Utilizing Model Structure in Particle Filtering

Tianshi Chen, Thomas B. Schön, Henrik Ohlsson and Lennart Ljung

AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

Idea and Contribution

A new particle filter (PF) which utilizes model structure is proposed. We will refer to the new PF as the decentralized PF (DPF). The idea of the DPF is closely related to that of the Rao-Blackwellized PF (RBPF). Similar to the RBPF, the DPF also *splits the filtering problem into two linked filtering sub-problems*. The RBPF solves one of the filtering sub-problems using a PF and the other using an optimal filter, typically a Kalman filter. The DPF on the other hand, handles both of the filtering sub-problems using PFs. By exploiting the *parallelism*, the DPF can shorten the execution time.

Problem Formulation

Consider the following nonlinear discrete-time system

$$\begin{aligned} \xi_{t+1} &= f_t(\xi_t, v_t), \\ y_t &= h_t(\xi_t, e_t). \end{aligned} \tag{1}$$

Suppose that the state ξ_t can be divided into two parts, i.e.,

$$\xi_t = \begin{bmatrix} x_t \\ z_t \end{bmatrix} \tag{2}$$

and accordingly that system (1) can be *decomposed* as follows

$$\begin{aligned} x_{t+1} &= f_t^x(x_t, z_t, v_t^x), \\ z_{t+1} &= f_t^z(x_t, z_t, v_t^z), \\ y_t &= h_t(x_t, z_t, e_t). \end{aligned} \tag{3}$$

We study the problem of recursively estimating the posterior probability density function $p(z_t, x_{0:t}|y_{0:t})$.

According to the following factorization

$$p(z_t, x_{0:t}|y_{0:t}) = p(z_t|x_{0:t}, y_{0:t})p(x_{0:t}|y_{0:t},) \tag{4}$$

where $x_{0:t} \triangleq \{x_0, ..., x_t\}$, the filtering problem (4) can be *split into two linked filtering sub-problems:*

1. estimating the probability density function $p(x_{0:t}|y_{0:t})$;
2. estimating the probability density function $p(z_t|x_{0:t}, y_{0:t})$.

Solution

The DPF handles both filtering sub-problems with PFs. Roughly speaking, the DPF solves the first filtering sub-problem by using a PF with N_x particles to estimate $p(x_{0:t}|y_{0:t})$. Then the second filtering sub-problem is solved by using N_x PFs with N_z particles each to estimate $p(z_t|x_{0:t}^{(i)}, y_{0:t})$, $i = 1, ..., N_x$.

A sketch of the solution is provided in the figure below.

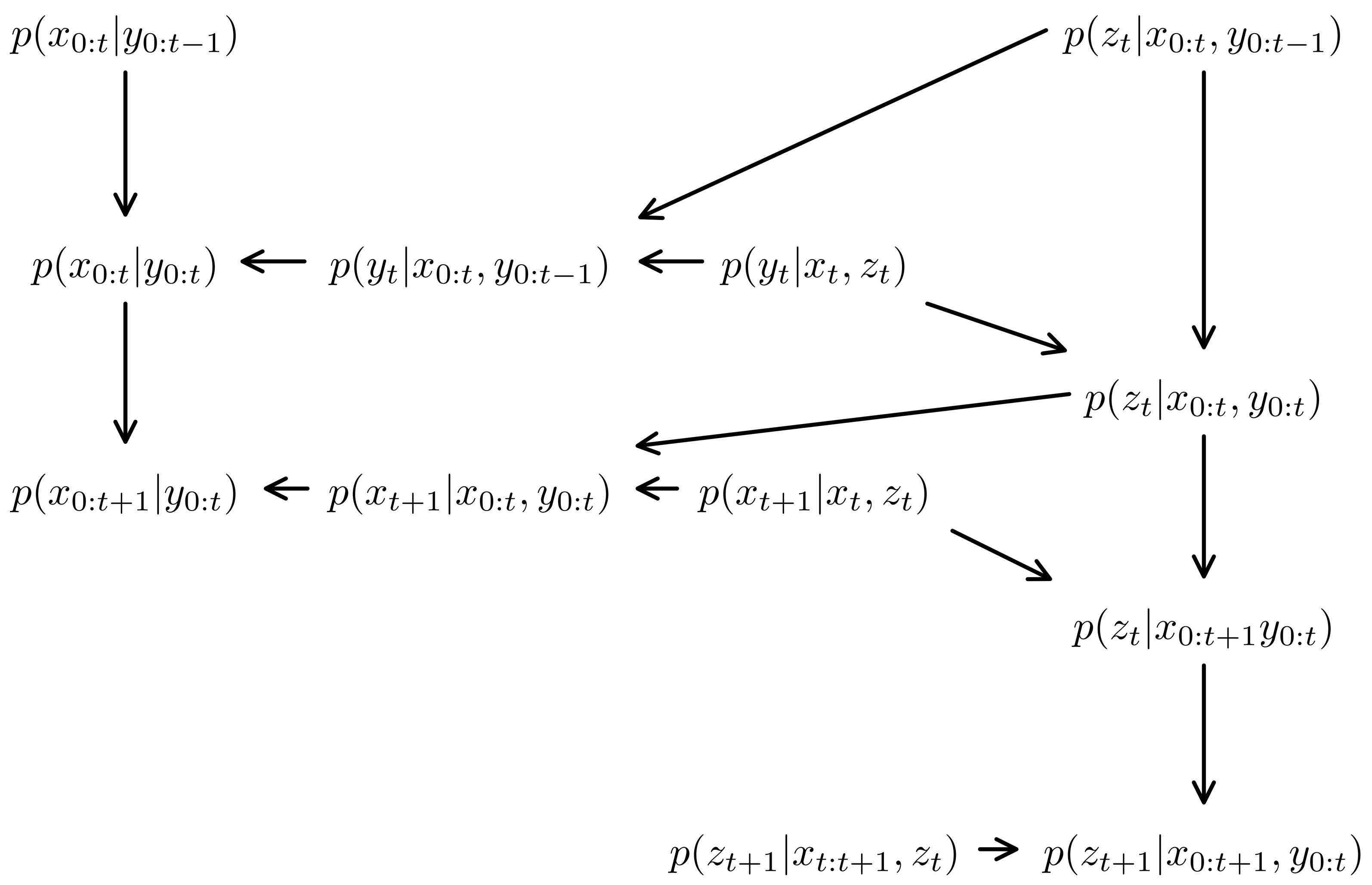

Figure 1: Sketch of the steps involved in deriving the DPF.

The N_x PFs used to handle the second filtering sub-problem can be implemented in parallel. By exploiting this *parallelism*, the DPF can shorten the execution time.

An Illustrative Example

Consider the following two dimensional nonlinear system

$$\begin{aligned} x_{t+1} &= x_t + \frac{z_t}{1+z_t^2} + v_t^x \\ z_{t+1} &= x_t + 0.5z_t + \frac{25z_t}{1+z_t^2} + 8\cos(1.2(t-1)) + v_t^z \\ y_t &= \text{atan}(x_t) + \frac{z_t^2}{20} + e_t \end{aligned} \tag{5}$$

where $[x_0 \;\; z_0]^T$ is assumed Gaussian distributed with $[x_0 \;\; z_0]^T \sim \mathcal{N}(0, I_{2\times2})$, $v_t = [v_t^x \;\; v_t^z]^T$ and e_t are assumed white and Gaussian distributed with

$$v_t \sim \mathcal{N}\left(0, \begin{bmatrix} 1 & 0.1 \\ 0.1 & 10 \end{bmatrix}\right) \text{ and } e_t \sim \mathcal{N}(0,1). \tag{6}$$

The result from a MCMC study comparing the bootstrap PF and the DPF is given in the table below. T_{pi} is the potential execution time of the parallel implementation of the bootstrap PF and the DPF.

Table 1: Simulation Result based on 20000 Monte Carlo runs

Case	RMSE for x	RMSE for z	T_{pi}
Parametric CRLB	1.89	1.54	–
Bootstrap PF, $M = 1000$	2.04	2.40	0.065
Bootstrap PF, $M = 2000$	1.98	2.32	0.105
DPF, $N_x = 100$, $N_z = 19$	2.02	2.40	0.028
DPF, $N_x = 120$, $N_z = 19$	2.01	2.42	0.029
DPF, $N_x = 110$, $N_z = 24$	2.01	2.38	0.030
DPF, $N_x = 120$, $N_z = 24$	2.00	2.36	0.031

It can be seen from the RMSE column in Table 1 that with suitably chosen N_x and N_z the DPF achieves the same level of performance as the bootstrap PF. Parallel implementation of the DPF has however the potential to shorten the execution time significantly.

Conclusions

In this work we have shown that it is possible to *exploit nontrivial structures* within a particle filtering framework. This was accomplished by acknowledging the particular structure by *decomposing the filtering problem into two linked filtering sub-problems*. These problems are then solved by two linked particle filters. Furthermore, this opens up for interesting *possibilities for parallelization*, which in turn can reduce the execution time.

Acknowledgment

Partially supported by the Swedish foundation for strategic research in the center MOVIII and by the Swedish Research Council in the Linnaeus center CADICS.