Nonlinear system identification enabled via sequential Monte Carlo

UPPSALA
UNIVERSITET

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

Joint work with present and former students: Fredrik Lindsten (Cambridge), Johan Dahlin (Linköping), Johan Wågberg (Uppsala), Christian A. Naesseth (Linköping), Andreas Svensson (Uppsala) and Liang Dai (Uppsala).

UPPSALA UNIVERSITET

Nonlinear system identification

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

$$\begin{aligned} x_{t+1} \mid x_t &\sim f_{\theta,t}(x_{t+1} \mid x_t, u_t), \\ y_t \mid x_t &\sim g_{\theta,t}(y_t \mid x_t, u_t), \\ x_1 &\sim \mu_\theta(x_1), \\ (\theta &\sim \pi(\theta)). \end{aligned} \qquad \begin{aligned} x_{t+1} &= a_\theta(x_t, u_t) + v_{\theta,t}, \\ y_t &= c_\theta(x_t, u_t) + e_{\theta,t}, \\ x_1 &\sim \mu_\theta(x_1), \\ (\theta &\sim \pi(\theta)). \end{aligned}$$

We observe

$$y_{1:T} \triangleq \{y_1, \dots, y_T\}, \quad \text{and possibly } u_{1:T} \triangleq \{u_1, \dots, u_T\}.$$

(leaving the latent variables $x_{1:T}$ unobserved).

Identification problem: Find θ based on $y_{1:T}$ (and $u_{1:T}$).

UPPSALA UNIVERSITET

Two commonly used problem formulations

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

$$\widehat{\theta}_{\text{ML}} = \arg\max_{\theta \in \Theta} \, p_{\theta}(y_{1:T}).$$

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

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

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

Central object – the likelihood

The central object in both formulations is the likelihood

$$p_\theta(y_{1:T}) = \prod_{t=1}^{T} p_\theta(y_t \mid y_{1:t-1}).$$

The likelihood is computed by marginalizing the joint density $p_\theta(x_{1:T}, y_{1:T})$ w.r.t. the state sequence $x_{1:T}$

$$p_\theta(y_{1:T}) = \int p_\theta(x_{1:T}, y_{1:T}) \mathrm{d}x_{1:T} = \prod_{t=1}^{T} \int g_\theta(y_t \mid x_t) p_\theta(x_t \mid y_{1:t-1}) \mathrm{d}x_t.$$

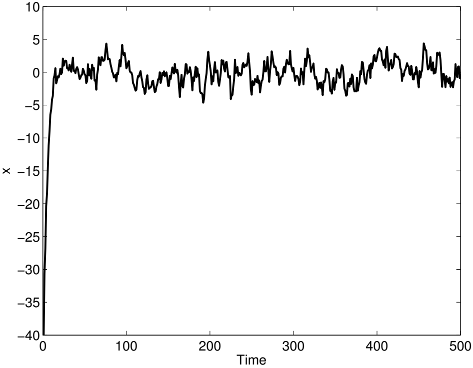
Key challenge: How to deal with the latent states.

Our solution: Sequential Monte Carlo (SMC) including particle filters/smoothers.

UPPSALA UNIVERSITET

Micro: MCMC – AR(1) example (I/II)

One realisation from $x[k+1] = 0.8x[k] + v[k]$ where $v[k] \sim \mathcal{N}(0,1)$. Initialise in $x[0] = -40$.

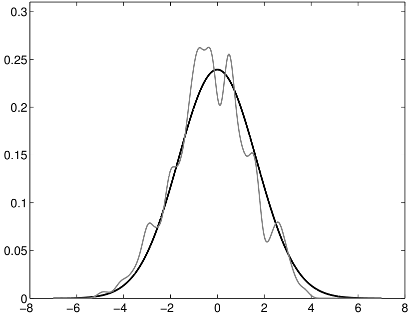


This will eventually generate samples from the following **stationary distribution**:

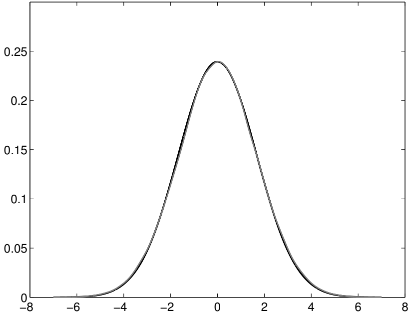
$$\pi^{\mathsf{s}}(x) = \mathcal{N}\left(x \,\middle|\, 0, \frac{1}{1-0.8^2}\right)$$

as $t \to \infty$.

Micro: MCMC – AR(1) example (II/II)



1 000 samples



100 000 samples

The true stationary distribution is showed in black and the empirical histogram obtained by simulating the Markov chain $x[k+1] = 0.8x[k] + v[k]$ is plotted in gray.

The initial $1\,000$ samples are discarded (burn-in).

Micro: MCMC

In the example, the Markov chain was fully specified and the stationary distribution could be expressed in closed form.

Not possible in the situations we are interested in, **but** we can (since 2010) find a Markov chain that has the target distribution (e.g. $p(\theta \mid y_{1:T})$) as its stationary distribution.

Two constructive ways of doing this are:

1. Metropolis Hastings (MH) algorithm
2. Gibbs sampling

Markov chain Monte Carlo (MCMC) methods allow us to generate samples from a **target distribution** by simulating a Markov chain which has the target distribution as its stationary distribution.

Outline

1. Problem formulation
2. Micro – MCMC
3. **Sketching identification strategies for nonlinear SSMs**
   a. Marginalization
   b. Data augmentation
4. Sequential Monte Carlo (SMC)
5. Using SMC as a proposal mechanism within MCMC
   a. Particle MCMC (PMCMC)
   b. A nontrivial example
6. The nonlinear SSM is just a special case...

UPPSALA UNIVERSITET

Identification strategies

The two identification strategies we are concerned with are:

- **Marginalization** Deal with the states by marginalizing them out.
- **Data augmentation** Deal with the states by treating them as auxiliary variables to be estimated along with the parameters.

	Marginalization	**Data augmentation**
ML	Direct optimization	Expectation Maximization
Bayesian	Metropolis Hastings	Gibbs sampling

Identification strategy – marginalization

Deal with the states by marginalizing them out.

1. Direct optimization work directly with the optimization problem

$$\widehat{\theta}_{\mathsf{ML}} = \arg\max_{\theta \in \Theta} \prod_{t=1}^{T} \int g_\theta(y_t \mid x_t) p_\theta(x_t \mid y_{1:t-1}) \mathrm{d}x_t.$$

Cannot be solved in closed form, use iterative numerical methods

$$\theta_{k+1} = \theta_k + \alpha_k s_k.$$

The search direction is typically computed according to

$$s_k = H_k g_k, \qquad g_k = \nabla_\theta p_\theta(y_{1:T})\big|_{\theta=\theta_k}.$$

SMC used to approximate the cost function and its derivative(s).

Identification strategy – marginalization

2. Metropolis Hastings (MH) is an MCMC method that produce a sequence of random variables $\{\theta[m]\}_{m \geq 1}$ by iterating

1. Propose a new sample θ'

$$\theta' \sim q(\cdot \mid \theta[m]).$$

2. Accept the new sample with probability

$$\alpha = \min\left(1, \frac{p_{\theta'}(y_{1:T})\pi(\theta')}{p_{\theta[m]}(y_{1:T})\pi(\theta[m])} \frac{q(\theta[m] \mid \theta')}{q(\theta' \mid \theta[m])}\right)$$

The above procedure results in a Markov chain $\{\theta[m]\}_{m \geq 1}$ with $p(\theta \mid y_{1:T})$ as its stationary distribution!

SMC used to approximate the likelihood $p_\theta(y_{1:T})$ in the acceptance probability.

Identification strategy – data augmentation

Deal with the states by treating them as auxiliary variables to be estimated along with the parameters.

Intuitively: Alternate between updating θ and $x_{1:T}$.

---

1. Expectation Maximization (EM)

(E) Compute a conditional expectation

$$\mathcal{Q}(\theta, \theta[k]) \triangleq \int \log p_\theta(x_{1:T}, y_{1:T}) \underbrace{p_{\theta[k]}(x_{1:T} \mid y_{1:T})} \mathrm{d}x_{1:T}.$$

(M) Maximize $\mathcal{Q}(\theta, \theta[k])$ w.r.t. θ

$$\theta[k+1] = \arg\max_\theta \; \mathcal{Q}(\theta, \theta[k]).$$

SMC is used to approximate the JSD $p_{\theta[k]}(x_{1:T} \mid y_{1:T})$.

Identification strategy – data augmentation

2. Gibbs sampling aim at compute $p(\theta, x_{1:T} \mid y_{1:T})$.

Gibbs sampling (blocked) for SSMs amounts to iterating

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

The above procedure results in a Markov chain,

$$\{\theta[m], x_{1:T}[m]\}_{m \geq 1}$$

with $p(\theta, x_{1:T} \mid y_T)$ as its stationary distribution!

SMC is used to generate a state sequence $x_{1:T}[m]$ from $p(x_{1:T} \mid \theta[m], y_{1:T})$.

UPPSALA UNIVERSITET

Outline

1. Problem formulation
2. Micro – MCMC
3. Sketching identification strategies for nonlinear SSMs
   a. Marginalization
   b. Data augmentation
4. **Sequential Monte Carlo (SMC)**
5. Using SMC as a proposal mechanism within MCMC
   a. Particle MCMC (PMCMC)
   b. A nontrivial example
6. 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,

$$\begin{aligned} x_{t+1} \mid x_t &\sim f_\theta(x_{t+1} \mid x_t), \\ y_t \mid x_t &\sim g_\theta(y_t \mid x_t), \\ x_1 &\sim \mu_\theta(x_1). \end{aligned}$$

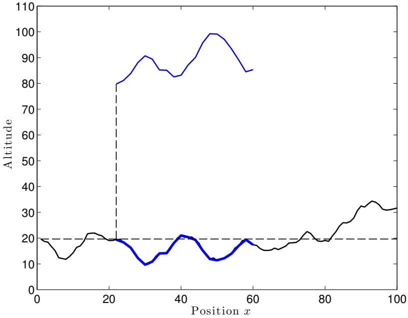
The particle filter maintains an empirical distribution made up of N samples (particles) $\{x_{1:t}^i\}_{i=1}^N$ and corresponding weights $\{w_{1:t}^i\}_{i=1}^N$

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

"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:

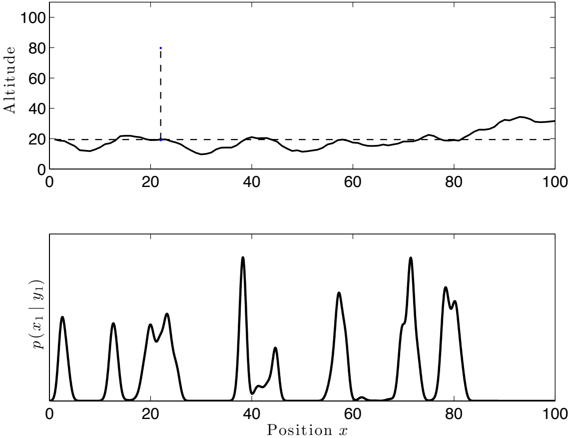
$$y_t = h(x_t) + e_t.$$

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. 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



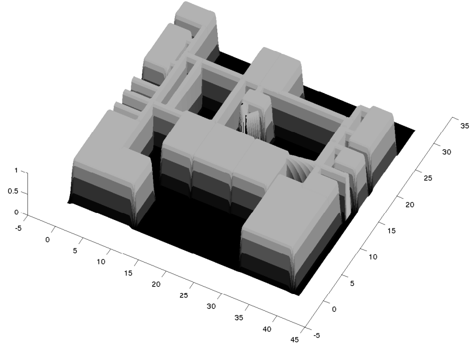
Highlights two **key capabilities** of the PF:

1. Automatically handles an unknown and dynamically changing number of hypotheses.
2. Work with nonlinear/non-Gaussian models.

UPPSALA UNIVERSITET

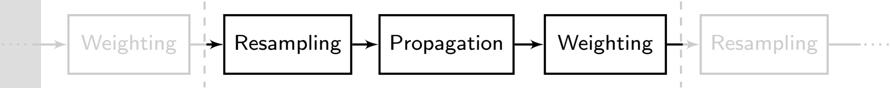
Example – indoor localization

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:** $\mathbb{P}\left(a_t^i = j\right) = \bar{w}_{t-1}^j / \sum_l \bar{w}_{t-1}^l$.
2. **Propagation:** $x_t^i \sim f_\theta(x_t \,|\, x_{1:t-1}^{a_t^i})$ and $x_{1:t}^i = \{x_{1:t-1}^{a_t^i}, x_t^i\}$.
3. **Weighting:** $\bar{w}_t^i = W_t(x_t^i) = g_\theta(y_t \,|\, x_t)$.

The **ancestor indices** $\{a_t^i\}_{i=1}^N$ are very **useful** auxiliary variables! They make the stochasticity of the resampling step explicit.

Sequential Monte Carlo – particle filter

Let

$$\boldsymbol{x}_t \triangleq \{x_t^1, \dots, x_t^N\}, \qquad \boldsymbol{a}_t \triangleq \{a_t^1, \dots, a_t^N\}$$

denote all particles and ancestor indices generated at time t.

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

$$\{\boldsymbol{x}_{1:T}, \boldsymbol{a}_{2:T}\} \in \mathsf{X}^{NT} \times \{1, \dots, N\}^{N(T-1)}$$

distributed according to

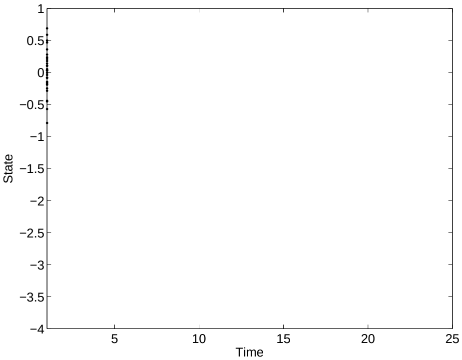
$$\psi(\boldsymbol{x}_{1:T}, \boldsymbol{a}_{2:T}) \triangleq \prod_{i=1}^{N} q_1(x_1^i) \prod_{t=2}^{T} \prod_{i=1}^{N} M_t(a_t^i, x_t^i),$$

where

$$M_t(a_t, x_t) = \frac{\bar{w}_{t-1}^{a_t}}{\sum_l \bar{w}_{t-1}^l} f_t(x_t \mid x_{1:t-1}^{a_t}).$$

UPPSALA UNIVERSITET

The particle system degenerates (illustration)



Clearly motivates the need for **particle smoothers**.

Self-contained introduction to particle smoothing using 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.

Using SMC within MCMC (PMCMC)

Particle MCMC (PMCMC) is a systematic way of combining SMC and MCMC.

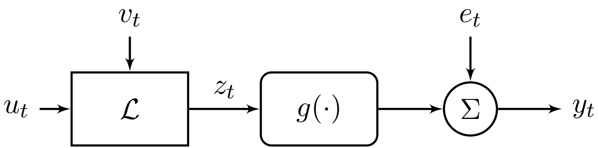
Intuitively: SMC is used as a high-dimensional proposal mechanism on the space of state trajectories X^T.

A bit more precise: Construct a Markov chain with $p(\theta \mid y_{1:T})$ (or $p(\theta, x_{1:T} \mid y_{1:T})$) as its stationary distribution.

Pioneered by the work

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.

Example – semiparametric Wiener model



Parametric LGSS and a nonparametric static nonlinearity:

$$
\begin{aligned}
x_{t+1} &= \underbrace{\begin{pmatrix} A & B \end{pmatrix}}_{\Gamma} \begin{pmatrix} x_t \\ u_t \end{pmatrix} + v_t, & v_t &\sim \mathcal{N}(0, Q), \\
z_t &= C x_t. & & \\
y_t &= g(z_t) + e_t, & e_t &\sim \mathcal{N}(0, R).
\end{aligned}
$$

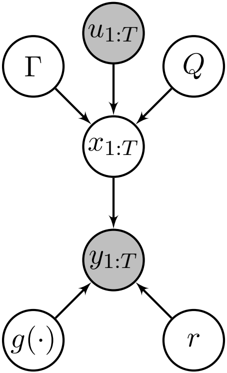
Example – semiparametric Wiener model

"Parameters": $\theta = \{A, B, Q, g(\,\cdot\,), r\}$.

Bayesian model specified by priors

- Conjugate priors for $\Gamma = [A \; B]$, Q and r,
  - $p(\Gamma, Q) =$ Matrix-normal inverse-Wishart
  - $p(r) =$ inverse-Wishart
- Gaussian process prior on $g(\,\cdot\,)$,

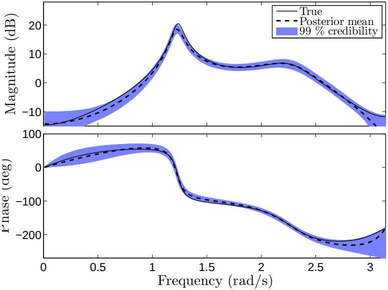
$$g(\,\cdot\,) \sim \mathcal{GP}(z, k(z, z')).$$



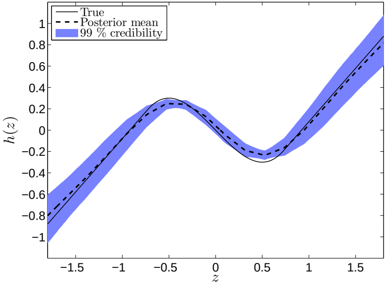
Inference using PGAS with $N = 15$ particles. $T = 1\,000$ measurements. We ran $15\,000$ MCMC iterations and discarded $5\,000$ as burn-in.

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).

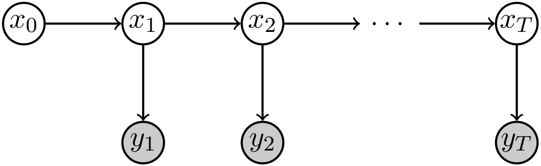
Fredrik Lindsten, Thomas B. Schön and Michael I. Jordan. **Bayesian semiparametric Wiener system identification**. *Automatica*, 49(7): 2053-2063, July 2013.

UPPSALA UNIVERSITET

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,

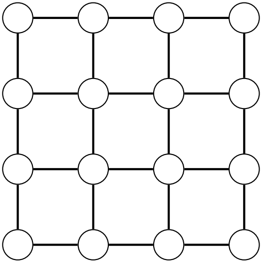
1. a set of **vertices** $\mathcal{V}$ (nodes) represents the random variables
2. a set of **edges** $\mathcal{E}$ containing elements $(i, j) \in \mathcal{E}$ connecting a pair of nodes $(i, j) \in \mathcal{V}$

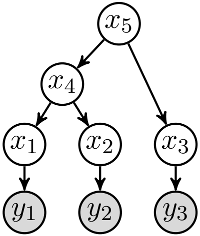


$$p(x_{0:T}, y_{1:T}) = p(x_0) \prod_{t=1}^{N} p(x_t \mid x_{t-1}) \prod_{t=1}^{N} p(y_t \mid x_t).$$

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.





Christian A. Naesseth, Fredrik Lindsten and Thomas B. Schön, **Sequential Monte Carlo methods for graphical models**. *Advances in Neural Information Processing Systems (NIPS)* 27, Montreal, Canada, December, 2014.

Fredrik Lindsten, Adam M. Johansen, Christian A. Naesseth, Bonnie Kirkpatrick, Thomas B. Schön, John Aston and Alexandre Bouchard-Côté. **Divide-and-Conquer with Sequential Monte Carlo**. *arXiv:1406.4993*, June 2014.

I will give a talk on this topic in Jerusalem on Thursday.

Conclusion

1. Overview of identification strategies for nonlinear SSMs.
2. Conveyed the intuition underlying SMC.
3. Mentioned Particle MCMC and used it for a Wiener problem.
4. Powerful tools useful also outside the class of nonlinear SSMs.

A lot of interesting research that remains to be done!!

Information about a PhD course (*Computational learning in dynamical systems*) on the topic is available via

`user.it.uu.se/~thosc112/CIDS.html`

Manuscript is also available (ask me for a draft if you want)

Thomas B. Schön and Fredrik Lindsten. **Learning of dynamical systems – Particle filters and Markov chain methods**, 2014.

UPPSALA UNIVERSITET

References to some of our work

Self-contained introduction to particle smoothing using 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.

ML identification of nonlinear SSMs

F. Lindsten, **An efficient stochastic approximation EM algorithm using conditional particle filters**, *Proceedings of the 38th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vancouver, Canadan, May 2013.

Thomas B. Schön, Adrian Wills and Brett Ninness. **System Identification of Nonlinear State-Space Models**. *Automatica*, 47(1):39-49, January 2011.

PMCMC for Bayesian identification of nonlinear SSMs (and more)

Johan Dahlin, Fredrik Lindsten and Thomas B. Schön. **Particle Metropolis Hastings using gradient and Hessian information**. *Statistics and Computing*, 2014. (accepted for publication)

Fredrik Lindsten, Michael I. Jordan and Thomas B. Schön. **Particle Gibbs with ancestor sampling**. *Journal of Machine Learning Research (JMLR)*, 15:2145-2184, June 2014.

SMC methods for graphical models

Christian A. Naesseth, Fredrik Lindsten and Thomas B. Schön, **Sequential Monte Carlo methods for graphical models**. *Advances in Neural Information Processing Systems (NIPS)* 27, Montreal, Canada, December, 2014.

Seminar: `http://www.newton.ac.uk/seminar/20140425104011151`

Fredrik Lindsten, Adam M. Johansen, Christian A. Naesseth, Bonnie Kirkpatrick, Thomas B. Schön, John Aston and Alexandre Bouchard-Côté. **Divide-and-Conquer with Sequential Monte Carlo**. *arXiv:1406.4993*, June 2014.