Comparison of inference performance

(a) RMSE of price(x_1) state

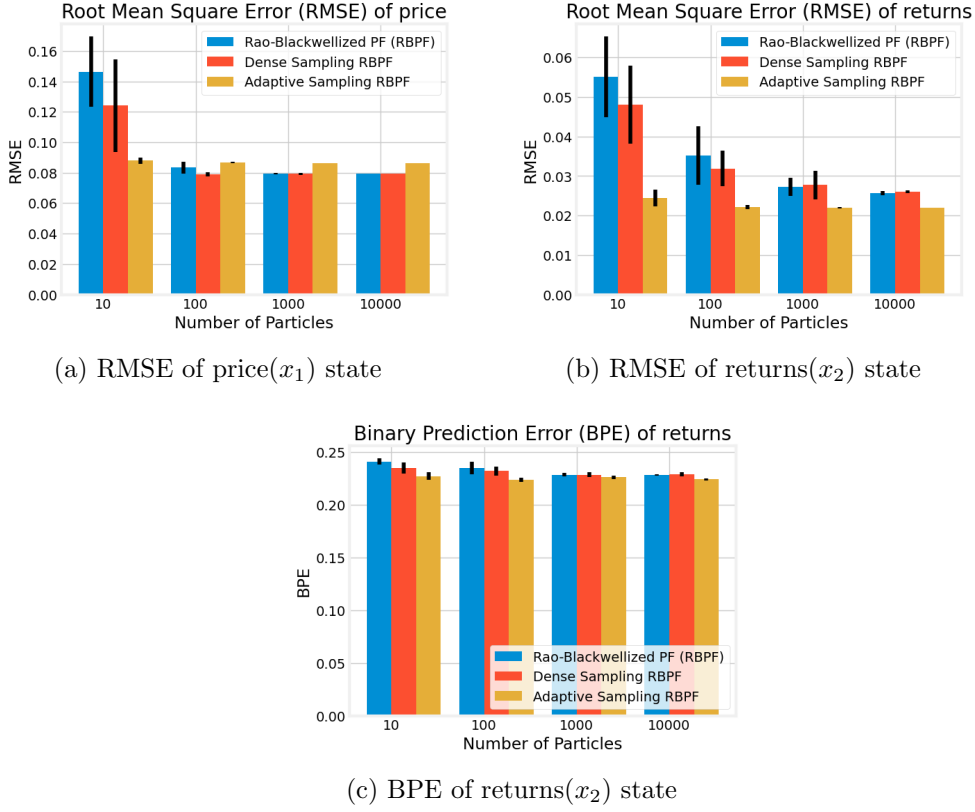(b) RMSE of returns(x_2) state

(c) BPE of returns(x_2) state

Figure 1: Comparison of inference performance between the 3 RBPF-based particle filters as N varies

Figure 1 shows how the performance of the 2 bootstrap particles filters vary as the number of particles, N is changed.

Firstly, we note that the performance of the 3 RBPF-based particle filters is in general better than the bootstrap particle filters across all performance metrics. This suggests that the ability of the RBPF-based particle filters to marginalise out the gaussian components of the hidden state is very useful in improving the inference performance of the particle filter.

Secondly, we note that in general, the performance of the dense sampling RBPF and the adaptive sampling RBPF is slightly better than that of the generic RBPF, suggesting that the dense sampling and adaptive sampling improvements are both useful in improving the inference performance of the generic RBPF. Figures 1a and 1b indicate that the performance of the generic RBPF degrades significantly as the number of particles (N) used drops, likely due to sample impoverishment. However, the dense sampling RBPF and adaptive sampling

RBPF are both more resistant to this degradation in performance, indicating that they are likely less affected by sample impoverishment.

In particular, the adaptive sampling RBPF is extremely resistant to the degradation in performance as the number of particles drops, with good inference performance even with as low as 10 particles.

Dense Sampling RBPF

We now examine the reduction in sample impoverishment due to the dense sampling improvement described in **??**.

Similarly to **??**, we demonstrate the problem of sample impoverishment by simulating a single time step of the particle filter, for varying values of $y_t - \mathbf{CA}\mu_k$. Fixing $\mu_k^{(i)}, \Sigma_k^{(i)}$ whilst varying y_k and N, we simulate one time step of the RBPF update step described above and present the normalised entropy of the RBPF particle weights with and without the dense sampling improvement in Figure **??**.
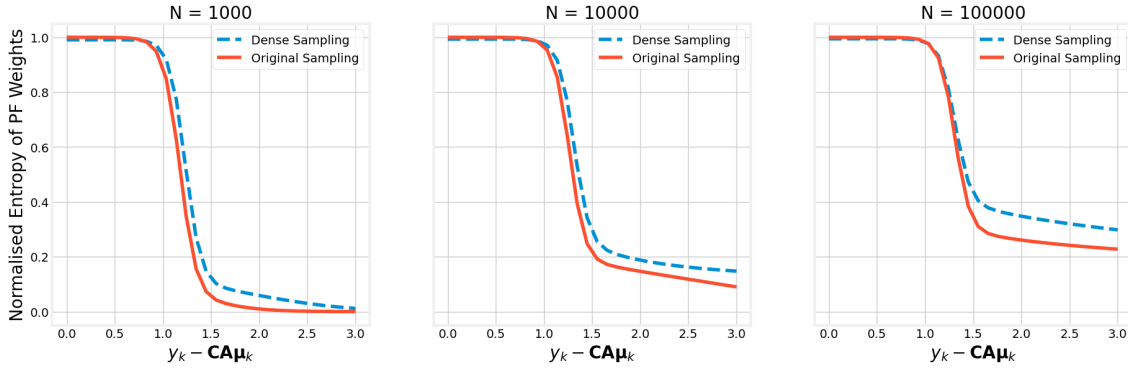

Figure 2: Improvement in normalised entropy of RBPF particle weights using dense sampling with hyperparameters: $(\epsilon_\lambda = 0.1, \, m = 2)$

We can see from Figure **??** that the dense sampling improvement causes the normalised entropy of the RBPF particle weights to increase for larger price outliers $y_t - \mathbf{CA}\mu_k$. A higher normalised entropy indicates that the weights of the RBPF particles are more evenly distributed and hence that there are a larger number of particles with effective weights under the dense sampling improvement.

Adaptive Sampling RBPF

Figure 3 shows the distribution of samples $\lambda_k^{(i)}$ from the Adaptive Sampling RBPF as well as the ground truth λ_k used to generate the simulation data. Figure 4 shows a similar figure
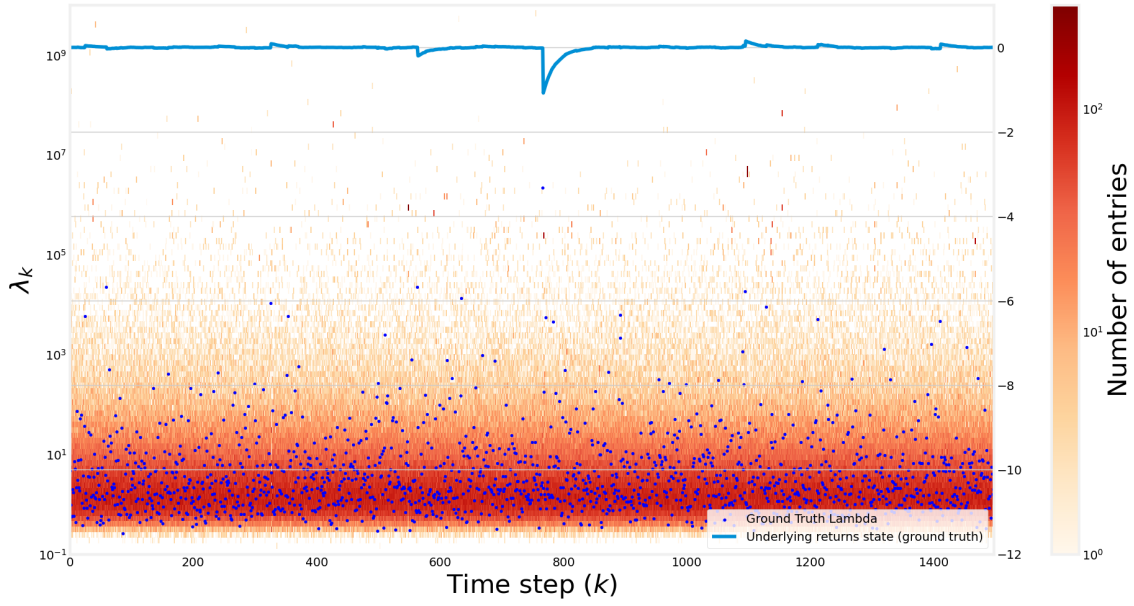
Figure 3: Distribution of samples $\lambda_k^{(i)}$ from the Generic RBPF
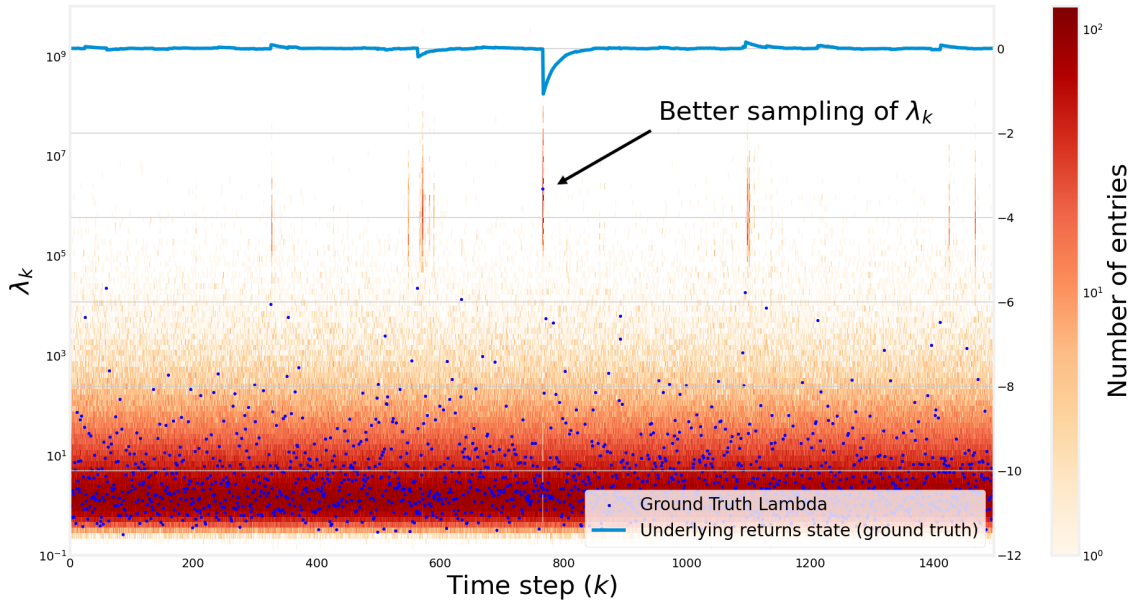

Figure 4: Distribution of samples $\lambda_k^{(i)}$ from the Adaptive Sampling RBPF

for the generic RBPF *after* the resampling step.

Comparing Figures 3 and 4, we can see qualitatively that the distribution of samples from the Adaptive Sampling RBPF is distributed more closely around the ground truth λ_k, resulting in a many more effective samples being generated using the Adaptive Sampling RBPF compared to the generic RBPF.

Note the extremely challenging nature of forming a good importance sampling estimate of the posterior $p(\lambda_k|y_k, \mu_{k-1}, \Sigma_{k-1})$ using only samples from the prior as λ_k ranges over 7 orders of magnitude (10^{-1} to 10^7) in this example. This makes it very difficult to form a good discrete importance sampling estimate, which would require a huge amount of particles to cover the entire possible range of λ_k (which theoretically has infinite support).

With the adaptive sampling improvement, we are able to use the information from the current observation to sample directly from the approximate posterior, which increases the sampling efficiency of the particle filter greatly.

We next investigate how few particles we can use with the new adaptive sampling particle filter before performance degrades unacceptably. Figure 5 shows how the RMSE of the inferred underlying returns process changes for very low values of N. Interestingly, we note that the adaptive sampling RBPF is still capable of good inference performance even with extremely low $N = 4$, unlike the generic RBPF, which suffers from degradation of inference performance when using a low number of particles.
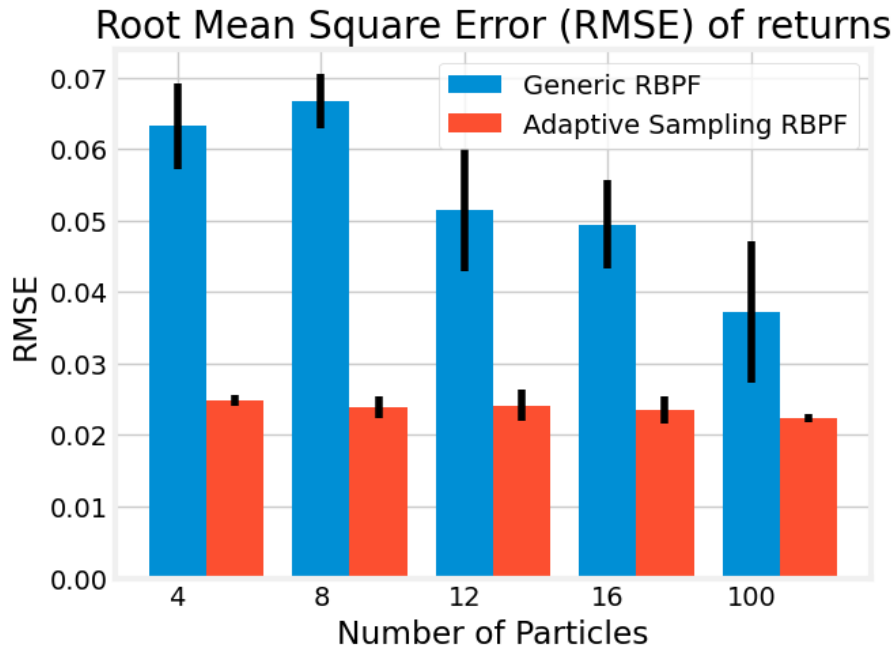

Figure 5: Performance of the RBPF-based particle filters for very low N

List of Figures

List of Tables