Sequential Monte Carlo methods and their use in graphical models

UPPSALA UNIVERSITET

Thomas Schön

Division of Systems and Control
Department of Information Technology
Uppsala University

Joint work with **Christian A. Naesseth** (Linköping University) and **Fredrik Lindsten** (University of Cambridge),

Dynamical systems are everywhere!

Some of the dynamical systems we have been working with,

We first have to learn the models. Then we can use them.

Outline

1. Probabilistic models of dynamical systems
2. State inference
3. Sequential Monte Carlo (SMC), the particle filter
   a) Key idea
   b) indoor localization example
4. Particle MCMC (very brief)
5. Inference in probabilistic graphical models

Basic representation: Two discrete-time stochastic processes,

- $\{x_t\}_{t\geq 1}$ representing the state of the system.
- $\{y_t\}_{t\geq 1}$ representing the measurements from the sensors.

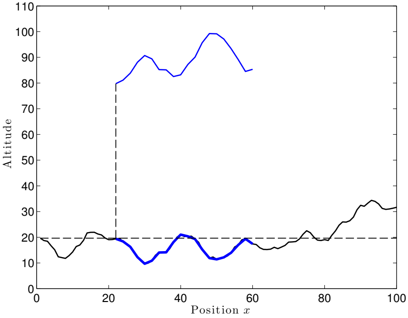
The probabilistic model is described using two (f and g) probability density functions (PDFs):

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

Model = PDF

This type of model is referred to as a **state space model (SSM)** or a **hidden Markov model (HMM)**.

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.

Aim: Compute a probabilistic representation of our knowledge of the state, based on information that is present in the measurements.

The **filtering PDF**

$$p(x_t \mid y_{1:t}),$$

provides a representation of the uncertainty about the state at time t, given all the measurements up to time t. **Measurement update**

$$p(x_t \mid y_{1:t}) = \frac{\overbrace{g(y_t \mid x_t)}^{\text{measurement model}} \overbrace{p(x_t \mid y_{1:t-1})}^{\text{prediction PDF}}}{p(y_t \mid y_{1:t-1})}.$$

Time update

$$p(x_t \mid y_{1:t-1}) = \int \underbrace{f(x_t \mid x_{t-1})}_{\text{dynamical model}} \underbrace{p(x_{t-1} \mid y_{1:t-1})}_{\text{filtering PDF}} \mathrm{d}x_{t-1}.$$

State inference – interesting case

Obvious question: what do we do in an interesting case, for example when we have a nonlinear model with non-Gaussian noise?

1. Need a general representation of the filtering PDF
2. Try to solve the equations

$$p(x_t \mid y_{1:t}) = \frac{g(y_t \mid x_t) p(x_t \mid y_{1:t-1})}{p(y_t \mid y_{1:t-1})},$$
$$p(x_t \mid y_{1:t-1}) = \int f(x_t \mid x_{t-1}) p(x_{t-1} \mid y_{1:t-1}) \mathrm{d}x_{t-1},$$

as accurately as possible.

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

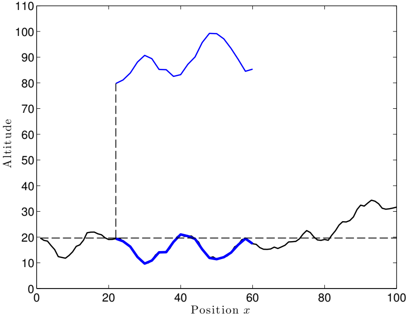
$$\begin{aligned} x_{t+1} \mid x_t &\sim f_t(x_{t+1} \mid x_t), \\ y_t \mid x_t &\sim g_t(y_t \mid x_t), \\ x_1 &\sim \mu(x_1). \end{aligned}$$

The particle filter maintains an empirical distribution made up of N samples (particles) $\{x_t^i\}_{i=1}^N$ and the corresponding weights $\{w_t^i\}_{i=1}^N$

$$\widehat{p}^N(x_t \mid y_{1:t}) = \sum_{i=1}^N w_t^i \delta_{x_t^i}(x_t).$$

"Think of each particle as one simulation of the system state. Only keep the good ones."

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:

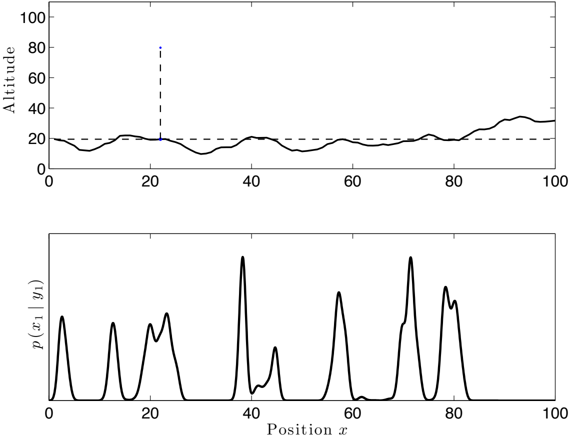
$$y_t = h(x_t) + e_t.$$

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. Details are available in,

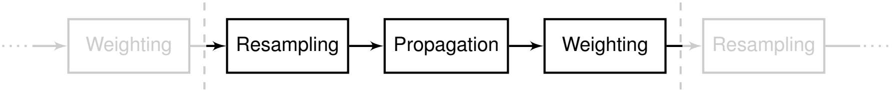
Thomas Schön, Fredrik Gustafsson, and Per-Johan Nordlund. **Marginalized particle filters for mixed linear/nonlinear state-space models**. *IEEE Transactions on Signal Processing*, 53(7):2279-2289, July 2005.

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.



1. **Resampling:** $\{x_{t-1}^i, w_{t-1}^i\}_{i=1}^N \to \{\tilde{x}_{t-1}^i, 1/N\}_{i=1}^N$.
2. **Propagation:** $x_t^i \sim q_t(x_t \mid \tilde{x}_{t-1}^i)$.
3. **Weighting:** $w_t^i = W_t(x_t^i, y_t)$.

The result is a new weighted set of particles $\{x_t^i, w_t^i\}_{i=1}^N$ targeting $p(x_t \mid y_{1:t})$.

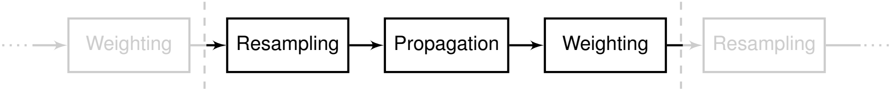
A systematic way of obtaining approximations that converge

Xiao-Li Hu, Thomas B. Schön and Lennart Ljung. **A basic convergence result for particle filtering**. *IEEE Transactions on Signal Processing*, 56(4):1337-1348, April 2008.

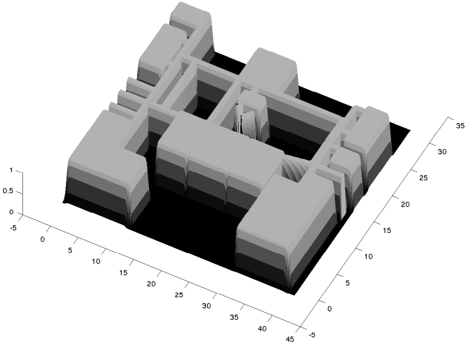
The particle filter has been around for roughly **20** years.

The use of particle methods for nonlinear system identification started to take off some **5** years ago.

Now this is a very active problem (and solution) within many fields.



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!

The idea underlying **PMCMC** is to make use of a certain SMC sampler to construct a Markov kernel leaving the joint smoothing distribution $p(x_{1:T} \mid \theta, y_{1:T})$ invariant.

This Markov kernel is then used in a **standard MCMC algorithm** (e.g. Gibbs, results in the **Particle Gibbs (PG)**).

Original paper

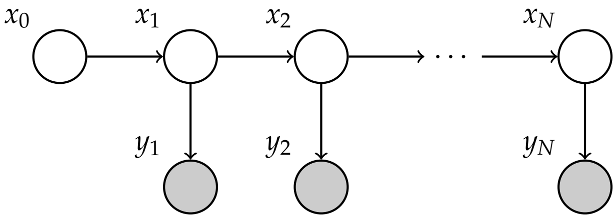
Christophe Andrieu, Arnaud Doucet and Roman Holenstein, **Particle Markov chain Monte Carlo methods**, *Journal of the Royal Statistical Society: Series B*, 72:269-342, 2010.

For a self-contained introduction (focused on BS and AS),

Fredrik Lindsten and Thomas B. Schön, **Backward simulation methods for Monte Carlo statistical inference**, *Foundations and Trends in Machine Learning*, 6(1):1-143, 2013.

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 $(i, j) \in \mathcal{V}$

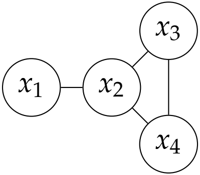


$$p(x_{0:T}, y_{1:T}) = \mu(x_0) \prod_{t=1}^{N} f(x_t \mid x_{t-1}) \prod_{t=1}^{N} g(y_t \mid x_t).$$

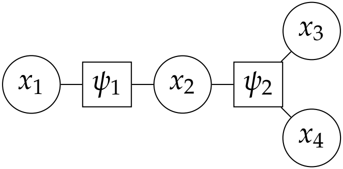
For an undirected graphical model (Markov random field), the joint PDF over all the involved random variables is

$$p(X_{\mathcal{V}}) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(X_C),$$

where $\mathcal{C}$ is the set of cliques in $\mathcal{G}$.



Undirected graph



Factor graph making interactions explicit.

SMC samplers are used to approximate a sequence of probability distributions on a sequence of probability spaces.

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

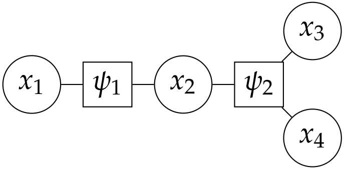
Key idea: Perform and make use of a **sequential decomposition** of the graphical model.

Using this SMC sampler within a particle MCMC sampler allows us to construct high-dimensional MCMC kernels for graphical models.

Sequential decomposition of GMs – pictures

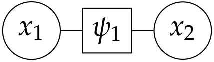
The joint PDF of the set of random variables indexed by $\mathcal{V}$, $X_{\mathcal{V}} \triangleq \{x_1, \dots, x_{|\mathcal{V}|}\}$

$$p(X_{\mathcal{V}}) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(X_C).$$

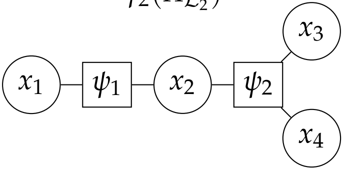


Sequential decomposition of the above factor graph (the target distributions are built up by adding factors at each iteration),

$\gamma_1(X_{\mathcal{L}_1})$



$\gamma_2(X_{\mathcal{L}_2})$



Example – Gaussian MRF

Consider a standard square lattice Gaussian MRF of size 10×10,

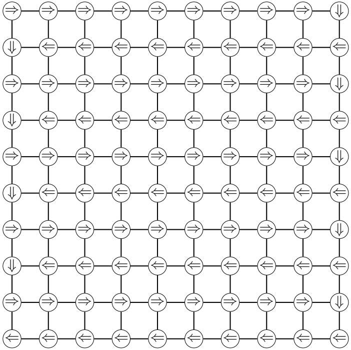
$$p(X_{\mathcal{V}}, Y_{\mathcal{V}}) \propto \prod_{i \in \mathcal{V}} e^{\frac{1}{2\sigma_i^2}(x_i - y_i)^2} \prod_{(i,j) \in \mathcal{E}} e^{\frac{1}{2\sigma_{ij}^2}(x_i - x_j)^2}$$

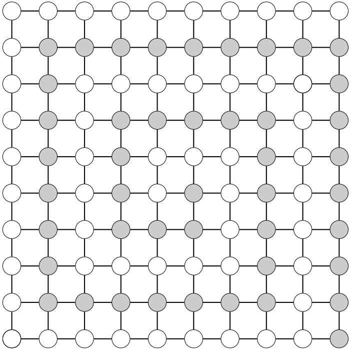
with latent variables $X_{\mathcal{V}} = \{x_1, \dots, x_{100}\}$ and measurements $Y_{\mathcal{V}} = \{y_1, \dots, y_{100}\}$ (simulated with $\sigma_i = 1$ and $\sigma_{ij} = 0.1$).

Goal: Compute the posterior distribution $p(X_{\mathcal{V}} \mid Y_{\mathcal{V}})$.

We run four MCMC samplers:

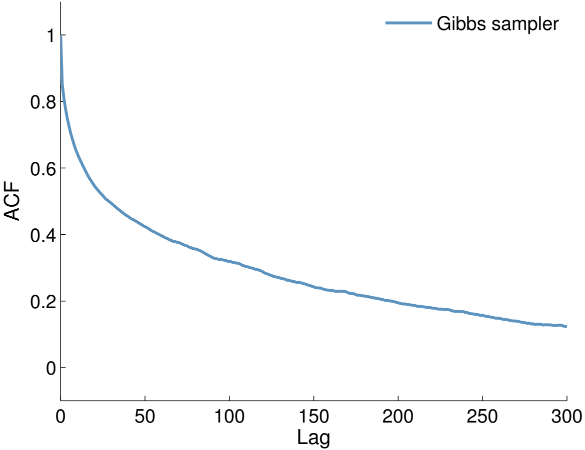
1. Standard one-at-a-time Gibbs
2. Tree sampler (Hamze & de Freitas, 2004)
3. PGAS – fully blocked ($N = 50$)
4. PGAS – partially blocked ($N = 50$)



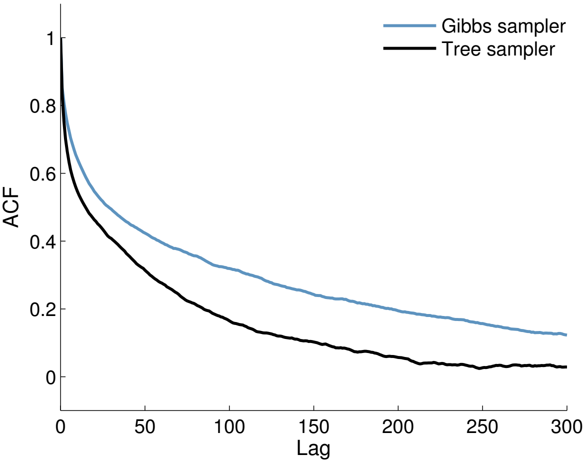


The arrows show the order in which the factors are added.

The two block structures used by the tree sampler and PGAS with partial blocking.

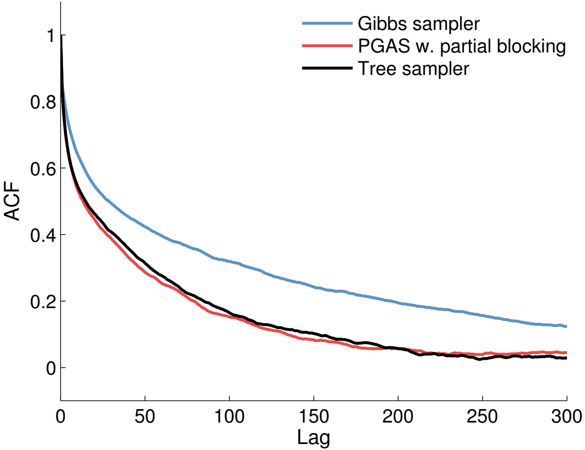


The one-step-at
-a-time Gibbs
sampler is strugg-
ling due to the
strong interactions.

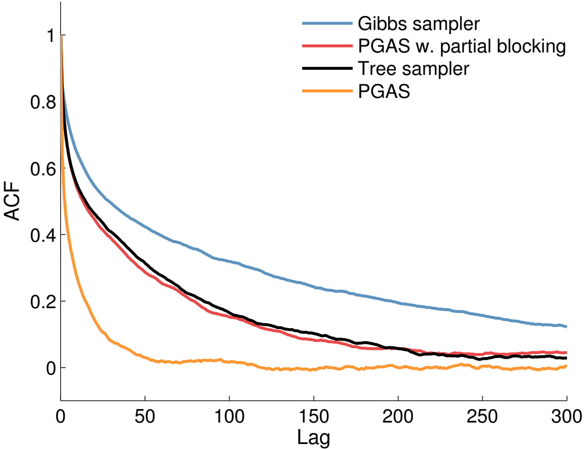


The tree sampler implements an “ideal” partially blocked Gibbs sampler.

Example – Gaussian MRF



PGAS with partial blocking is an **approximation of the tree sampler**. Already for relatively few particles we obtain a performance similar to the “ideal” tree sampler.



The fully blocked PGAS performs best, which is not surprising, since it samples all the (dependent) latent variables jointly.

The downside of PGAS is that it is computationally more expensive.

Conclusions

- Probabilistic models of dynamical systems.
- Sequential Monte Carlo introduced via the particle filter.
- Briefely mentioned PMCMC for Bayesian inference.
- **Key insight:** We exploit a sequential decomposition of the graphical model.

  "Standard SMC samplers using a non-standard construction of the intermediate target distributions"
- New mathematics looking for interesting problems (we have already found some, maybe you have some interesting ones as well?)

There is a lot of interesting research that remains to be done!!

SMC and PMCMC methods for graphical models

Christian A. Naesseth, Fredrik Lindsten and Thomas B. Schön, **Sequential Monte Carlo methods for graphical models**. *Preprint at arXiv:1402:0330*, June, 2014.

Christian A. Naesseth, Fredrik Lindsten and Thomas B. Schön, **Capacity estimation of two-dimensional channels using Sequential Monte Carlo**. *Soon on arXiv (IT)*, May, 2014.

Novel introduction of PMCMC (very nice paper!)

Christophe Andrieu, Arnaud Doucet and Roman Holenstein, **Particle Markov chain Monte Carlo methods**, *Journal of the Royal Statistical Society: Series B*, 72:269-342, 2010.

Self-contained introduction to BS and AS (not limited to SSMs)

Fredrik Lindsten and Thomas B. Schön, **Backward simulation methods for Monte Carlo statistical inference**, *Foundations and Trends in Machine Learning*, 6(1):1-143, 2013.

Particle Gibbs with ancestor sampling (PGAS)

Fredrik Lindsten, Michael I. Jordan and Thomas B. Schn. **Particle Gibbs with ancestor sampling**. *Journal of Machine Learning Research (JMLR)*, 2014. (accepted for publication)

Fredrik Lindsten, Michael I. Jordan and Thomas B. Schön, **Ancestor sampling for particle Gibbs**, *Advances in Neural Information Processing Systems (NIPS) 25*, Lake Tahoe, NV, US, December, 2012.

Thank you!!