Using particle filters to identify nonlinear systems

“The particle filter provides a systematic way of exploring the state space”

Thomas Schön
Division of Systems and Control
Department of Information Technology
Uppsala University.

Email: thomas.schon@it.uu.se,
www: user.it.uu.se/~thosc112

The talk is based on this paper:

Introduction

A state space model (SSM) consists of a Markov process \( \{x_t\}_{t \geq 1} \) that is indirectly observed via a measurement process \( \{y_t\}_{t \geq 1} \),

\[
\begin{align*}
x_{t+1} | x_t & \sim f_\theta(x_{t+1} | x_t, u_t), & x_{t+1} = a_\theta(x_t, u_t) + v_{\theta,t}, \\
y_t | x_t & \sim g_\theta(y_t | x_t, u_t), & y_t = c_\theta(x_t, u_t) + e_{\theta,t}, \\
x_1 & \sim \mu_\theta(x_1), & x_1 \sim \mu_\theta(x_1), \\
(\theta \sim \pi(\theta)) & & (\theta \sim \pi(\theta)).
\end{align*}
\]

Identifying the nonlinear SSM: Find \( \theta \) based on \( y_{1:T} \triangleq \{y_1, y_2, \ldots, y_T\} \) (and \( u_{1:T} \)). Hence, the off-line problem.

One of the key challenges: The states \( x_{1:T} \) are unknown.

Aim of the talk: Reveal the structure of the system identification problem arising in nonlinear SSMs and highlight where SMC is used.
Two commonly used problem formulations

**Maximum likelihood (ML) formulation** – model the unknown parameters as a deterministic variable and solve

\[
\hat{\theta}_{\text{ML}} = \arg \max_{\theta \in \Theta} p_{\theta}(y_{1:T}).
\]

**Bayesian formulation** – model the unknown parameters as a random variable \( \theta \sim \pi(\theta) \) and compute

\[
p(\theta | y_{1:T}) = \frac{p(y_{1:T} | \theta)\pi(\theta)}{p(y_{1:T})} = \frac{p_{\theta}(y_{1:T})\pi(\theta)}{p(y_{1:T})}.
\]

The **combination** of ML and Bayes is probably more interesting than we think.
Central object – the likelihood

The likelihood is computed by marginalizing the joint density

\[ p_\theta(x_1:T, y_1:T) = \mu_\theta(x_1) \prod_{t=1}^{T} g_\theta(y_t \mid x_t) \prod_{t=1}^{T-1} f_\theta(x_{t+1} \mid x_t), \]

w.r.t. the state sequence \( x_1:T, \)

\[ p_\theta(y_1:T) = \int p_\theta(x_1:T, y_1:T) dx_1:T. \]

We are averaging \( p_\theta(x_1:T, y_1:T) \) over all possible state sequences.

Equivalently we have

\[ p_\theta(y_1:T) = \prod_{t=1}^{T} p_\theta(y_t \mid y_{1:t-1}) = \prod_{t=1}^{T} \int g_\theta(y_t \mid x_t) p_\theta(x_t \mid y_{1:t-1}) dx_t. \]

key challenge
Sequential Monte Carlo

The need for computational methods, such as SMC, is tightly coupled to the intractability of the integrals on the previous slide.

SMC offers numerical approximations to state estimation problems.

The particle filter and the particle smoother maintain empirical approximations

\[
\hat{p}_\theta(x_t \mid y_{1:t}) = \sum_{i=1}^{N} w_t^i \delta_{x_t^i}(x_t), \quad \hat{p}_\theta(x_{1:t} \mid y_{1:t}) = \sum_{i=1}^{N} w_t^i \delta_{x_{1:t}^i}(x_{1:t}).
\]

Converge to the true distributions as \( N \rightarrow \infty \).
Using SMC for nonlinear system identification

SMC can be used to approximately

1. Compute the likelihood and its derivatives.
2. Solve state smoothing problems, e.g. compute \( p(x_{1:T} \mid y_{1:T}) \).
3. Simulate from the smoothing pdf, \( \tilde{x}_{1:T} \sim p(x_{1:T} \mid y_{1:T}) \).

These three capabilities are key components in implementing various nonlinear system identification strategies.
Identification strategies – overview

**Marginalisation** Deal with $x_{1:T}$ by marginalizing (integrating) them out and view $\theta$ as the only unknown quantity.

- Frequentistic formulation: Prediction Error Method (PEM) and direct maximization of the likelihood.
- Bayesian formulation: the Metropolis Hastings sampler.

**Data augmentation** Deal with $x_{1:T}$ by treating them as auxiliary variables to be estimated along with $\theta$.

- Frequentistic formulation: Expectation Maximization (EM) algorithm.
- Bayesian formulation: the Gibbs sampler.

Only data augmentation strategies in this talk.
1. Problem formulation
2. Identification strategies for nonlinear SSMs
3. **Sequential Monte Carlo (SMC)**
4. Using SMC as a proposal mechanism within MCMC
5. Data augmentation
   a) Expectation maximization (EM)
   b) Gibbs sampling
6. Snapshots of current research
   a) The Gaussian process SSM and regularization
   b) The nonlinear SSM is just a special case...
Sequential Monte Carlo – particle filter

The particle filter provides an approximation $p(x_{1:t} \mid y_{1:t})$, when the state evolves according to an SSM,

$$
x_{t+1} \mid x_t \sim f_\theta(x_{t+1} \mid x_t),
$$

$$
y_t \mid x_t \sim g_\theta(y_t \mid x_t),
$$

$$
x_1 \sim \mu_\theta(x_1).
$$

The particle filter maintains an empirical distribution made up of $N$ samples (particles) $\{x_{1:t}^i\}_{i=1}^N$ and corresponding weights $\{w_{1:t}^i\}_{i=1}^N$

$$
\hat{p}(x_{1:t} \mid y_{1:t}) = \sum_{i=1}^N w_{t}^i \delta_{x_{1:t}^i}(x_{1:t}).
$$

"Think of each particle as one simulation of the system state. Keep the ones that best explains the measurements."
The particle filter – toy problem

Consider a toy 1D localization problem.

Dynamic model:

\[ x_{t+1} = x_t + u_t + v_t, \]

where \( x_t \) denotes position, \( u_t \) denotes velocity (known), \( v_t \sim \mathcal{N}(0, 5) \) denotes an unknown disturbance.

Measurements:

\[ y_t = h(x_t) + e_t. \]

where \( h(\cdot) \) denotes the world model (here the terrain height) and \( e_t \sim \mathcal{N}(0, 1) \) denotes an unknown disturbance.

The same idea has been used for the Swedish fighter JAS 39 Gripen,


thomas.schon@it.uu.se
The particle filter – toy problem

Highlights two key capabilities of the PF:

1. Automatically handles an unknown and dynamically changing number of hypotheses.

2. Work with nonlinear/non-Gaussian models.
Example – indoor localization

Aim: Compute the position of a person moving around indoors using sensors (inertial, magnetometer and radio) located in an ID badge and a map.

Show movie
Sequential Monte Carlo – particle filter

SMC = resampling + sequential importance sampling

1. Resampling: \( P(a^i_t = j) = \bar{w}^j_{t-1} / \sum_l \bar{w}^l_{t-1} \).

2. Propagation: \( x^i_t \sim f_{\theta}(x_t | x^{a^i_t}_{1:t-1}) \) and \( x^i_{1:t} = \{ x^{a^i_t}_{1:t-1}, x^i_t \} \).

3. Weighting: \( \bar{w}^i_t = W_t(x^i_t) = g_{\theta}(y_t | x_t) \).

The ancestor indices \( \{ a^i_t \}_{i=1}^N \) are very useful auxiliary variables! They make the stochasticity of the resampling step explicit.
Sequential Monte Carlo – particle filter

Let

\[ x_t \triangleq \{ x^1_t, \ldots, x^N_t \}, \quad a_t \triangleq \{ a^1_t, \ldots, a^N_t \} \]

denote all particles and ancestor indices generated at time \( t \).

The SMC algorithm generates a single realization of a collection of random variables

\[ \{ x_{1:T}, a_{2:T} \} \in X^{NT} \times \{ 1, \ldots, N \}^{N(T-1)} \]
distributed according to

\[ \psi(x_{1:T}, a_{2:T}) \triangleq \prod_{i=1}^{N} q_1(x^i_1) \prod_{t=2}^{T} \prod_{i=1}^{N} M_t(a^i_t, x^i_t), \]

where

\[ M_t(a_t, x_t) = \frac{\bar{w}^{a_t}_{t-1}}{\sum_l \bar{w}^l_{t-1}} f_t(x_t \mid x^{a_t}_{1:t-1}). \]
The particle system degenerates (illustration)

Clearly motivates the need for particle smoothers.

Self-contained introduction to particle smoothing using BS and AS

**Exact approximation and PMCMC**

**Exact approximation:** \( p(\theta \mid y_{1:T}) \) is recovered exactly, despite the fact that we employ an SMC approximation of the likelihood using a finite number of particles \( N \).

This is one of the members in the particle MCMC (PMCMC) family introduced by


The idea underlying PMCMC is to make use of SMC algorithms to propose (simulate) state trajectories \( x_{1:T} \). These state trajectories are then used within standard MCMC algorithms.

1. Particle Metropolis Hastings
2. Particle Gibbs
Outline

1. Problem formulation
2. Identification strategies for nonlinear SSMs
3. Sequential Monte Carlo (SMC)
4. Using SMC as a proposal mechanism within MCMC
5. Data augmentation
   a) Expectation maximization (EM)
   b) Gibbs sampling
6. Snapshots of current research
   a) The Gaussian process SSM and regularization
   b) The nonlinear SSM is just a special case...
Identification strategy – data augmentation

Motivation: If we had access to the complete likelihood

\[ p_{\theta}(x_{1:T}, y_{1:T}) = \mu_{\theta}(x_1) \prod_{t=1}^{T} g_{\theta}(y_t \mid x_t) \prod_{t=1}^{T-1} f_{\theta}(x_{t+1} \mid x_t) \]

the problem would be much easier.

Key idea: Treat the state sequence \( x_{1:T} \) as an auxiliary variable that is estimated together with \( \theta \).

The data augmentation strategy breaks the original problem into two new and closely linked problems.

Intuitively the data augmentation strategy amounts to iterating between updating \( x_{1:T} \) and \( \theta \).
Data augmentation – EM

Maximum likelihood (ML) formulation – model the unknown parameters as a deterministic variable and solve

$$\hat{\theta}_{\text{ML}} = \arg\max_{\theta \in \Theta} p_{\theta}(y_1:T).$$

The expectation maximization algorithm is an iterative approach to compute ML estimates of unknown parameters ($\theta$) in probabilistic models involving latent variables (the state trajectory $x_{1:T}$).

Expectation maximization (EM) employs the complete likelihood $p_{\theta}(x_{1:T}, y_{1:T})$ as a substitute for the observed likelihood $p_{\theta}(y_{1:T})$,

$$p_{\theta}(x_{1:T}, y_{1:T}) = p_{\theta}(x_{1:T} \mid y_{1:T}) p_{\theta}(y_{1:T}).$$
Data augmentation – EM

EM works by iteratively computing

\[ Q(\theta, \theta_k) = \int \log p_\theta(x_{1:T}, y_{1:T}) p_{\theta_k}(x_{1:T} | y_{1:T}) \, dx_{1:T} \]

and then maximizing \( Q(\theta, \theta_k) \) w.r.t. \( \theta \).

Problem: The E-step requires us to solve a smoothing problem, i.e. to compute an expectation under \( p_{\theta_k}(x_{1:T} | y_{1:T}) \).

SMC is used to approximate the smoothing pdf \( p_{\theta_k}(x_{1:T} | y_{1:T}) \).
Example – blind Wiener identification

\[ x_{t+1} = \begin{pmatrix} A & B \end{pmatrix} \begin{pmatrix} x_t \\ u_t \end{pmatrix}, \quad u_t \sim \mathcal{N}(0, Q), \]

\[ z_t = Cx_t, \quad y_t = h(z_t, \beta) + e_t, \quad e_t \sim \mathcal{N}(0, R). \]

**Id. problem:** Find \( \mathcal{L}, \beta, r_1, \) and \( r_2 \) based on \( \{y_{1,1:T}, y_{2,1:T}\} \).
Example – blind Wiener identification

- Second order LGSS model with complex poles.
- Results obtained using $T = 1000$ samples.
- Employ the EMPS with $N = 100$ particles.
- The plots are based on 100 realisations of data.
- Nonlinearities (dead zone and saturation) shown on next slide.

Bode plot of estimated mean (black), true system (red) and the result for all 100 realisations (gray).
Example – blind Wiener identification

Estimated mean (black), true static nonlinearity (red) and the result for all 100 realisations (gray).

**Bayesian** formulation – model the unknown parameters as a random variable $\theta \sim \pi(\theta)$ and compute

$$p(\theta \mid y_{1:T}) = \frac{p(y_{1:T} \mid \theta)\pi(\theta)}{p(y_{1:T})} = \frac{p_\theta(y_{1:T})\pi(\theta)}{p(y_{1:T})}.$$

Gibbs sampling amounts to sequentially sampling from conditionals of the target distribution $p(\theta, x_{1:T} \mid y_{1:T})$.

A (blocked) example:

- Draw $\theta[m] \sim p(\theta \mid x_{1:T}[m-1], y_{1:T})$; \textbf{OK!}
- Draw $x_{1:T}[m] \sim p(x_{1:T} \mid \theta[m], y_{1:T})$. \textbf{Hard!}

**SMC** is used to simulate from the smoothing pdf $p(x_{1:T} \mid y_{1:T})$. 
Example – semiparametric Wiener model

Parametric LGSS and a nonparametric static nonlinearity:

\[ x_{t+1} = \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix} + v_t, \quad v_t \sim \mathcal{N}(0, Q), \]

\[ z_t = Cx_t, \]

\[ y_t = g(z_t) + e_t, \quad e_t \sim \mathcal{N}(0, R). \]
"Parameters": \( \theta = \{ A, B, Q, g(\cdot), r \} \).

**Bayesian model** specified by priors
- Conjugate priors for \( \Gamma = [A \ B] \), \( Q \) and \( r \),
  - \( p(\Gamma, Q) = \text{Matrix-normal inverse-Wishart} \)
  - \( p(r) = \text{inverse-Wishart} \)
- Gaussian process prior on \( g(\cdot) \),
  \[ g(\cdot) \sim \mathcal{GP}(z, k(z, z')) . \]

**Inference** using PGAS with \( N = 15 \) particles. \( T = 1000 \) measurements. We ran 15 000 MCMC iterations and discarded 5 000 as burn-in.
Example – semiparametric Wiener model

Show movie

Bode diagram of the 4th-order linear system. Estimated mean (dashed black), true (solid black) and 99% credibility intervals (blue).

Static nonlinearity (non-monotonic), estimated mean (dashed black), true (black) and the 99% credibility intervals (blue).

In stochastic approximation EM (SAEM) $Q(\theta, \theta_k)$ is replaced by

$$\hat{Q}_k(\theta) = (1 - \gamma_k)\hat{Q}_{k-1}(\theta) + \gamma_k \log p_\theta(x_{1:T}[k], y_{1:T})$$

where $x_{1:T}[k]$ denotes a sample from $p_{\theta_k}(x_{1:T} \mid y_{1:T})$.

Outline

1. Problem formulation
2. Identification strategies for nonlinear SSMs
3. Sequential Monte Carlo (SMC)
4. Using SMC as a proposal mechanism within MCMC
5. Data augmentation
   a) Expectation maximization (EM)
   b) Gibbs sampling
6. Snapshots of current research
   a) The Gaussian process SSM and regularization
   b) The nonlinear SSM is just a special case...
Consider the Gaussian Process SSM (GP-SSM):

\[
\begin{align*}
    x_{t+1} &= f(x_t) + w_t, \quad \text{s.t.} \quad f(x) \sim \mathcal{GP}(0, \kappa_{\theta, f}(x, x')), \\
    y_t &= g(x_t) + e_t, \quad \text{s.t.} \quad g(x) \sim \mathcal{GP}(0, \kappa_{\theta, g}(x, x')), 
\end{align*}
\]

The model functions \( f \) and \( g \) are assumed to be realizations from a Gaussian process prior and \( w_t \sim \mathcal{N}(0, Q), \ e_t \sim \mathcal{N}(0, R) \).

We can now find the posterior distribution

\[
p(f, g, Q, R, \theta | y_{1:T}),
\]

by making use of PGAS.

Regularizing nonlinear SSMs

Regularization allows us to tune the model complexity.

Place a GP-SSM prior on the nonlinear transition \( f(\cdot) \),

\[
f(x) = \sum_{k=1}^{\infty} \omega^k \phi^k(x) \approx \sum_{k=1}^{m} \omega^k \phi^k(x)
\]

Regularize using a prior on the weights, e.g.,

\[
p([\omega^1, \ldots, \omega^m]) = \mathcal{N}(0, P^{-1}).
\]

PSAEM

The nonlinear SSM is just a special case...

A **graphical model** is a probabilistic model where a graph \( \mathcal{G} = (\mathcal{V}, \mathcal{E}) \) represents the conditional independency structure between random variables,

1. a set of **vertices** \( \mathcal{V} \) (nodes) represents the random variables
2. a set of **edges** \( \mathcal{E} \) containing elements \( (i, j) \in \mathcal{E} \) connecting a pair of nodes \( (i, j) \in \mathcal{V} \)

\[
p(x_{0:T}, y_{1:T}) = p(x_0) \prod_{t=1}^{N} p(x_t | x_{t-1}) \prod_{t=1}^{N} p(y_t | x_t).
\]
The nonlinear SSM is just a special case...

Constructing an artificial sequence of intermediate target distributions for an SMC sampler is a powerful (and quite possibly underutilized) idea.


## Conclusion

<table>
<thead>
<tr>
<th>ML</th>
<th>Bayesian</th>
<th>Marginalization</th>
<th>Data augmentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct optimization</td>
<td>Metropolis Hastings</td>
<td>Expectation Maximization</td>
<td>Gibbs sampling</td>
</tr>
</tbody>
</table>

SMC is used to realize all of these approaches for nonlinear SSMs.

SMC can be used to approximately

1. Compute the likelihood and its derivatives.
2. Solve state smoothing problems, e.g. compute \( p(x_{1:T} \mid y_{1:T}) \).
3. Simulate from the smoothing pdf, \( \tilde{x}_{1:T} \sim p(x_{1:T} \mid y_{1:T}) \).

---

1 day tutorial at ICASSP (Shanghai) on March 20, 2016,


A lot of interesting research that remains to be done!!
Open tenure track position in ML

If you are interested, send an e-mail to Thomas Schön
thomas.schon@it.uu.se

If you know someone else who might be interested feel free to help us spread this information as much as you can and want.

Deadline for applications: November 30, 2015. Details here:
www.uu.se/en/about-uu/join-us/details/?positionId=75817

2. I am also looking for **new PhD students**. Feel free to spread this information as well!! Topic: Nonlinear inference in system identification and machine learning.

Deadline for applications: December 20, 2015.
www.uu.se/en/about-uu/join-us/details/?positionId=78032