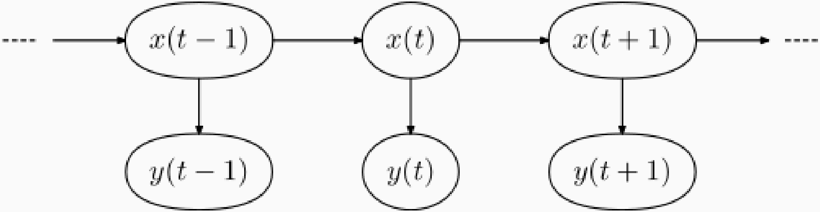
Embedded Hidden Markov Model

Sampling latent state from non-linear state space models

Alan Chau & James Thornton & Valerie Bradley

Department of Statistics, University of Oxford

Hidden Markov Model



Consider a standard state space model with latent state $x_i \in \mathcal{X}$ and observation $y_i \in \mathcal{Y}$ at each time $i \in \{1, ..., n\}$. The system has the following form:

$$
\begin{aligned}
& x_1 \sim \mu(\cdot) \\
\text{Evolution density:} \quad & x_{i+1} \sim f(x_{i+1}|x_i) \\
\text{Observation density:} \quad & y_{i+1} \sim g(y_{i+1}|x_{i+1})
\end{aligned}
$$

Motivation

With non-linear and non-Gaussian f and g, it is difficult to sample from the filtering and posterior distributions using standard MCMC methods. The main difficulties are summarised below:

1. High dimensional state space makes constructing proposals difficult
2. Slow mixing time / Failure to converge

Embedded HMM

The Embedded HMM MCMC procedure introduces L **auxiliary variables** at each time point t, known as **pool states**, $x_t^{[l]} \in \mathcal{X}$ where $l \in \{1, \ldots, L\}$. At each iteration:

1. Update pool states $x_t^{[l]}$ for each l at each time t
2. Update state x_t for each t by sampling from the corresponding pool states, $\{x_t^{[l]}\}_l$.

Embedded HMM: Step 1

Step 1 may be performed by setting $x_t^{[l_t]}$ to x_t (the value of the current state sequence at time t) where l_t is sampled uniformly from $\{1, \dots, L\}$. The remaining $L-1$ pool states are generated by running a Markov chain with the correct invariant distribution $\kappa_t(x)$.

More specifically, the Markov chain transitions must satisfy

$$\kappa_t(x) R_t(x'|x) = \kappa_t(x') \tilde{R}_t(x|x') \tag{1}$$

Embedded HMM: Step 2

We need to calculate the forward probabilities $\alpha_t(x)$ from $t = 1 : n$ for all $x \in \mathcal{P}_t$ to perform step 2.

$$\alpha_1(x) = \frac{p(x)p(y_1|x)}{\kappa_1(x)} \tag{2}$$

$$\alpha_t(x) = \frac{p(y_t|x)}{\kappa_t(x)} \sum_{l=1}^{L} p(x|x_{t-1}^{[l]})\alpha_{t-1}(x_{t-1}^{[l]}) \qquad t > 1 \tag{3}$$

Embedded HMM: Backward Selection

Finally, we sample the new state sequence using the forward probabilities.

1. Select x'_n from $\mathcal{P}_n$ with probabilities proportional to $\alpha_n(x)$
2. For remaining $t = n-1, ..., 1$, sample the state x_t from the pool with probability proportional to $\alpha_{t-1}(x)p(x'_t|x)$

EHMM Pool State Generation

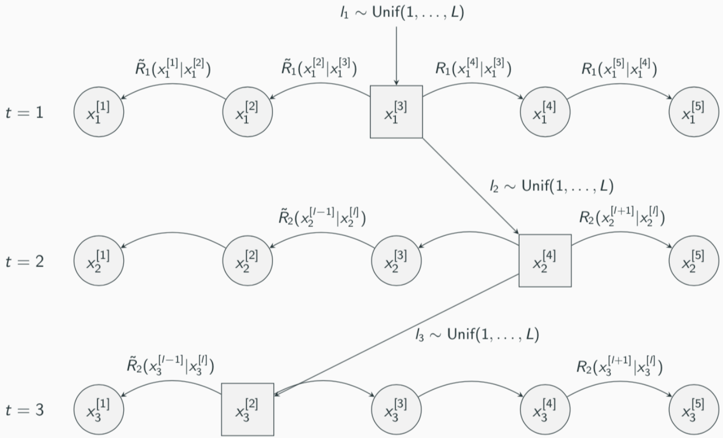


Figure 1: Pool state generation for original Embedded HMM

Illustration of EHMM

We did some simple simulation to illustrate how EHMM are used to capture the latent states of the following models.

- State Transition:

$$P(x_t|x_{t-1}) = \mathcal{N}(x_t|\tanh(\eta x_{t-1}), \sigma^2) \tag{4}$$

- Pool states transition:

$$R_t(x'|x) = \mathcal{N}(x'|\rho x, \tau^2) \tag{5}$$

- Observation output
    - Model 1: Gaussian Output: $\mathcal{N}(y|x, \sigma_m^2)$
    - Model 2: Poisson Output: $Poisson(y|\exp(a + bx))$

Experiment: Gaussian Transition & Gaussian Observations

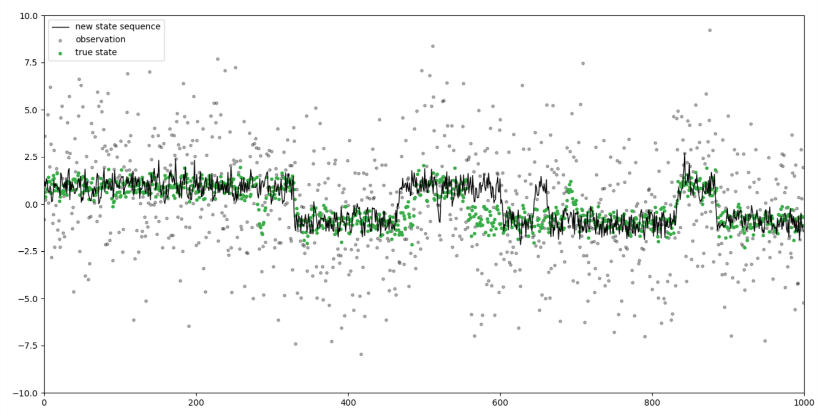


Figure 2: A simple illustration of embedded HMM with Gaussian transition and Gaussian observations. Grey dots being the observation and green dots are the true states. We have set $\sigma_m = 2.5, \sigma = 0.4, \eta = 2.5, \tau = 1, \rho = 0$.

Experiment: Gaussian Transition & Poisson Observation

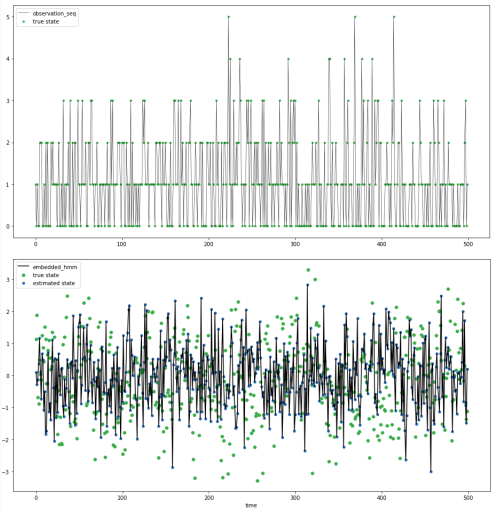


Figure 3: A simple illustration of embedded HMM with Gaussian transition and Poisson observations.

Particle Markov Chain Monte Carlo (PMCMC) [1]

PMCMC: Broad class of methods introduced by Andrieu, Doucet and Hollenstein [2010]

- Alternative method to EHMMs
- Use SMC to construct proposal distributions for MCMC that are efficient in high-dimensions.
- Instead of sampling from $p(x_{1:t}|y_{1:t})$, sample from an unbiased, exact approximation $\hat{p}(x_{1:t}|y_{1:t})$ based on SMC

Focus here: **Particle Gibbs with Backward Sampling** (PGBS)

- Subclass of PMCMC based on Gibbs sampler
- Relies on *conditional* SMC updates - the current state of the sequence is included in the set of particles sampled from in each iteration

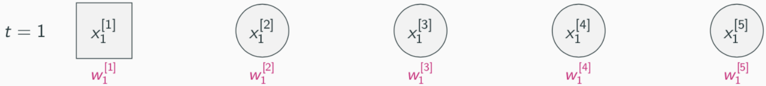
PGBS - Overview

Step 1: First, initialize the sampler with some arbitrary sequence $x_{1:n}$
Step 2: At each iteration:

1. Generate auxillary particles $x_t^{[l]}$, $l \in \{1, \ldots, L\}$, from proposal distribution at each time step $t \in \{1, \ldots, n\}$, conditional on previous time step
2. Sample a new state sequence $x'_{1:n}$ from the generated particles with a stochastic backwards pass

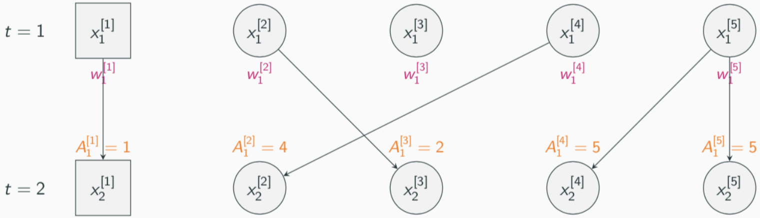
PGBS - Overview

PGBS - Overview

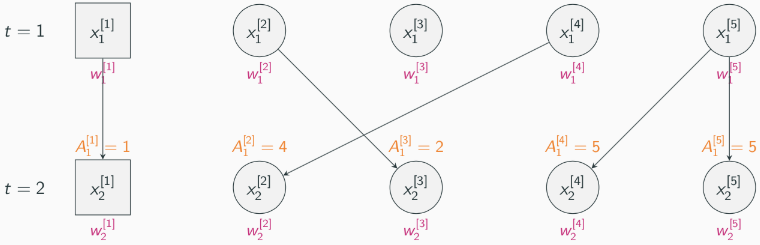


PGBS - Overview

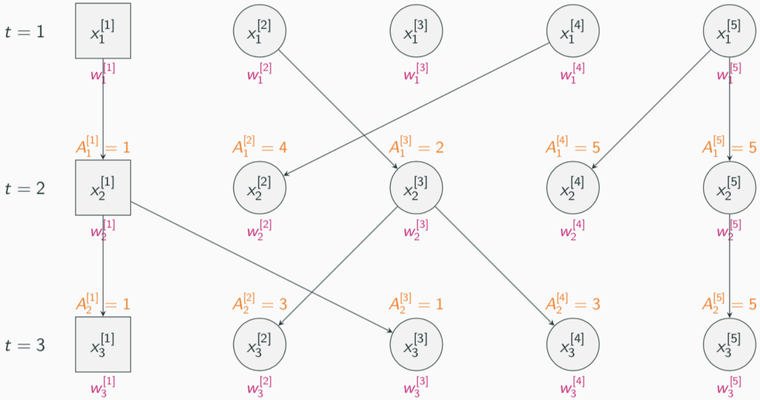
PGBS - Overview



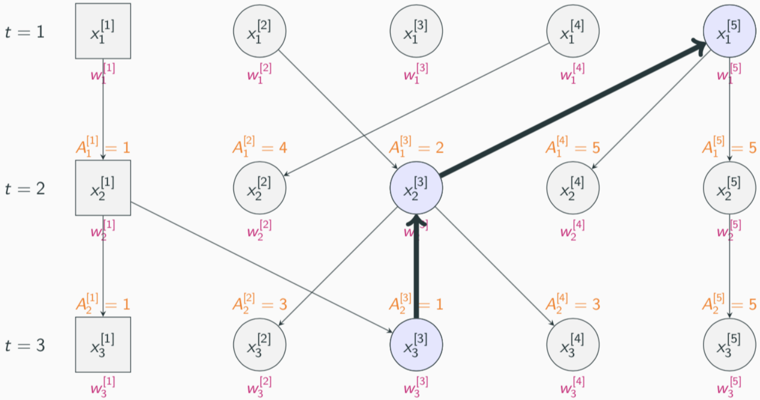
PGBS - Overview



PGBS - Overview



PGBS - Overview



PGBS - Generating Particles $t = 1$

Generating particles

At $t = 1$:

1. Set first particle $x_1^{[1]}$ to current value of the sequence at $t = 1$, x_1
2. Sample $L - 1$ remaining particles from $q(x|y_1)$
3. Calculate normalized importance weights $W_t^{[l]}$ for each particle

$$w_1^{[l]} = \frac{p(x_1^{[l]})p(y_1|x_1^{[l]})}{q_1(x_1^{[l]}|y_1)}$$
$$W_1^{[l]} = \frac{w_1^{[l]}}{\sum_{m=1}^{L} w_1^{[m]}}$$

PGBS - Generating Particles $t > 1$

For $t > 1$:

1. Set first particle $x_t^{[1]} = x_t$, making $A_{t-1}^{[1]} = 1$
2. Sample $L - 1$ remaining ancestor states, $A_{t-1}^{[l]} \forall l > 1$, with probability proportional to importance weights $W_{t-1}^{[l]}$ of particles at $t - 1$
3. Sample new particles $x_t^{[l]}$ from proposal distribution $q_t(x_t^{[l]} | y_t, x_{t-1}^{A_{t-1}^{[l]}})$
4. Calculate normalized importance weights of new particles

$$w_t^{[l]} = \frac{p(x_t^{[l]} | x_{t-1}^{A_{t-1}^{[l]}}) p(y_t | x_t^{[l]})}{q_t(x_t^{[l]} | y_t, x_{t-1}^{A_{t-1}^{[l]}})}$$

$$W_t^{[l]} = \frac{w_t^{[l]}}{\sum_{m=1}^{L} w_t^{[m]}}$$

PGBS - Backward Sampling

Sample new state sequence x' from particles with backward pass

1. Sample x'_n, new state at time n, with probability proportional to $W_n^{[l]}$
2. Set $x'_{n-1} = x_{n-1}^{[l]}$ with probability

$$\frac{w_t^{[l]} p(x'_{t+1} | x_t^{[l]})}{\sum_{m=1}^{L} w_t^{[m]} p(x'_{t+1} | x_t^{[m]})}$$

Extended EHMM [2]

Recall EHMM: Pool states generated *independently* at each time step from Markov chain with invariant density κ_t. Calculate forward probabilities $\alpha_t(x)$ and select new state sequence with backward pass.

Extension from EHMM: Generate pool states *sequentially*. Specifically, set pool densities κ_t^f to

$$\kappa_1^f(x) \propto p(x)p(y_1|x)$$

$$\kappa_t^f(x|\mathcal{P}_{t-1}) \propto p(y_t|x) \sum_{l=1}^{L} p(x|x_{t-1}^{[l]})$$

Pro: Forward probabilities are all constant

Con: Complexity is still of $\mathcal{O}(nL^2)$, instead of $\mathcal{O}(nL)$ of PGBS

Extended EHMM [2]

Solution: Think of κ_t^f as the marginal of an augmented pool state density

$$\lambda_t(x, l) \propto p(y_t|x)p(x|x_{t-1}^{[l]})$$

on extended space $l \in \{1, \ldots, L\}$.

Pro: No more sums, so computation time is O(nL)!
Con: Proposals based on single pool state $x_{t-1}^{[l]}$ instead of entire distribution at the previous time step $\mathcal{P}_{t-1}$

Pool State Samplers [2]

One can sample from $\lambda_t(x, l)$ in a variety of ways, including via Metropolis Hastings with the following proposals:

- **Autoregressive Metropolis** for x
  - Given target $p(x)p(y|x)$ and $x \sim \mathcal{N}(\mu, \Sigma)$
  - Sample $x'|x$ from: $x' = \mu + \sqrt{1-\epsilon^2}(x - \mu) + \epsilon L n$ where L is the lower triangle form the Cholesky decomposition of Σ, $\epsilon \in (-1, 1)$ and $n \sim \mathrm{N}(0, I^d)$
- **Shift Sampler** for (x, l)
  - l' is sampled from any method on support $\{1, \ldots, L\}$
  - x' is proposed such that the relationship between x'_t and $x_{t-1}^{[l']}$ mirrors that of x_t and $x_{t-1}^{[l]}$ e.g. $x'_t = x_t - x_{t-1}^{[l]} + x_{t-1}^{[l']}$
- **Flip updates**
  - For every $x_t^{[l]}$ included in the pool, also include $-x_t^{[l+1]}$
  - Alternate between flip update and another method

Experiment: Gaussian Transition & Poisson Observations

Consider the following model, as detailed in [2], for $n = 250$ and $P = 10$, $\forall t \in \{1, \dots, n\}$, $\forall j \in \{1, \dots, P\}$:

$$X_1 \sim \mathcal{N}(0, \Sigma_{init})$$
$$X_t \sim \mathcal{N}(0.9x_{t-1}, \Sigma)$$
$$X_t = (X_{t,1}, \dots, X_{t,P})'$$
$$Y_{t,j} \sim Po(exp(-0.4 + 0.6x_{t,j}))$$
$$\{\Sigma_{init}\}_{i,j} = \begin{cases} \frac{0.7}{1-0.9^2} & \text{if } i \neq j \\ \frac{1}{1-0.9^2} & \text{if } i = j \end{cases}$$
$$\{\Sigma\}_{i,j} = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j \end{cases}$$

Sampling schemes:

- Metropolis Hastings with Autoregressive Proposal (MH)
- Particle Gibbs Backward Sampling (PGBS)
- Particle Gibbs Backward Sampling with Metropolis Hastings (PGBS-MH)
- Embedded HMM MCMC (EHMM)

Experiment: Gaussian Transition & Poisson Observations

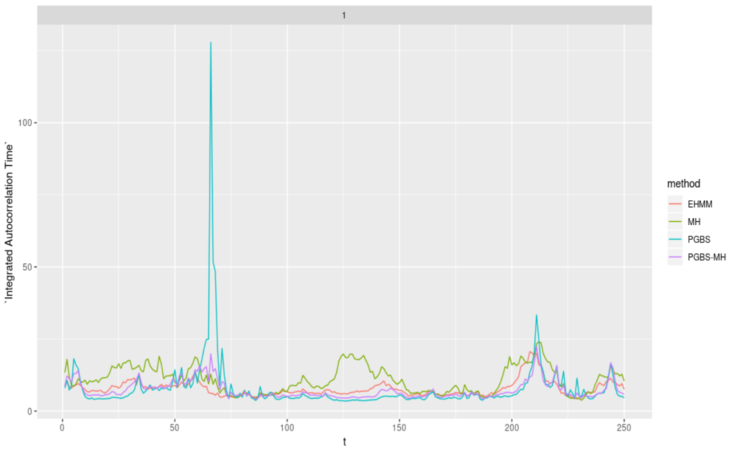


Figure 4: Integrated Autocorrelation Time for $x_{t,1}$ through time for the various sampling procedures

Experiment: Gaussian Transition & Poisson Observations

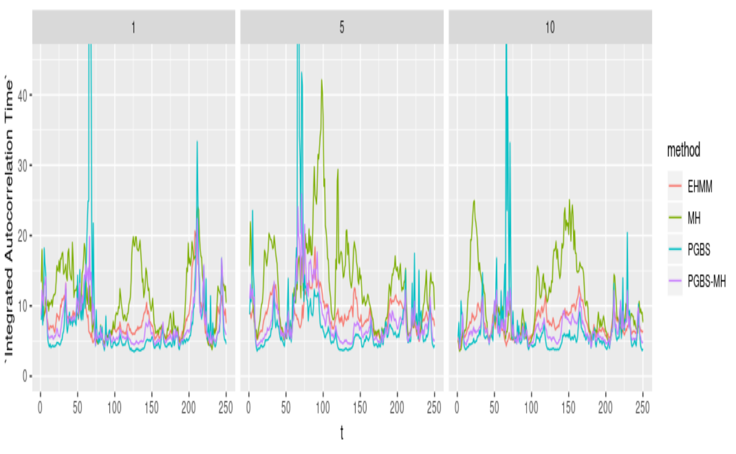


Figure 5: Integrated Autocorrelation Time for $x_{t,1}, x_{t,5}, x_{t,10}$ through time for the various sampling procedures, zoomed-in

Experiment: Gaussian Transition & Poisson Observations

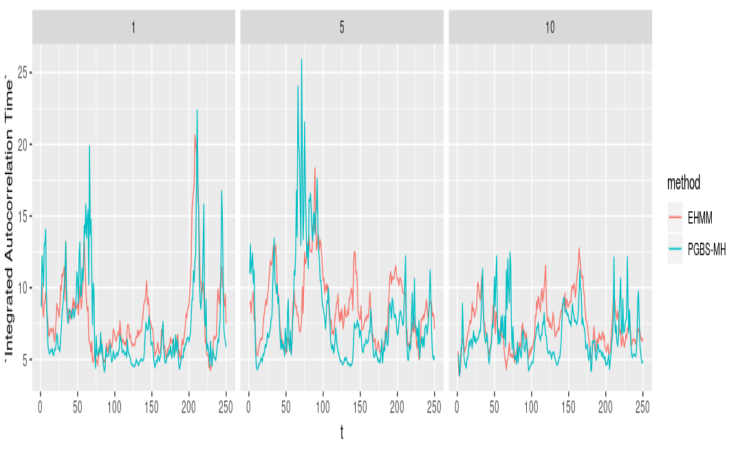


Figure 6: Integrated Autocorrelation Time for $x_{t,1}, x_{t,5}, x_{t,10}$ through time for the EHMM and PGBS-MH sampling procedures, zoomed-in

References

Andrieu C, Doucet A, Holenstein R.
Particle markov chain monte carlo methods.
Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2010;72(3):269–342.

Shestopaloff AY, Neal RM, et al.
Sampling latent states for high-dimensional non-linear state space models with the embedded HMM method.
Bayesian Analysis. 2018;13(3):797–822.