SMC^2: an efficient algorithm for sequential analysis of state-space models
Supplement

N. CHOPIN (CREST-ENSAE),
P.E. JACOB (CEREMADE, Université Paris Dauphine),
O. PAPASPILIOPOULOS (Universitat Pompeu Fabra)

This supplement is organised as follows. Section 1 provides additional graphs and comments for the volatility example in Section 4.1 of the paper. Sections 2 and 3 treat two additional examples that were not covered in the paper. Section 2 considers the local level model. Section 3 considers a population growth model taken from Peters et al. (2010).

For the volatility example, the algorithmic parameters (ESS threshold, number of particles, and so on) are exactly those given in the paper; in fact these additional plots were obtained from the same runs as those reported in the paper. For the two extra examples, we experimented with alternative algorithmic parameters. Unless specified otherwise, the ESS criterion was set to 80%, and we used Particle Marginal Metropolis–Hastings moves with a random walk proposal. The random walk is Gaussian with variance equal to $c\Sigma$ where Σ is the variance of the current particles and c is set to 10%. We use a reflective random walk: if the prior distribution is defined on $[a, \infty[$ for instance, and if the proposed value y is lower than a, we propose $y^* = a + |y - a| = 2a - y$ instead of y. This results in a cost-free improvement of the acceptance ratio. Only one PMMH move for each θ-particle is performed at each resample-move step, except when specified otherwise in Section 3.

These simulations results can be reproduced using the generic software package py-smc2, written in Python and C, available at http://code.google.com/p/py-smc2/.

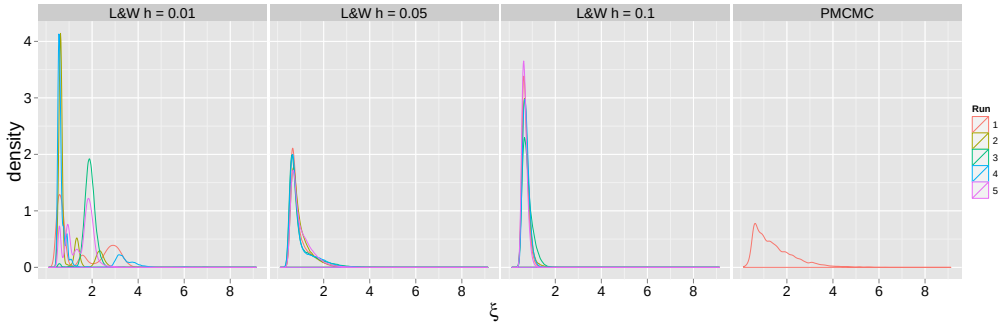
1 Additional plots and comments for volatility example

1.1 Comparison to Liu and West's algorithm

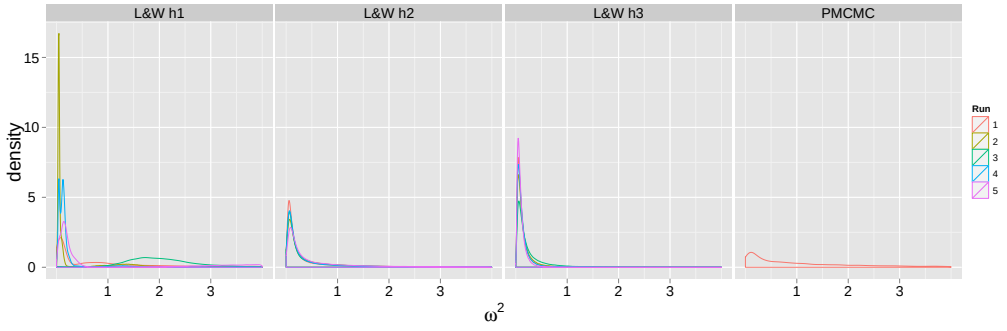
In this section, we provide more details and more plots on the comparison between SMC^2 and L&W, ie. Liu and West (2001)'s algorithm. We recall that L&W is a particle filter such that the hidden states are extended to include the parameters (as in SOPF), and that the θ-components of the particles are diversified using a Gaussian move that leaves the variance of the particle sample unchanged. Compared to the general method described in Liu and West (2001), we do not implement any look-ahead scheme in the spirit of the auxiliary particle filter, and hence the proposal distribution for the x-components is set to $f_\theta(x_{t+1}|x_t)$. The θ-components are updated using a Normal distribution as follows: at time $t+1$ the θ-component of the m-th particle is drawn from $\mathcal{N}(\cdot|m_t^m, h^2 V_t)$, where the mean is $m_t^m = (1-a)\theta^m + a\bar{\theta}_t$ with $a \in [0,1]$, i.e. an average between the particle and the empirical mean at the previous time, and the variance is defined by a product between a smoothing parameter h and the empirical variance V_t at time t. By setting $a = \sqrt{1-h^2}$, the move step guarantees that the mean and the variance of the θ-component remains constant. In the end the particles are indeed diversified, but the Gaussian move does not keep the distribution of the particles invariant: it only guarantees the invariance of the first two moments, and the method is thus biased.

We have already seen in the paper that L&W provides significantly biased results regarding the posterior marginal distributions, see Fig. 4 of the paper and comments around. Figure 1 show the same phenomenon for other parameters, and other choices of the smoothing parameter h. We see

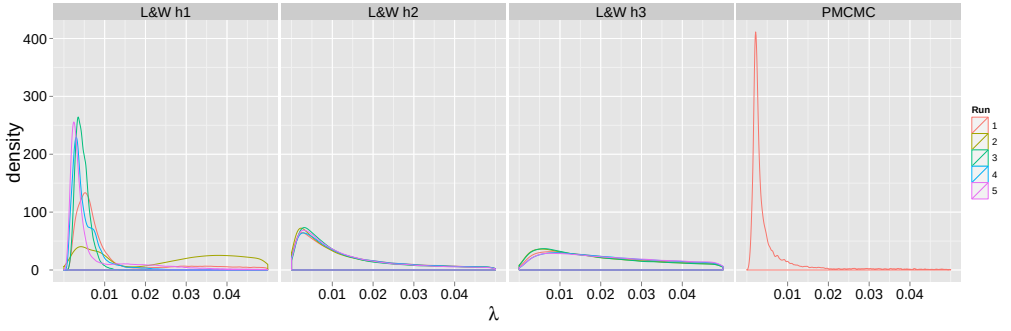
that for the smallest value h, the variation between runs becomes important, while the bias is still large compared to the distribution estimated using PMCMC.



(a)



(b)

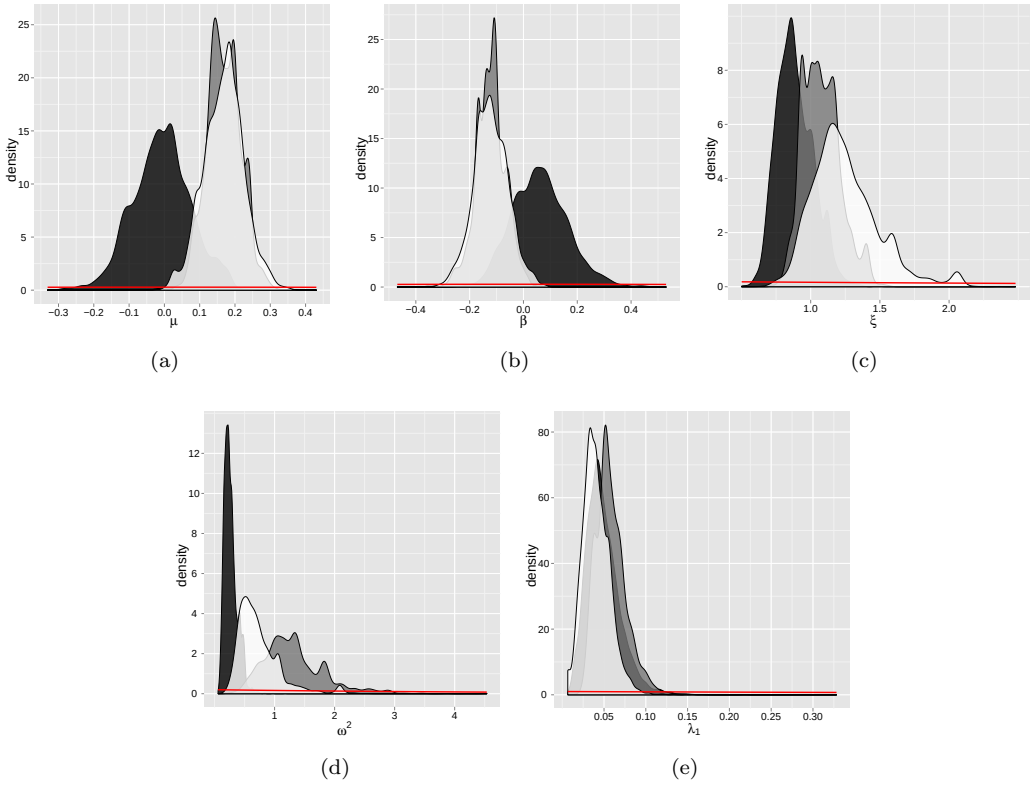


(c)

Figure 1: Single-factor stochastic volatility model, synthetic dataset. Posterior distribution obtained using L&W for parameters ξ, ω^2, and λ for various values of h: $h_1 = 0.01$, $h_2 = 0.05$ and $h_3 = 0.1$. For each value of h, 5 independent runs are represented by overlaid estimated density curves. The PMCMC posterior distribution is shown on the right for reference.

1.2 Model comparison

The plots in this section corresponds to the analysis of the SP500 dataset with three different models (single factor, double factor without leverage, double factor with leverage), as explained at the end of Section 4.1 of the paper. Figures 2 and 3 summarize the estimation of the posterior distribution of the parameters, with kernel density estimates computed using all three runs pooled together.



(a) (b) (c) (d) (e)

Figure 2: Volatility example, SP500 data, (a) to (e): prior (in red) and posterior distributions of common parameters across three different models: single factor (white), two-factor without leverage (grey) and full model (black).

In terms of volatility estimation, our findings are in accordance with those of Griffin and Steel (2006) which have analysed the S&P 500 over a different period. The inclusion of leverage terms causes significant shift in the distribution of β, which concentrates mostly on positive values. This is financially more sensible - the skewness is explained by the leverage term. Additionally, there are shifts in the distribution of λ_2 and there is more weight on the slow-volatility component.

2 Local level

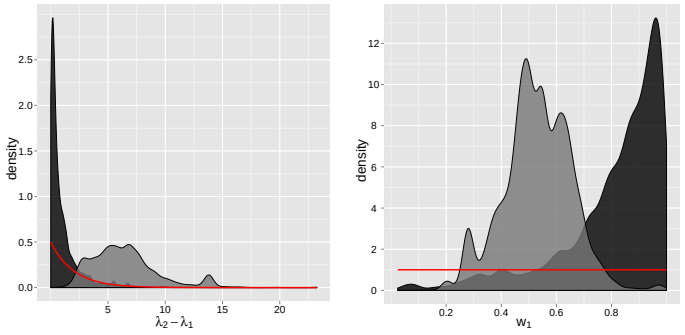
The local level model, with parameter $\theta = (\sigma_y^2, \sigma_x^2)$, is defined as follows:

$$\begin{cases} y_t & = x_t + \sigma_y \varepsilon_t, & \varepsilon_t \sim \mathcal{N}(0,1), \\ x_{t+1} & = x_t + \sigma_x \eta_t, & \eta_t \sim \mathcal{N}(0,1), \\ x_0 & \sim \mathcal{N}(0,1). \end{cases}$$

This is a linear Gaussian model, hence the Kalman filter (Kalman and Bucy, 1961) can be used to compute the likelihood. Thus, this is essentially toy example, which we use to validate the method.

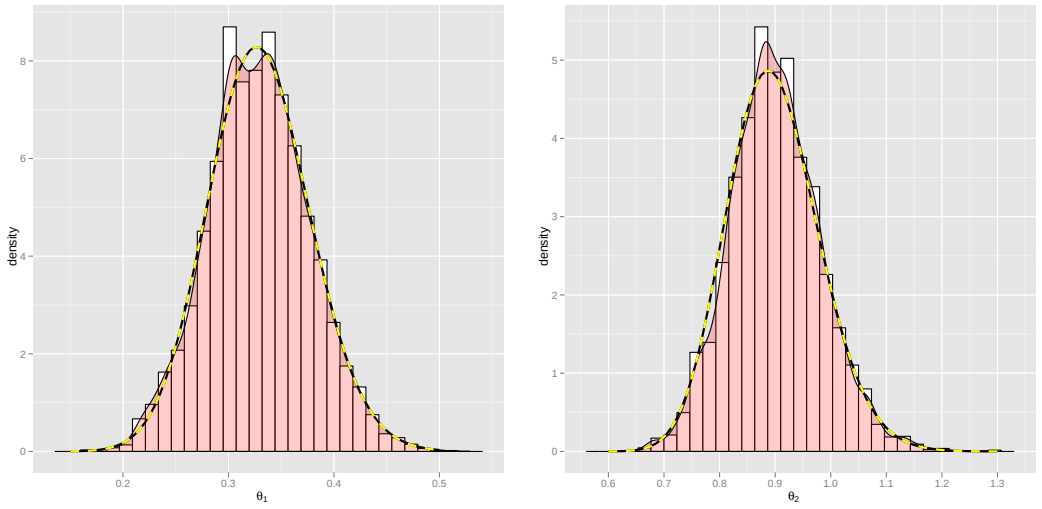
2.1 First study

We take standard inverse-gamma priors: $\sigma_y^2 \sim \mathcal{IG}(2,1)$, $\sigma_x^2 \sim \mathcal{IG}(2,1)$, and simulate $T = 1000$ observations with $\sigma_y^2 = 0.4$ and $\sigma_x^2 = 0.8$. Figure 4 compares the histogram of the final θ-particles produced with the SMC^2 algorithm, with $N_\theta = 5000$ and $N_x = 1000$, with the true marginal posterior densities. (Since the joint posterior is tractable, these two marginal densities are obtained



(a) (b)

Figure 3: Volatility example, SP500 data: prior (in red) and posterior distributions of parameters common to the the two-factor model without leverage (grey) and the the full model (black).



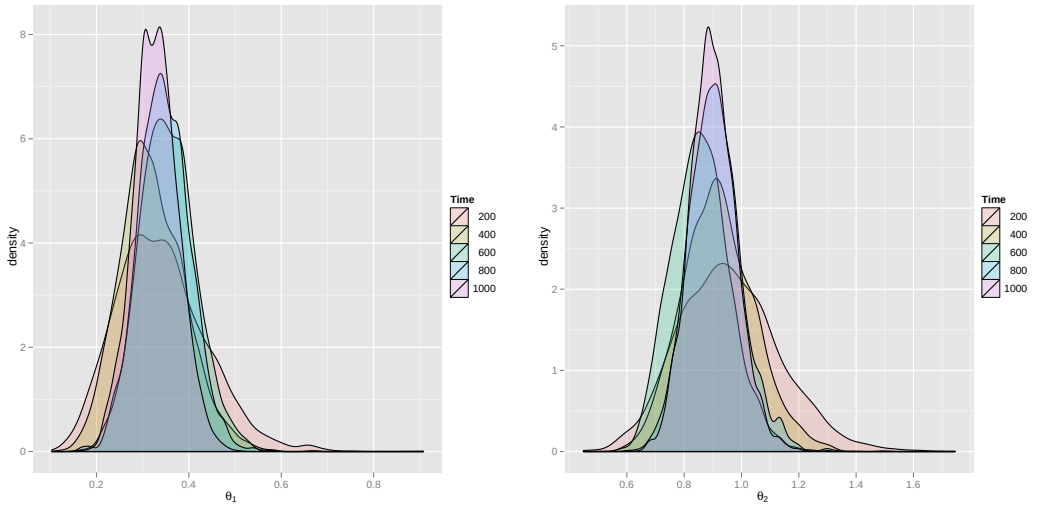
(a) σ_y^2 (b) σ_x^2

Figure 4: Local level example: Histogram of the θ-particles, kernel density estimator (full line) and the true posterior density (dashed line) computed with the Kalman filter, at time $T = 1000$.

by quadrature.) Figure 5 gives the posterior approximations at times $t = 200$, 400, 600, 800 and 1000, obtained successively with the same run of the algorithm.

To monitor the SMC^2 algorithm we can plot the effective sample size (ESS) along the iterations, as well as the acceptance rates of the move steps. These are shown in Figure 6. Since the ESS threshold is here set at 80% and $N_\theta = 5000$, a resample-move is triggered when ESS goes below 4000, as can be seen on the left plot of Figure 6.

As expected, the frequency of the resample-move steps seems to decrease over time. On the other hand, the acceptance rate decreases along the iterations, although it stays above 20% until time $T = 1000$. This is is a simple example, where setting N_x to a fixed, sufficiently large value is an expedient solution, but we shall see in the next example that a dynamic calibration of N_x may be particularly beneficial.



(a) σ_y^2 (b) σ_x^2

Figure 5: Concentration of the posterior mass in the Local level example: kernel density estimations of the posterior distribution approximated by SMC^2, at times $t = 200$, 400, 600, 800 and 1000.

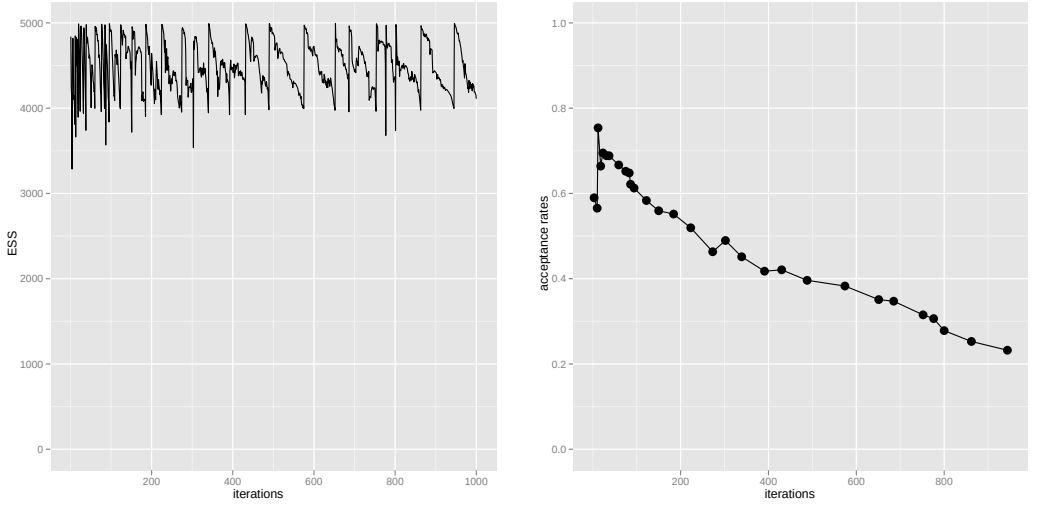


Figure 6: Local level model, first dataset ($T = 1000$): ESS along the iterations (left), and acceptance rate at each move step (right).

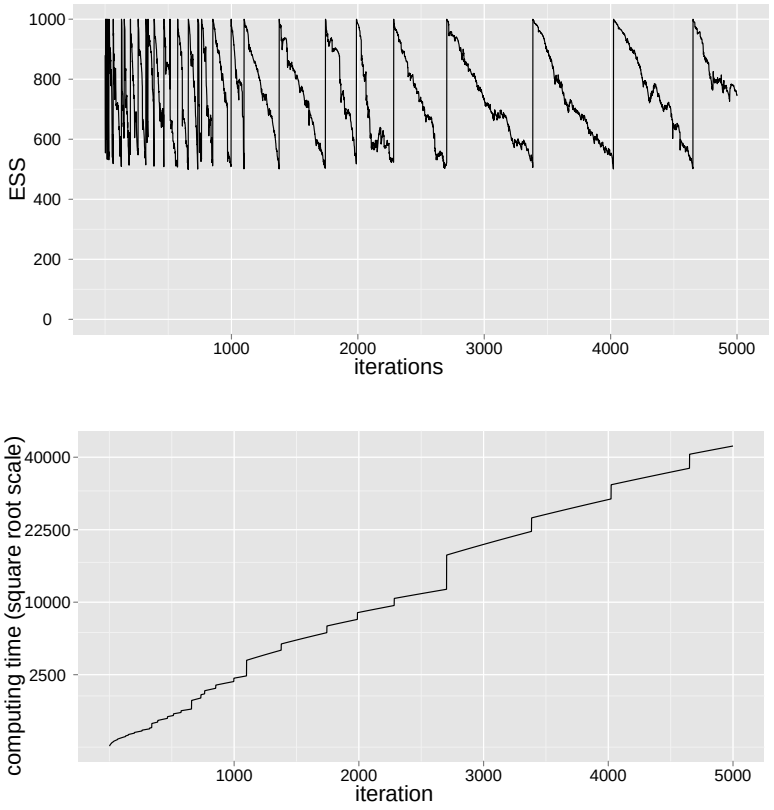


Figure 7: Local level, longer dataset ($T = 5000$): ESS along the iterations (left), and acceptance rate at each move step (right).

2.2 Frequency of resample-move steps in the long run

To better evaluate the long-term behaviour of SMC^2, when the automatic calibration of N_x is used, we simulate a longer dataset, i.e. $T = 5000$, from the same model and the same parameter values as in the previous section. The number of $x-$particles N_x is started at value 100, and increased dynamically as explained in Section 3.6 of the paper. Figure 7 plots both the ESS (Effective sample size, see (5) in the paper), and the CPU time (wall time) as a function of the number of iterations. Note that a square root scale is used for the latter plot.

These plots supports the theoretical discussion of the complexity of SMC^2 in Section of the paper, that is that the frequency of the resample-move steps decreases at a logarithmic rate, and that the cost of the algorithm is quadratic in the number of processed datapoints.

3 Ecological non-linear state space models

Peters et al. (2010) discuss four models to describe the evolution of animal population sizes. One of the models, called "theta-logistic", is the following:

$$\begin{cases} y_t & = n_t + \sigma_w \varepsilon_t \\ \log n_{t+1} & = \log n_t + b_0 + b_2 (n_t)^{b_3} + \sigma_\epsilon \eta_t \\ \log n_0 & = \mu_0 \end{cases}$$

Like Peters et al. (2010) we include the initial state μ_0 in the parameter θ to estimate. The parameter is therefore: $\theta = (\sigma_\epsilon^2, \sigma_w^2, \mu_0, b_0, b_2, b_3)$, and we use the following prior distribution:

$$b_0, b_2, b_3 \sim \mathcal{N}(0, 1) \quad \sigma_\epsilon^2, \sigma_w^2 \sim \mathcal{IG}(2, 1) \quad \mu_0 \sim \mathcal{N}(0, 4)$$

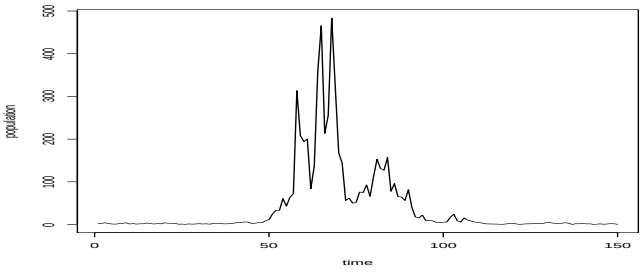
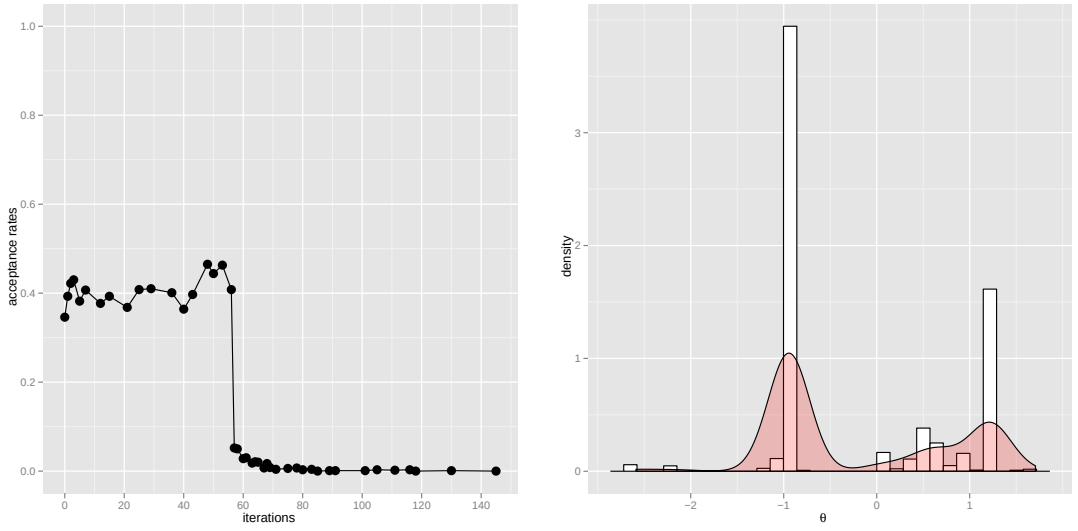


Figure 8: Ecological example: synthetic data set with T = 150 observations, generated from the theta-logistic model.



(a) Acceptance rates

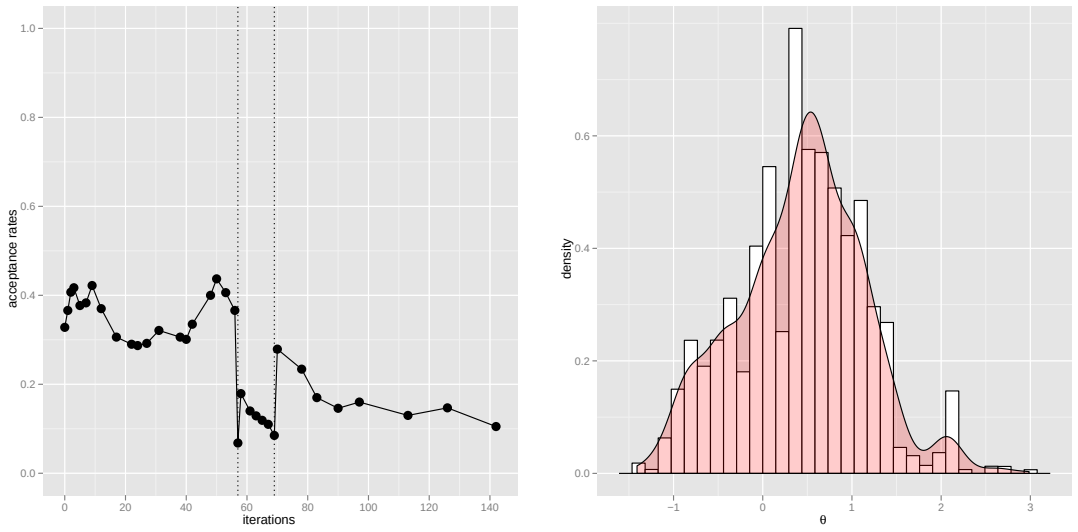
(b) μ_0

Figure 9: Ecological example, fixed number $N_x = 1000$ of x-particles: acceptance rates along the iterations (left) and histogram of the final posterior distribution for $\theta_3 = \mu_0$ (right).

which is the same as in Peters et al. (2010), except that Peters et al. (2010) use inverse-gamma distributions with parameters $\alpha = T/2$, $\beta = 2(\alpha - 1)/10$.

3.1 Simulated dataset, automatic adjustment of N_x

Like Peters et al. (2010) we generate a synthetic data set using $\sigma_\epsilon^2 = 0.47^2$,$\sigma_w^2 = 0.39^2$, $\mu_0 = \log(1.27)$, $b_0 = 0.15$, $b_2 = -0.125$, $b_3 = 0.1$; see Figure 8. On this synthetic data set with $T = 150$ points, we first use $N_\theta = 1000$ and $N_x = 1000$. The reason for these large values is twofold: (1) T is small, hence we can use large values of N_x and N_θ without making the algorithm unreasonnably costly to run, and (2) looking at the observations on Figure 8, which presents huge pikes between $t = 50$ and $t = 100$, a large value for N_x should help to get decent acceptance rates after $t = 50$. The acceptance rates are plotted on Figure 9, on the left side. The right side shows the resulting approximation of the posterior distribution of $\theta_3 = \mu_0$. Since the acceptance rates are very low, SMC behaves like a sequence of importance sampling steps without any diversification of the particles, leading to a strong degeneracy of the particles. The plot of Figure 9 shows that the acceptance rates become really low after the first pikes around $t = 60$. Hence the number of x-particles was set too low. We use this opportunity to illustrate the automatic adjustment of N_x.



(a) Acceptance rates

(b) μ_0

Figure 10: Ecological example, automatic adjustment of N_x: acceptance rates along the iterations (left) with vertical dotted lines when the increases of N_x occur, and histogram of the final posterior distribution for μ_0 (right).

The strategy here is to double the number of x-particles when the acceptance rate goes below 10%, and we initialize the algorithm with the same value $N_x = 1000$. The results of the SMC^2 with automatic adjustment of N_x are presented on Figure 10, and must contrasted wih Figure 9.

As shown on Figure 10 the acceptance rate goes first below 10% at time $t = 57$. At this time N_x goes from 1000 to 2000. Then the acceptance rate becomes higher for a few iterations and goes below 10% again at time $t = 69$. The number N_x is doubled again, and stays at $N_x = 4000$ until the final time $T = 150$. The right hand side of the figure shows the final histogram of μ_0, which shows much less degeneracy than the histogram of Figure 9.

We present the kernel density estimation of the posterior distribution of each of the six parameters, at $t = 50$, $t = 100$ and $T = 150$, in Figure 11. Note that since N_x has changed twice between $t = 50$ and $t = 100$, the approximation of the posterior distribution uses 1000 x-particles at time $t = 50$ and 4000 x-particles at time $t = 100$ and $T = 150$.

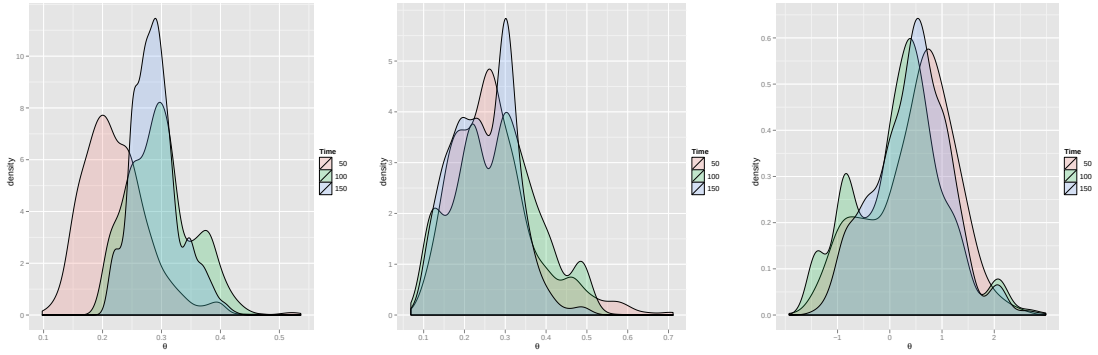
3.2 Real data set *Accipiter nisus* and comparison with PMMH

The real data set is a time series of a population of sparrowhawks (*Accipiter nisus*) in South Scotland, well studied in the literature and taken as an example by Peters et al. (2010). The data set is numbered 6575 in the Global Population Database (NERC (1999)). Peters et al. (2010) among others find that the posterior distribution of the parameters is strongly multimodal for this data set. One of our objectives of this example is to see whether SMC^2 finds the modes more easily than a PMCMC method, at a fixed computational cost. We use the same prior as previously for the synthetic dataset.

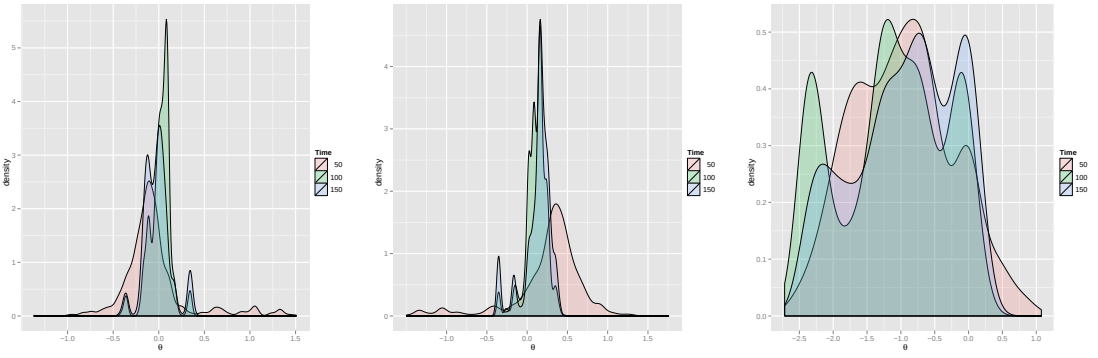
We compare SMC^2 with the adaptive PMMH algorithm implemented in Peters et al. (2010). It consists in a modified PMMH where the covariance of the random walk proposal of the Metropolis–Hastings sampler depends on the past values of the Markov chain (θ_t). We use their mixture proposal:

$$q(\theta^*|\theta_t) = w_1 \mathcal{N}\left(\theta^* \left| \theta_t, \frac{(2.38)^2}{d}\Sigma_t \right.\right) + w_2 \mathcal{N}\left(\theta^* \left| \theta_t, \frac{(0.1)^2}{d} I_{d,d} \right.\right)$$

where Σ_t is the empirical covariance matrix of the past values of the chain $(\theta_1, \dots \theta_t)$, d is the



(a) σ_ϵ^2 (b) σ_w^2 (c) μ_0



(d) b_0 (e) b_2 (f) b_3

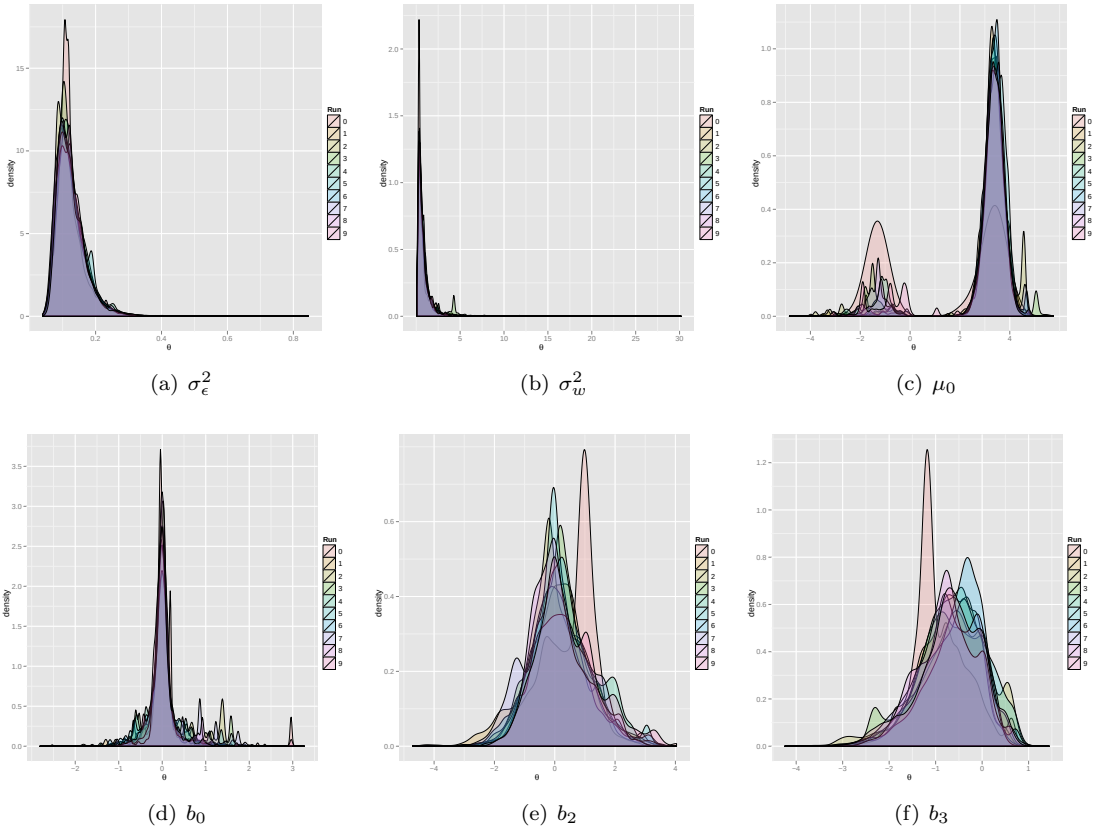
Figure 11: Concentration of the posterior mass: kernel density estimators at times $t = 50$, 100, and 150, obtained with the SMC^2 algorithm with automatic adjustment of N_x.

dimension of the parameter space and 2.38, 0.1 are values justified by Roberts and Rosenthal (2009) and $I_{d,d}$ is the $d \times d$ identity matrix. Hence the first component of the mixture is adaptive and the second component has a constant variance. In the simulations we use $w_1 = w_2 = 1/2$.

The results are to be compared to Figure 9 of Peters et al. (2010), though we use a slightly different prior on $\sigma_w^2, \sigma_\epsilon^2$, as mentioned in the beginning of section 3. For the SMC^2 algorithm we use $N_x = 500$ (as in Peters et al. (2010)) and $N_\theta = 5000$. The algorithm performs 3 move steps at each resample-move, which results in around $200 \times N_x \times N_\theta$ evaluations of the likelihood function $g_\theta(y_t|x_t)$. Since there is $T = 18$ observations and $200 \approx 10 \times T$, we run the adaptive PMMH algorithm with $N_x = 500$ and $n = 10 \times N_\theta = 50,000$ iterations. The starting point of each chain is generated from the prior distribution. We discard the first $10,000$ iterations of each run, as burn-in. With this choice of N_x, the acceptance rate is around 20%.

Here we match the computational cost of both methods in terms of number of calls to the weight function. In terms of memory, a non-adaptive PMMH algorithm only needs to store the last accepted value of the parameter and the corresponding x-particles, hence it is much lighter than SMC^2.

The results for 10 independent runs of SMC^2 are shown on Figure 12. The plots show that for some parameters (especially μ_0), the posterior distribution is indeed multimodal and that all the 10 SMC^2 runs have at least visited the various modes, even though the approximation of the target density seems very variable from one run to another.

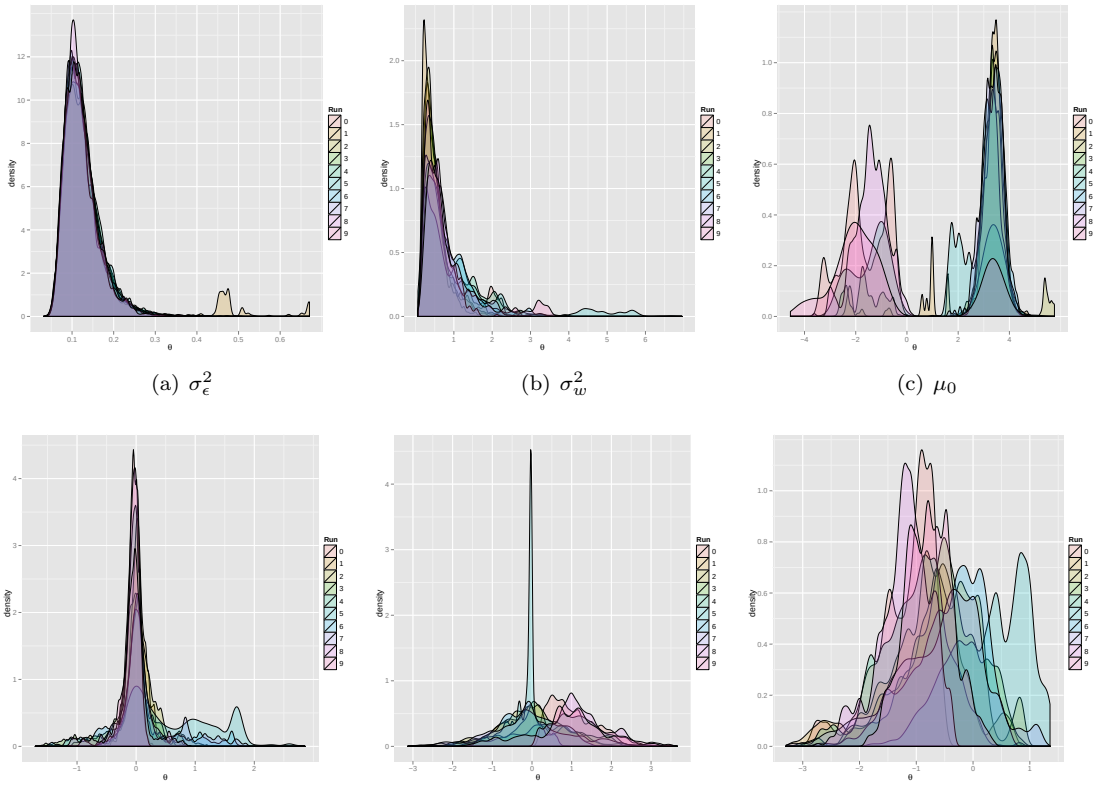


(a) σ_ϵ^2 (b) σ_w^2 (c) μ_0

(d) b_0 (e) b_2 (f) b_3

Figure 12: Ecological example, with real dataset: overlaid kernel density estimations of the posterior distribution estimated by SMC^2, for 10 independent runs.

If we plot the approximation of the target density by 10 independent runs of the adaptive PMMH algorithm, we obtain Figure 13. These plots show that the chains have not all explored the various modes. Especially for μ_0, some chains have been trapped in the negative values while some chains have stayed in the positive values. Hence for this example the chains did not mix well



(a) σ_ϵ^2 (b) σ_w^2 (c) μ_0

(d) b_0 (e) b_2 (f) b_3

Figure 13: Ecological example, with real dataset: overlaid kernel density estimations of the posterior distribution estimated by PMMH, for 10 independent runs.

even after $50,000$ iterations.

This phenomenon appears clearly when we look at trace plots of the Markov chain, for μ_0, in Figure 14. The plot shows that the chains swap only very rarely from one mode to the other, and some chains never visit more than one mode.

The presence of local modes makes the estimation based on the SMC^2 particles more precise than the estimation based on the PMMH chains, as can be seen on the box-plots of Figure 15, based on 10 runs from each method. The box-plots represent the posterior means of the parameters computed for each of the runs. The variance of the results is particularly different for μ_0, since all the SMC^2 particle filters have a lot of mass in the main mode (in the positive values), while some of the chains have only visited the minor mode (in the negative values), as seen on the previous plots.

With enough iterations, the Markov chain generated by the adaptive PMMH algorithm is eventually able to visit all the modes and the posterior distribution can be well recovered, as in Peters et al. (2010) who use $50,000$ iterations as burn-in, and $150,000$ post burn-in, on top of a preliminary exploration of the target distribution using an annealed sequence of distributions. However our point is that for a given computational cost, the SMC^2 algorithm seems to better explore the modes of the target distribution than the PMMH.

References

Griffin, J. and Steel, M. (2006). Inference with non-Gaussian Ornstein-Uhlenbeck processes for stochastic volatility. *Journal of Econometrics*, 134(2):605–644.

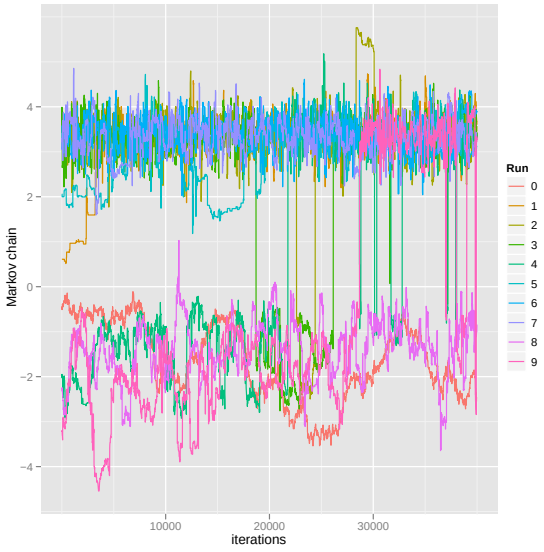


Figure 14: Trace plot of the $40,000$ iterations post burn-in of 10 Markov chains targetting the posterior distribution of μ_0, generated by adaptive PMMH.

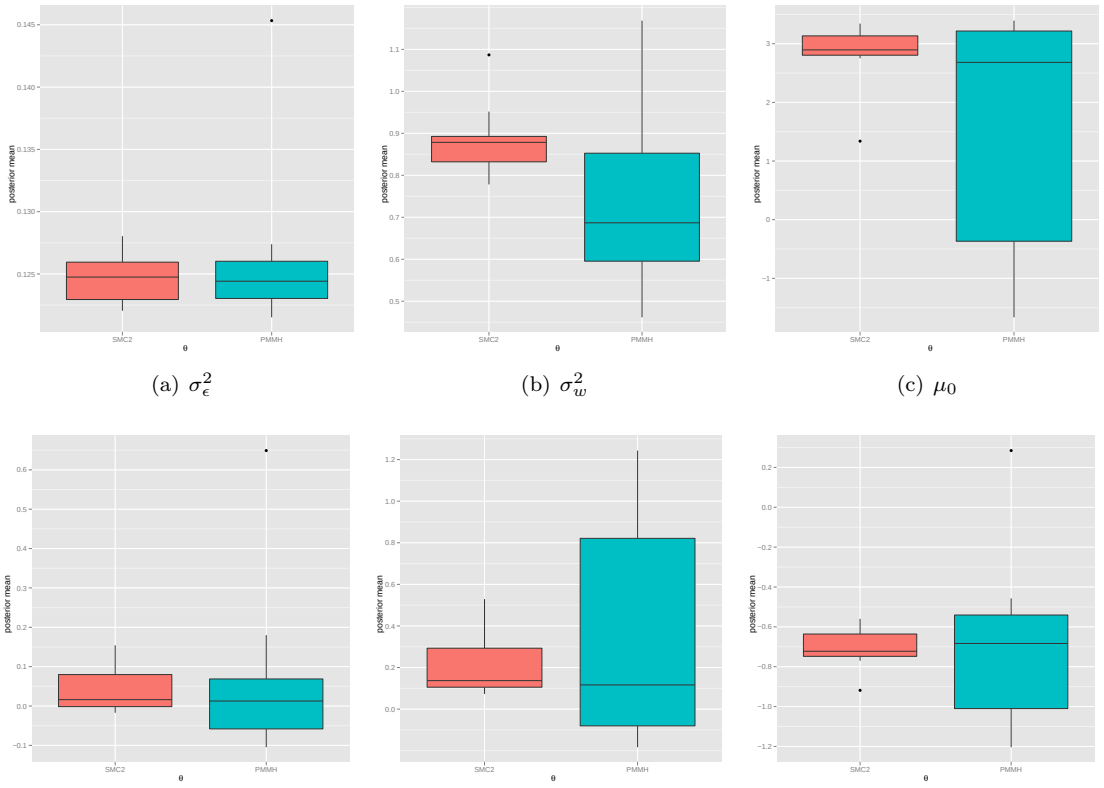
Kalman, R. E. and Bucy, R. S. (1961). New results in linear filtering and prediction theory. *Trans. Amer. Soc. Mech. Eng., J. Basic Eng.*, 83:95–108.

Liu, J. and West, M. (2001). Combined parameter and state estimation in simulation-based filtering. In Doucet, A., de Freitas, N., and Gordon, N. J., editors, *Sequential Monte Carlo Methods in Practice*, pages 197–223. Springer-Verlag.

NERC (1999). Centre for population biology, imperial college, the global population dynamics database. http://www.sw.ic.ac.uk/cpb/cpb/gpdd.html.

Peters, G., Hosack, G., and Hayes, K. (2010). Ecological non-linear state space model selection via adaptive particle markov chain monte carlo. *Arxiv preprint arXiv:1005.2238*.

Roberts, G. and Rosenthal, J. (2009). Examples of adaptive markov chain monte carlo. *Journal of Computational and Graphical Statistics*, 18(2):349–367.



(a) σ_ϵ^2 (b) σ_w^2 (c) μ_0

(d) b_0 (e) b_2 (f) b_3

Figure 15: For each parameter, comparison of the posterior means computed with 10 independent runs of each method. SMC^2 on the left (red) and adaptive PMMH on the right (blue) of each plot.