# Inference and Filtering for Partially Observed Diffusion Processes via Sequential Monte Carlo

**Ed Ionides**

Department of Statistics

University of Michigan

# Outline

- Partially observed diffusion models.
- A new filter algorithm called Conditional Particle Filter (CPF).
- Implementation of CPF for partially observed diffusions.
- CPF as a Sequential Monte Carlo algorithm.
- Properties of CPF.

# Partially observed diffusion models

- An unobserved vector-valued "state" process $x_t$

$$dx_t = \mu(x_t, t)dt + \sigma(x_t, t)dB_t, \quad t \in [0, T]$$

- A discrete time "observation" process $y_t$

$$y_t = g(x_t, \eta_t), \quad t = 1, 2, \ldots, T$$

  for $\eta_t$ an independent sequence of "observation noise".

- Some generalizations are possible.

- A key to likelihood-based inference is to solve the filtering problem, i.e. to evaluate $f(x_t \mid y_1, \ldots y_t)$.

## Applications

- $x_t$ models a system (usually with some unkown parameters).
- $y_t$ models the available observations.

| | $x_t$ | $y_t$ |
|---|---|---|
| (i) | Population or disease dynamics. | Birth, death, disease status data. |
| (ii) | Meteorological models based on atmospheric science. | Meteorological station, balloon, satellite data. |
| (iii) | Volatility of financial markets. | Market prices, or individual trades. |

## Conditional Particle Filtering (CPF)

- A “conditional particle”, $\hat{x}_t$, is a Markov process conditional on the observations. $\hat{x}_t$ is drawn from the conditional distribution of $x_t$ given $y_t$ and $x_{t-1}{=}\hat{x}_{t-1}$, namely $f(x_t \mid y_t, x_{t-1}{=}\hat{x}_{t-1})$.
- $(\hat{x}_t, W_t)$ is a properly weighted importance sample for the filtering distribution $f(x_t \mid y_1, \ldots y_t)$ if $W_t = \prod_{s=1}^{t} w_s$ for $w_s = f(y_s \mid x_{s-1}{=}\hat{x}_{s-1})$.
- “Properly weighted” means $E[h(\hat{x}_t)W_t] \propto E[h(x_t)|y_1, \ldots, y_t]$.
- In practice, we must truncate: $\hat{W}_t = \prod_{s=t-k}^{t} w_s$.

## Questions

- Can CPF be implemented for partially observed diffusions? (YES)
- How does CPF relate to other filtering strategies?
- Can CPF apply for small or singular observation noise, e.g. when $x_t = (x_{t,1}, x_{t,2})$ and $y_t = x_{t,1}$? (YES)

## Simulating from $\hat{x}_t$

- Recall that $x_t$ solves an SDE, $dx_t = \mu(x_t)dt + \sigma(x_t)dB_t$

  $\hat{x}_t$ also solves an SDE, $d\hat{x}_t = \hat{\mu}(\hat{x}_t)dt + \sigma(\hat{x}_t)dB_t$

- The conditional process, $\hat{x}_t$, is also a diffusion.

- $\hat{x}_t$ and $x_t$ have the same infinitesimal variance $\sigma(\cdot)$.

- $\hat{\mu}$ is difficult to calculate for nonlinear models.

- Approximate $\hat{x}_t$ by $\tilde{x}_t$ with $d\tilde{x}_t = \tilde{\mu}(\tilde{x}_t)dt + \sigma(\tilde{x}_t)dB_t$

  (e.g. by linearization).

## Importance Sampling

- Let $P$, $\hat{P}$ and $\tilde{P}$ be the laws of $x_t$, $\hat{x}_t$ and $\tilde{x}_t$ respectively.
- To sample from $\hat{P}$, we can simulate from $\tilde{P}$ and attach importance weight $d\hat{P}/d\tilde{P}$.
- Importance weights are only needed up to a constant of proportionality, so we calculate $dP/d\tilde{P} \propto d\hat{P}/d\tilde{P}$.
- This is Radon-Nikodym notation for the familiar result

$$f_{X|Y}(x|y) = \frac{f_{XY}(x,y)}{f_Y(y)} \propto f_{XY}(x,y)$$

- In some models, $d\hat{P}/dP$ does not exist, and extra care is required.

## Conditional Particle Filtering (CPF) is a variation of Sequential Monte Carlo (SMC)

- SMC methods are “interacting particle filters”. CPF is a special case where the particles do not interact.
- The original SMC algorithm of Gordon et al (1993) is sometimes called a “particle filter” (PF).
- In CPF, the truncation of weights plays a similar role to the resampling of particles in PF.
- There is a continuum of possible algorithms between CPF and PF. These algorithms extend existing generalizations of SMC (Doucet et al, 2001).

## An Example

- For a linear Gaussian system

$$\begin{aligned} dz_t &= -\alpha z_t dt + dB_t \\ y_t &= z_t + \eta_t \end{aligned}$$

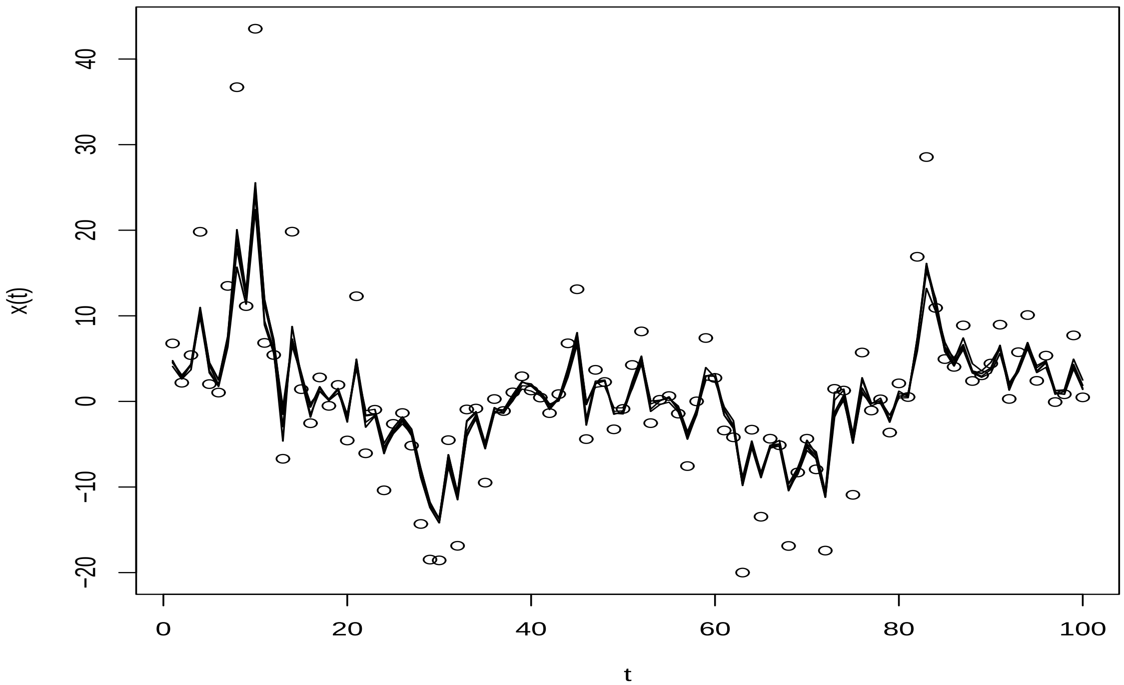
The likelihood of $y_1, \ldots, y_T$ can be found using the Kalman filter (KF).

- We compare algorithms on the nonlinear system $x_t = h(z_t)$ with $y_t = h^{-1}(x_t) + \eta_t$ and $\eta_t \sim N(0, \tau^2)$.
- Here, $h(z) = [(|z| + 1)^2 - 1]sgn(z)$, so that $h$, its inverse and their derivatives exist on $[-\infty, \infty]$.
- Filters were compared by their root mean square error in calculating the likelihood – a quantity we call the “accuracy”. Accuracy can be decomposed into bias, filter variance and Monte Carlo variance.

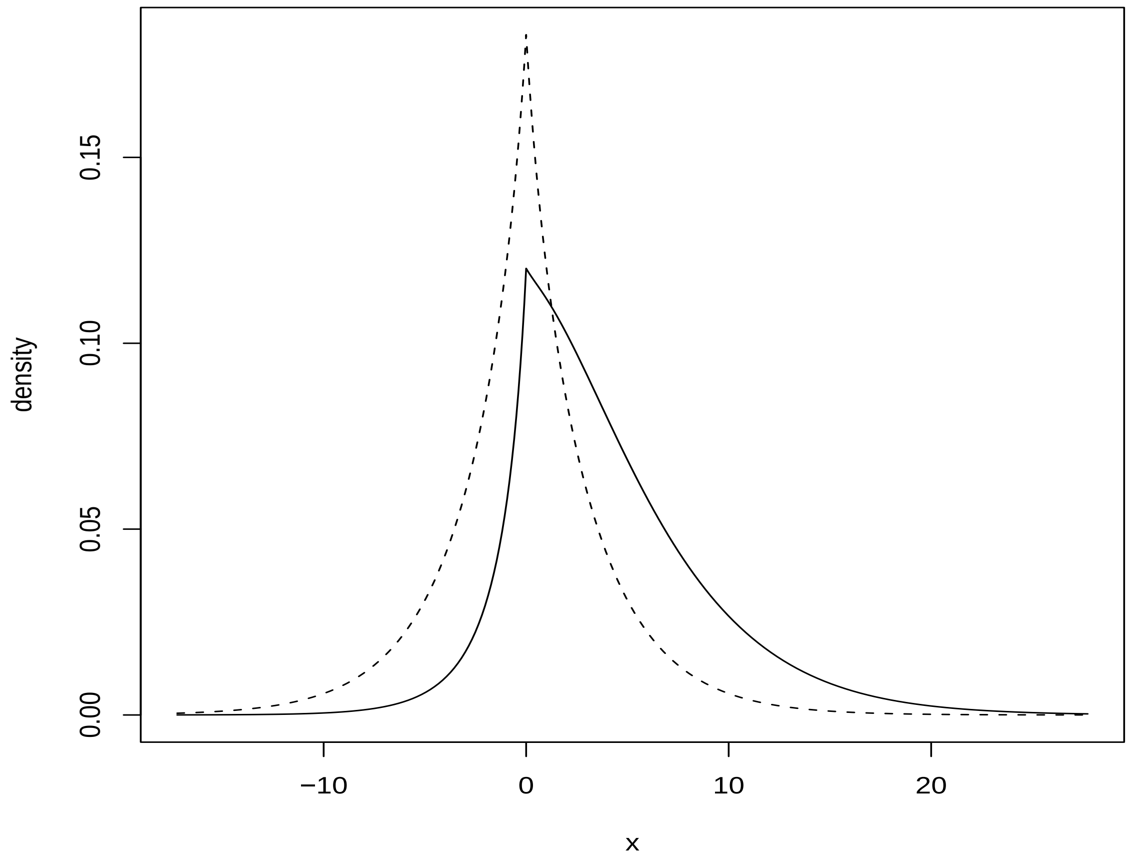
# Table 1.

| | $\tau$ | $b \times 10^2$ | $\sigma_f \times 10^2$ | $\sigma_m \times 10^2$ | $A \times 10^2$ |
|---|---|---|---|---|---|
| EKF | 1 | -3.32 | 25.1 | n/a | 25.4 |
| LCPF | 1 | -1.62 | 15.5 | 4.67 | 16.4 |
| PF | 1 | -0.45 | 3.26 | 8.54 | 9.27 |
| CPF | 1 | -0.16 | 3.94 | 3.99 | 5.67 |
| EKF | 0.25 | -8.1 | 42.7 | n/a | 43.9 |
| LCPF | 0.25 | -5.3 | 30.7 | 2.94 | 31.8 |
| PF | 0.25 | -2.52 | 13.7 | 21.7 | 29.8 |
| CPF | 0.25 | -0.82 | 9.78 | 3.56 | 10.5 |

- Bias, $b$, filter error, $\sigma_f$, Monte Carlo error, $\sigma_m$, and accuracy, $A$.
- Calculated for the conditional particle filter (CPF), linearized conditional particle filter (LCPF), particle filter (PF) and extended Kalman filter (EKF).




- A realization with $T = 100$; transformed observations, $h(y_t)$, are shown as points.
- The filtering means, estimating $x_t$, are shown superimposed for EKF, PF, LCPF and CPF. The methods agree closely on their point estimates of $x_t$ (indistinguishable lines).




- Two realizations of the prediction density,
- Calculated by transforming the analytically tractable linear model.

## Algorithmic parameters for Table 1

| | Description | Required for | Value |
|---|---|---|---|
| $N_p$ | # of particles | PF, LCPF, CPF | 200 |
| $N_t$ | # of steps per observation for numerical solution of SDE | PF, CPF | 5 |
| $N_f$ | # of lags for filtering | LCPF, CPF | 3 |
| $N_r$ | # of trials for importance resampling | CPF | 10 |

# Example (Brownian Bridge)

- For $x_t = B_t$, conditioning on $x_0 = x_1 = y_1 = 0$ makes $\hat{x}_t$ a Brownian bridge, $d\hat{x}_t = -\hat{x}_t(1-t)^{-1}dt + dW_t$.

- Set $P_\delta$, $\hat{P}_\delta$ to be the laws of $x_t$, $\hat{x}_t$ for $t \in [0, 1-\delta]$.

$$\frac{dP_\delta}{d\hat{P}_\delta}(\xi) = \exp\left\{2\int_0^{1-\delta} \frac{\xi_t\, d\xi_t}{1-t} + \int_0^{1-\delta} \frac{\xi_t^2}{(1-t)^2}\, dt\right\}.$$

- $\lim_{\delta\to 0} dP_\delta / d\hat{P}_\delta$ does not exist. This might cause problems for our method: (i) simulate from $\tilde{x}_t \approx \hat{x}_t$, (ii) weight according to $dP/d\tilde{P}$.

- Remarkably (see Table 2), we can still use small $\delta > 0$ and estimate

$$f(y_1|x_0) = E_{\tilde{P}_\delta}\left[f(y_1|x_{1-\delta}{=}\tilde{x}_{1-\delta})\frac{dP_\delta}{d\tilde{P}_\delta}\{\tilde{x}_t\}\right].$$

**Table 2.**

| $\delta$ | $\hat{f}(y_1\|x_0)$ | $dP_\delta/d\tilde{P}_\delta$ | $f(y_1\|\tilde{x}_{1-\delta})$ |
|---|---|---|---|
| 0.2 | 0.399(0.216) | 1.00(2.63) | 0.57(0.28) |
| 0.1 | 0.399(0.235) | 0.92(3.68) | 0.79(0.40) |
| 0.04 | 0.396(0.248) | 0.83(8.60) | 1.24(0.64) |
| 0.02 | 0.398(0.251) | 0.68(18.0) | 1.75(0.91) |

- Monte Carlo estimates, $\hat{f}(y_1|x_0)$, of $f(y_1|x_0)$ using CPF for a Brownian bridge, truncating at $t = 1-\delta$.
- Table shows mean, with SD in parentheses.
- The true value is $f(y_1|x_0) = 1/\sqrt{2\pi} = 0.399$

# Conclusions

- CPF is applicable to a wide class of challenging and important nonlinear filtering problems.
- In an example of a nonlinear system for which an exact filter is available, the conditional particle filter (CPF) was considerably more accurate than the standard particle filter (PF).
- CPF had lower Monte Carlo error than PF, particularly for low observation noise.
- CPF (unlike PF) can be made to work with singular observation noise.
- Currently we have only a basic understanding of why CPF succeeds on models with low or singular observation noise.
- Future work is indicated on truncating weights as a Sequential Monte Carlo technique.

# References

Doucet, A., de Freitas, N., and Gordon, N. J. (2001). *Sequential Monte Carlo Methods in Practice*. Springer, New York.

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

Ionides, E. L. (2004). Inference and filtering for partially observed diffusion processes via sequential Monte Carlo. *Submitted to Journal of Computational and Graphical Statistics*. `http://www.stat.lsa.umich.edu/~ionides/pubs/jcgs.pdf`. Also see Tech. report #405, Univ. of Michigan Dept. of Statistics.

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