
- For linear Gaussian state space models, Kalman filter can be applied;
- Other models can be estimated by MCMC using auxiliary latent variables (Kim et al., 1998);
- An alternative is to sample from the state in blocks in an MCMC scheme (Shephard and Pitt, 1997);
- In general, the integrals in equations (1a)–(1c) are computationally intractable and the SPF algorithm was proposed by Gordon et al. (1993) as a method for approximating them; and
- Pitt and Shephard (1999) propose the auxiliary particle filter method (ASIR) which is more efficient than SPF when the observation density is informative relative to the transition density.

General ASIR method

- The general auxiliary SIR (ASIR) filter of Pitt and Shephard (1999) may be thought of as a generalisation of the SIR method of Gordon et al. (1993).
- The following algorithm describes the one time step ASIR update and is initialized with samples $x_0^k \sim p(x_0)$ with mass $1/M$ for $k = 1, ..., M$.

Algorithm

Given samples $x_t^k \sim p(x_t|y_{1:t})$ *with mass* π_t^k *for* $k = 1, ..., M$.
For $t = 0, .., T-1$:

1. *For* $k = 1 : M$, *compute* $\omega_{t|t+1}^k = g(y_{t+1}|x_t^k)\pi_t^k, \quad \pi_{t|t+1}^k = \dfrac{\omega_{t|t+1}^k}{\sum_{i=1}^M \omega_{t|t+1}^i}$.
2. *For* $k = 1 : M$, *sample* $\widetilde{x}_t^k \sim \sum_{i=1}^M \pi_{t|t+1}^i \delta(x_t - x_t^i)$. ($\delta$*: Dirac function.*)
3. *For* $k = 1 : M$, *sample* $x_{t+1}^k \sim g(x_{t+1}|\widetilde{x}_t^k; y_{t+1})$.
4. *For* $k = 1 : M$, *compute*

$$\omega_{t+1}^k = \frac{p(y_{t+1}|x_{t+1}^k)p(x_{t+1}^k|\widetilde{x}_t^k)}{g(y_{t+1}|\widetilde{x}_t^k)g(x_{t+1}^k|\widetilde{x}_t^k; y_{t+1})}, \qquad \pi_{t+1}^k = \frac{\omega_{t+1}^k}{\sum_{i=1}^M \omega_{t+1}^i}.$$

- In Step 2, multinomial sampling may be employed but stratified sampling is generally to be preferred.
- Note that the true joint density may be written as,

$$p(y_{t+1}|x_{t+1})p(x_{t+1}|x_t) = p(y_{t+1}|x_t)p(x_{t+1}|x_t;y_{t+1}),$$

  where

$$p(y_{t+1}|x_t) = \int p(y_{t+1}|x_{t+1})p(x_{t+1}|x_t)dx_{t+1},$$
$$p(x_{t+1}|x_t;y_{t+1}) = p(y_{t+1}|x_{t+1})p(x_{t+1}|x_t)/p(y_{t+1}|x_t).$$

- Typically this fully adapted form is unavailable but when it is the approximating joint density may be chosen to be the true joint. That is,

$$g(y_{t+1}|x_t)g(x_{t+1}|x_t;y_{t+1}) = p(y_{t+1}|x_t)p(x_{t+1}|x_t;y_{t+1}).$$

- In this case Step 4 becomes redundant as $\omega_{t+1}^k = 1$, ($\pi_{t+1}^k = 1/M$) and the method reduces to what Pitt and Shephard (2001) call the fully adapted algorithm.

- The SIR method of Gordon et al. (1993) arises when the joint proposal is chosen as,

$$g(y_{t+1}|x_t) \times g(x_{t+1}|x_t) = 1 \times p(x_{t+1}|x_t),$$

in which case, $g(y_{t+1}|x_t)$ is constant and $g(x_{t+1}|x_t; y_{t+1}) = p(x_{t+1}|x_t)$.
- In this case, Step (1) above leaves the weights unchanged (as $\pi^k_{t|t+1} = \pi^k_t$).
- The goal of the auxiliary particle filter is to get as close to full adaption as possible, when full adaption is not analytically possible.
- This is achieved by making

$$g(y_{t+1}|x_t) \quad \text{close to} \quad p(y_{t+1}|x_t)$$

as a function of x_t and

$$g(x_{t+1}|x_t; y_{t+1}) \quad \text{close to} \quad p(x_{t+1}|x_t; y_{t+1}).$$

- The general ASIR estimator of $p(y_t|y_{1:t-1})$, introduced and used by Pitt (2002), is

$$\widehat{p}^A(y_t|y_{1:t-1}) = \left\{ \sum_{k=1}^{M} \frac{\omega_t^k}{M} \right\} \left\{ \sum_{k=1}^{M} \omega_{t-1|t}^k \right\}. \qquad (3)$$

- We define the information in the swarm of particles at time t as $\mathcal{A}_t = \{x_t^k; \pi_t^k\}$.
- For full adaption $\omega_t^k = 1$ and $\omega_{t-1|t}^k = p(y_t|x_{t-1}^k)/M$ and the first summation in (3) disappears.
- For the SIR method, $\omega_t^k = p(y_t|x_t^k)$ and $\omega_{t-1|t}^k = \pi_{t-1}^k$ and the second summation in (3) disappears.

- The theorem below establishes that the ASIR Algorithm together with the estimator of (3) is unbiased (see Del Moral (2004) and our paper).
- It enables very efficient likelihood estimators from the ASIR method to be used within an MCMC algorithm.

Theorem

The ASIR likelihood

$$\widehat{p}^{A}(y_{1:t}) = \widehat{p}^{A}(y_1) \prod_{t=2}^{T} \widehat{p}^{A}(y_t | y_{1:t-1}) \tag{4}$$

is unbiased in the sense that

$$E(\widehat{p}^{A}(y_{1:t})) = p(y_{1:t}).$$

- The ASIR likelihood $\widehat{p}^A(y_{1:t})$ is called the simulated likelihood.
- Andrieu et al. (2010) show that we can view $\hat{p}(y|\theta) = p_S(y|\theta, u)$, where u is a set of auxiliary (uniform) variables u, the subscript S denotes a simulated likelihood, and

$$\int p_S(y|\theta, u)p(u)du = p(y|\theta). \tag{5}$$

- It follows that the posterior $p_S(\theta|y) = p(\theta|y)$ so that a method that simulates from $p_S(\theta, u|y)$ yields iterates from the correct posterior $p(\theta|y)$.
- The variance of the log of the simulated likelihood is $O(T/M)$ so it will be necessary to take the number of particles $M = O(T)$ to keep a constant standard deviation as T increases.
- This implies that the particle filter MCMC algorithm is of order $O(T^2)$ for T large and means that it is important to make the particle filter as efficient as possible.

Adaptive sampling for the simulated likelihood

- The target density for posterior inference is

$$p_S(\theta, u|y) \propto p_S(y|\theta, u)p(\theta)p(u).$$

- We may use a Metropolis-Hastings simulation method.
- Suppose that given some initial θ_0, the $j-1$ iterates $(\theta_1, u_1), \ldots, (\theta_{j-1}, u_{j-1})$ have been generated.
- The jth iterate, (θ_j, u_j), is generated from the proposal density $q_j(\theta; \tilde{\theta})p(u)$, which may also depend on some other value of θ which we call $\tilde{\theta}$.
- Let (θ_j^p, u_j^p) be the proposed value of (θ_j, u_j) generated from $q_j(\theta; \theta_{j-1})p(u)$.

Adaptive sampling for the simulated likelihood

- Then we take $(\theta_j, u_j) = (\theta_j^p, u_j^p)$ with probability

$$\alpha(\theta_{j-1}, u_{j-1}; \theta_j^p, u^p) = \min\left\{1, \frac{p_S(y|\theta_j^p, u_j^p)p(\theta_j^p)}{p_S(y|\theta_{j-1}, u_{j-1})p(\theta_{j-1})} \frac{q_j(\theta_{j-1}; \theta_j^p)}{q_j(\theta_j^p; \theta_{j-1})}\right\}, \quad (6)$$

  with $p(u_j^p)$ and $p(u_{j-1})$ cancelling out, and take $(\theta_j, u_j) = (\theta_{j-1}, u_{j-1})$ otherwise.
- We say that the proposal is independent if $q_j(\theta; \tilde{\theta}) = q_j(\theta)$.
- In adaptive sampling the parameters of $q_j(\theta; \tilde{\theta})$ are estimated from the iterates $\theta_1, \ldots, \theta_{j-2}$.
- We use the adaptive independent Metropolis Hastings scheme of Giordani and Kohn (2010) and the adaptive random walk Metropolis scheme of Roberts and Rosenthal (2009).

Adaptive sampling and parallel computation

- For (adaptive) independent Metropolis-Hastings proposals we can use the following three step approach. Let θ^c the current value of θ generated by the sampling scheme and $q_c(\theta)$ the current proposal density for θ.
  (a) For each of J processors generate K proposed values of θ, which we write as $\theta_{j,k}^{(p)}, k = 1, \ldots, K$, and compute the corresponding logs of the ratios $\hat{p}(y|\theta_{j,k}^{(p)})p(\theta_{j,k}^{(p)})/q(\theta_{j,k}^{(p)})$.
  (b) After each K block of proposed values is generated for each processor, carry out Matropolis Hastings selection of the JK proposed $\{\theta_{j,k}^{(p)}\}$ parameters using a single processor to obtain $\{\theta_{j,k}\}$ draws from the chain. This last step is fast because it is only necessary to draw uniform variates.
  (c) Use the previous iterates and the $\theta_{j,k}$ to update the proposal density $q_c(\theta)$ and θ_c. In our applications of this approach, K is chosen so that KJ is approximately the time between updates of the adaptive independent Metropolis Hastings sampling scheme.

Adaptive sampling and parallel computation

- A second approach applies to all Metropolis Hastings sampling schemes, and in particular to the adaptive random walk Metropolis proposal.
- Suppose that J processors are available.
- The likelihood is estimated for a given θ on each of the processors using the particle filter with M particles.
- These estimates are then averaged to get an estimate of the likelihood based on JM particles.
- This approach is similar to, but faster, than using a single processor.
- It is then possible to estimate the likelihood using a large number of particles.
- However, for a given number of generated particles, the first approach can be shown to be statistically more efficient than the second.

Estimating the marginal likelihood

- Marginal likelihoods are often used to compare two or more models.
- For a given model, let θ be the vector of model parameters, $p(y|\theta)$ the likelihood of the observations y and $p(\theta)$ the prior for θ.
- The marginal likelihood is defined as

$$p(y) = \int p(y|\theta)p(\theta)d\theta. \tag{7}$$

which in our case can also be written as

$$p(y) = \int p_S(y|\theta, u)p(\theta)p(u)d\theta\, du. \tag{8}$$

- It is often difficult to evaluate or estimate $p(y)$ in non-Gaussian state space models, although auxiliary variable methods can be used in some problems.
- The marginal likelihood can be estimated using bridge or importance sampling, with the computation carried out within the adaptive sampling framework so that a separate simulation run is unnecessary.

Comparing the standard SIR particle filter with adapted ASIR particle filters

- In the examples, we make the comparison in terms of three criteria.
- The first is the acceptance rate of the adaptive independent Metropolis Hastings sampler, which we define as the percentage of accepted draws.
- The second is the inefficiencies of the iterates of the parameters obtained using the adaptive independent Metropolis Hastings method of Giordani and Kohn (2010).
- The third is the standard deviation of the simulated log-likelihood $p_S(y|\theta, u)$ evaluated at the true value of θ, which is a good measure of how close the particle filter likelihood is to the true likelihood $p(y|\theta)$.

- We define the inefficiency of the sampling scheme for a given parameter as the variance of the parameter estimate divided by its variance if the sampling scheme generates independent iterates.
- We estimate the inefficiency factor for a given parameter as

$$\text{IF} = 1 + 2\sum_{j=1}^{L^\star} \hat{\rho}_j,$$

where $\hat{\rho}_j$ is the estimated autocorrelation of the parameter iterates at lag j. As a rule of thumb, the maximum number of lags $L^\star$ that we use is $L^\star = \min\{1000, L\}$, where L is the lowest index j such that $|\hat{\rho}_j| < 2/\sqrt{K}$ where K is the sample size used to compute $\hat{\rho}_j$.

Example 1: Autogressive model observed with noise

Consider the following first order autoregression (AR(1)) plus noise model,

$$y_t|x_t \sim \mathcal{N}(x_t, \sigma^2)$$

$$x_{t+1}|x_t \sim \mathcal{N}(\mu + \phi(x_t - \mu), \tau^2) \quad (9)$$

$$x_0 \sim \mathcal{N}(\mu, \tau^2/(1-\phi^2)). \quad (10)$$

The prior distributions are

- $\mu \sim \mathcal{N}(0, 100)$,
- $\phi \sim \mathcal{U}(0, 1)$,
- $\sigma^2 \sim \mathcal{IG}(0.1, 0.1)$, and
- $\tau^2 \sim \mathcal{IG}(0.1, 0.1)$.

- Our simulation study uses 50 replicated data sets with 500 observations each.
- We set $\mu = 0$, $\phi = 0.6$, $\tau^2 = 1$, $x_0 \sim \mathcal{N}(\mu, \tau^2/(1-\phi^2))$ and two values for $\sigma^2 = \{0.01, 1.0\}$, corresponding to high and low signal to noise ratios.
- We ran 30 000 iterations of the adaptive independent Metropolis Hastings for the posterior distribution using the standard particle filter and the fully adapted particle filter with differing number of particles.
- The update times for the adaptive independent Metropolis Hastings were at iterations 100, 200, 500, 1000, 1500, 2000, 3000, 4000, 5000, 10000, 15000 and 20000.
- We initialized the adaptive independent Metropolis Hastings based on a normal proposal formed from 5000 draws of a previous run of the adaptive random walk Metropolis and the Kalman filter.

Results for the high signal to noise case $\sigma^2 = 0.01$

Table: AR(1) + noise. High signal to noise. Medians and interquantile ranges (IQR) of the estimated medians and standard deviations of the log of the simulated likelihood function at the true value for 50 different data sets.

N. Particles	Median		Standard Deviation	
	Median	IQR	Median	IQR
	Standard Particle Filter			
100	-839.12	83.34	44.0381	21.0316
500	-729.43	26.09	10.5420	8.2875
1000	-719.10	20.13	5.5507	5.2818
2000	-714.95	18.16	2.8977	2.4716
	Fully Adapted Particle Filter			
100	-711.69	17.74	0.1431	0.0160

Results for the high signal to noise case $\sigma^2 = 0.01$

Table: AR(1) + noise. High signal to noise. Medians and interquartile range (IQR) of the acceptance rates and the inefficiencies over 50 replications of the autoregressive model using different particle filters and adaptive independent Metropolis Hastings.

N. Particles	Ac. Rate		σ^2		τ^2		μ		ϕ	
	Median	IQR	Median	IQR	Median	IQR	Median	IQR	Median	IQR
	Kalman Filter									
	72.18	4.89	1.93	0.34	1.85	0.35	1.76	0.20	1.83	0.24
	Standard Particle Filter									
1000	9.04	5.75	70.25	35.10	63.76	30.98	59.76	58.41	64.64	42.55
2000	21.81	10.87	21.84	20.64	22.97	17.62	20.48	23.16	25.09	24.07
4000	33.27	9.19	9.33	6.33	9.11	7.20	9.66	6.58	8.95	8.13
	Fully Adapted Particle Filter									
100	58.83	3.04	3.02	0.56	2.91	0.66	2.63	0.43	2.80	0.45
500	67.64	2.30	2.23	0.31	2.08	0.31	2.02	0.20	2.10	0.24

Results for the low signal to noise case $\sigma^2 = 1.0$

Table: AR(1) + noise. Low signal to noise. Medians and interquantile ranges (IQR) of the estimated medians and standard deviations of the log of the simulated likelihood function at the true value for 50 different data sets.

N. Particles	Median		Standard Deviation	
	Median	IQR	Median	IQR
	Standard Particle Filter			
100	-904.0827	19.0013	2.4479	0.2372
500	-901.7877	18.8020	1.0793	0.1170
1000	-901.4966	18.7909	0.7629	0.0550
	Fully Adapted Particle Filter			
100	-901.4727	18.8540	0.7057	0.0398

Results for the low signal to noise case $\sigma^2 = 1.0$

Table: AR(1) + noise. Low signal to noise. Medians and interquartile range (IQR) of the acceptance rates and the inefficiencies over 50 replications of the autoregressive model using different particle filters and adaptive independent Metropolis Hastings.

N. Particles	Ac. Rate		σ^2		τ^2		μ		ϕ	
	Median	IQR	Median	IQR	Median	IQR	Median	IQR	Median	IQR
	Kalman Filter									
	73.81	2.26	2.04	0.38	2.12	0.46	1.73	0.12	1.94	0.25
	Standard Particle Filter									
500	6.84	6.28	92.85	76.67	95.95	71.78	71.63	51.26	85.93	68.60
1000	29.86	18.15	19.67	27.30	21.10	24.91	10.87	16.07	18.18	28.90
2000	42.47	13.37	11.87	11.20	10.72	8.98	5.36	3.62	9.02	6.88
4000	52.95	11.85	6.38	6.86	6.16	4.53	3.28	2.54	5.02	4.11
	Fully Adapted Particle Filter									
100	53.94	9.24	3.48	0.88	3.53	0.84	3.27	1.32	3.62	0.92

Performance of the adaptive sampling schemes on real examples

- We illustrate the difference in performance between the adaptive random walk Metropolis sampling scheme of Roberts and Rosenthal (2009) and that the adaptive independent Metropolis Hastings scheme of Giordani and Kohn (2010).
- The comparison between the two schemes is in terms of three criteria.
- The first two are the acceptance rate of the Metropolis-Hastings method and the inefficiency factors (IF) of the parameters. They are independent of the way the algorithms are implemented.
- To obtain an overall measure of the effectiveness of a sampler, we define its equivalent computing time $ECT = 1000 \times IF \times t$, where t is the time per iteration of the sampler. We interpret ECT as the time taken by the sampler to attain the same accuracy as that attained by 1000 independent draws of the same sampler.
- The results presented are for a single processor and the two parallel methods discussed before.
- To simplify the presentation, we mainly present results for the standard particle filter.

Example 1: Stochastic volatility model with leverage and outliers

- The example considers the univariate stochastic volatility (SV) model

$$\begin{aligned} y_t &= K_t \exp(x_t/2)\varepsilon_t, & \varepsilon_t &\sim \mathcal{N}(0,1) \\ x_{t+1} &= \mu + \phi(x_t - \mu) + \sigma_\eta \eta_t, & \eta_t &\sim \mathcal{N}(0,1) \end{aligned} \tag{11}$$

  where $\text{corr}(\varepsilon_t, \eta_t) = \rho$, $\Pr(K_t = 2.5) = \omega$ and $\Pr(K_t = 1) = 1 - \omega$, with $\omega << 1$.
- The model also allows for outliers in the observation equation because the standard deviation of y_t given x_t can be 2.5 its usual size when $K_t = 2.5$.
- The prior distributions:
  - $\mu \sim \mathcal{N}(0, 10^2)$,
  - $\phi \sim \mathcal{TN}_{(0,1)}(0.9, 0.1)$,
  - $\sigma_\eta^2 \sim \mathcal{IG}(0.01, 0.01)$, and
  - $\rho \sim \mathcal{TN}_{(-1,1)}(0, 10^6)$.

S&P 500 index

- We apply the SV model (11) to the Standard and Poors (S&P) 500 data from 02/Jan/1970 to 14/Dec/1973 obtained from Yahoo Finance web site[1]. The data consists of $T = 1\ 000$ observations.
- Table 5 shows the acceptance rates, the inefficiencies and the equivalent computing time over 10 replications of the stochastic volatility model using the standard particle filter and the two adaptive Metropolis Hastings schemes.
- In the table, SP stands for a single processor, MP_1 for multiprocessor method 1 and MP_2 for multiprocessor method 2 (where the simulated likelihood is obtained as an average).
- We use eight processors for both the MP_1 and MP_2 schemes.

[1] http://au.finance.yahoo.com/q/hp?s=ˆGSPC

S&P 500 index

- The basic number of particles in this example is $K = 500$, which means that SP uses 4000 particles in a single processor, MP_1 uses 4000 particles in each processor and MP_2 uses 500 particles in each processor.
- We ran all the algorithm for 10000 iterations and took the last 5000 to compute the results.
- The equivalent computing time is obtained by taking the overall time divided by the number of iterations times the inefficiency times 1000.
- The update times for the adaptive independent Metropolis Hastings using SP or MP_2 were at 100, 200, 500, 1000, 2000, 3000, 4000, 5000, 6000 and 7500.
- The block sizes (also the update times) for the adaptive adaptive independent Metropolis Hastings MP_1 were 15, 25, 60, 125, 250, 375, 500, 625, 750 and 940.

S&P 500 index

Table: SV model. Medians and interquartile range (between brackets) of the acceptance rates, the inefficiencies and the equivalent computing time (t×inefficiency×1000) over 10 replications of the stochastic volatility model using the standard particle filter and differing adaptive Metropolis Hastings schemes. SP = single processor, MP_1 = multiprocessor Metropolis Hastings and MP_2 = multiprocessor averaging the likelihood function.

Algorithm	Ac. Rate	Inefficiency			Equivalent Computing Time		
		μ	logit(ϕ)	$\log(\sigma_\eta^2)$	μ	logit(ϕ)	$\log(\sigma_\eta^2)$
ARWM-SP	24.5 (0.6)	25.47 (16.33)	30.20 (5.12)	20.00 (2.80)	13463.7 (8589.8)	15972.9 (2655.3)	10603.0 (1514.0)
ARWM-MP_2	22.2 (5.5)	25.04 (12.96)	31.07 (14.02)	20.51 (8.83)	3594.5 (1886.9)	4453.4 (1994.3)	2930.4 (1284.3)
AIMH-SP	51.6 (0.8)	6.45 (7.12)	3.46 (0.83)	3.08 (0.07)	3430.9 (3777.9)	1841.7 (439.7)	1638.7 (39.9)
AIMH-MP_1	52.3 (3.3)	3.44 (2.13)	3.42 (0.76)	3.63 (0.50)	237.8 (147.4)	235.3 (50.5)	250.0 (34.6)
AIMH-MP_2	53.6 (3.2)	4.40 (6.74)	3.52 (3.37)	3.15 (0.56)	627.1 (975.4)	504.4 (482.5)	451.1 (75.8)

S&P 500 index

Table: Logarithms of the marginal likelihoods for four different SV models for the two particle filter algorithms computed using the adaptive independent Metropolis Hastings algorithm. BS and IS mean bridge sampling and importance sampling.

Model	Standard Particle Filter		Partially Adapted Particle Filter	
	$\log(p_{BS}(y))$	$\log(p_{IS}(y))$	$\log(p_{BS}(y))$	$\log(p_{IS}(y))$
SV	-1072.9	-1072.9	-1072.9	-1072.9
SV Lev.	-1065.0	-1065.0	-1065.0	-1065.0
SV Out.	-1076.6	-1076.6	-1076.5	-1076.4
SV Lev. Out.	-1069.3	-1069.3	-1069.2	-1069.3

S&P 500 index

Table: S&P 500 data: Estimated posterior means and standard deviations for all four stochastic volatility models.

Parameter	SV		SV Lev.		SV Out.		SV Lev. Out	
	Mean	S. Dev.	Mean	S. Dev.	Mean	S. Dev.	Mean	S. Dev.
μ	-0.4329	1.2314	-0.5642	0.1500	-0.1786	2.2812	-0.5756	0.3497
ϕ	0.9879	0.0097	0.9811	0.0063	0.9907	0.0086	0.9830	0.0065
τ^2	0.0142	0.0068	0.0106	0.0037	0.0116	0.0053	0.0091	0.0034
ρ	———	———	-0.7608	0.0960	———	———	-0.7652	0.0960

Conclusion

- The partially or fully adapted particle filter can be much more efficient than the standard particle filter.
- Adaptive Metropolis Hastings algorithms can be much more efficient than the usual optimal random walk methods since the cost of constructing a good adaptive proposal is negligible compared to the cost of evaluating the simulated likelihood.
- We also show that the marginal likelihood of any state space model can be obtained in an efficient and unbiased manner by using the particle filter making model comparison straightforward.

References

Andrieu, C., Doucet, A., and Holenstein, R. (2010), “Particle Markov chain Monte Carlo methods,” *Journal of the Royal Statistical Society, Series B*, 72, 1–33.

Del Moral, P. (2004), *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*, New York: Springer.

Giordani, P. and Kohn, R. (2010), “Adaptive independent Metropolis-Hastings by fast estimation of mixture of normals,” *Journal of Computational and Graphical Statistics*, 19, 243–259.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. (1993), “A novel approach to non-linear and non-Gaussian Bayesian state estimation,” *Radar and Signal Processing, IEE Proceedings F*, 140, 107–113.

Kim, S., Shephard, N., and Chib, S. (1998), “Stochastic volatility: Likelihood inference and comparison with ARCH models,” *Review of Economic Studies*, 65, 361–393.

Pitt, M. and Shephard, N. (1999), “Filtering via simulation: auxiliary particle filter,” *Journal of the American Statistical Association*, 94, 590–599.

— (2001), “Auxiliary variable based particle filters,” in *Sequential Monte Carlo Methods in Practice*, eds. de Freitas, N., Doucet, A., and Gordon, N. J., New York: Springer-Verlag, pp. 273–293.

Pitt, M. K. (2002), “Smooth particle filters for likelihood evaluation and maximization,” .

Roberts, G. O. and Rosenthal, J. S. (2009), “Examples of adaptive MCMC,” *Journal of Computational and Graphical Statistics*, 18, 349–367.

Shephard, N. and Pitt, M. (1997), “Likelihood analysis of non-Gaussian measurement time series,” *Biometrika*, 84, 653–667.

Thank you!

Marginal likelihood evaluation using bridge and importance sampling

Suppose that $q(\theta)$ is an approximation to $p(\theta|y)$ which can be evaluated explicitly. Bridge sampling (Meng and Wong, 1996) estimates the marginal likelihood as follows. Let

$$t(\theta) = \left(\frac{p(y|\theta)p(\theta)}{U} + q(\theta) \right)^{-1},$$

where U is a positive constant. Let

$$A = \int t(\theta)q(\theta)p(\theta \mid y)d\theta \ . \qquad \text{Then,}$$

$$A = \frac{A_1}{p(y)} \qquad \text{where} \qquad A_1 = \int t(\theta)q(\theta)p(y \mid \theta)p(\theta)d\theta \ . \tag{12}$$

Marginal likelihood evaluation using bridge and importance sampling

Suppose the sequence of iterates $\{\theta^{(j)}, j = 1, \ldots, M\}$ is generated from the posterior density $p(\theta|y)$ and a second sequence of iterates $\{\tilde{\theta}^{(k)}, k = 1, \ldots, M\}$ is generated from $q(\theta)$. Then

$$\widehat{A} = \frac{1}{M}\sum_{j=1}^{M} t(\theta^{(j)})q(\theta^{(j)}), \quad \widehat{A}_1 = \frac{1}{M}\sum_{k=1}^{K} t(\tilde{\theta}^{(k)})p(y|\tilde{\theta}^{(k)})p(\tilde{\theta}^{(k)}) \quad \text{and} \quad \widehat{p}_{BS}(y) = \frac{\widehat{A}_1}{\widehat{A}}$$

are estimates of A and A_1 and $\widehat{p}_{BS}(y)$ is the bridge sampling estimator of the marginal likelihood $p(y)$.

In adaptive sampling, $q(\theta)$ is the mixture of normals proposal.

Although U can be any positive constant, it is more efficient if U is a reasonable estimate of $p(y)$. One way to do so is to take $\hat{U} = p(y|\theta^*)p(\theta^*)/q(\theta^*)$, where θ^* is the posterior mean of θ obtained from the posterior simulation.

Marginal likelihood evaluation using bridge and importance sampling

An alternative method to estimate of the marginal likelihood $p(y)$ is to use importance sampling based on the proposal distribution $q(\theta)$ (Geweke, 1989; Chen and Shao, 1997). That is,

$$\widehat{p}_{IS}(y) = \frac{1}{K}\sum_{k=1}^{K}\frac{p(y|\theta^{(k)})p(\theta^{(k)})}{q(\theta^{(k)})}.$$

Since our proposal distributions have at least one heavy tailed component, the importance sampling ratios are likely to be bounded and well-behaved, as in the examples in this paper.