An Adaptive Sequential Monte Carlo Approach for Bayesian Model Comparison

Yan Zhou   Adam M. Johansen   John A. D. Aston

University of Warwick

September 21, 2012

Outline

Bayesian model comparison

Basic formulas

Given a collection of (at most countable) models $\{M_k\}_{k\in\mathcal{K}}$,

$$\pi(M_k|\boldsymbol{y}) = \frac{\pi(M_k)p(\boldsymbol{y}|M_k)}{p(\boldsymbol{y})}$$

$$p(\boldsymbol{y}|M_k) = \int_{\Theta_k} \pi(\theta_k|M_k)p(\boldsymbol{y}|\theta_k, M_k)\, \mathrm{d}\,\theta_k$$

Bayesian model comparison

Basic formulas

Given a collection of (at most countable) models $\{M_k\}_{k \in \mathcal{K}}$,

$$\pi(M_k|\boldsymbol{y}) = \frac{\pi(M_k)p(\boldsymbol{y}|M_k)}{p(\boldsymbol{y})}$$

$$p(\boldsymbol{y}|M_k) = \int_{\Theta_k} \pi(\theta_k|M_k)p(\boldsymbol{y}|\theta_k, M_k)\, \mathrm{d}\,\theta_k$$

Common approaches

Evaluate posterior model probabilities $\pi(M_k|\boldsymbol{y})$ directly

Evaluate marginal likelihood $p(\boldsymbol{y}|M_k)$ individually

Evaluate Bayes factors $\frac{p(\boldsymbol{y}|M_{k+1})}{p(\boldsymbol{y}|M_k)}$ sequentially

Sequential Monte Carlo Approach for Bayesian model comparison

What do we want?

Better estimates at *less computational cost* with *less manual calibration*

Sequential Monte Carlo Approach for Bayesian model comparison

What do we want?

Better estimates at *less computational cost* with *less manual calibration*

Better estimates

Unbiased or almost unbiased

Consistent

Smaller variance or smaller MSE if biased

Sequential Monte Carlo Approach for Bayesian model comparison

What do we want?

Better estimates at *less computational cost* with *less manual calibration*

Better estimates

Unbiased or almost unbiased

Consistent

Smaller variance or smaller MSE if biased

Less computational cost

Smaller number of samples

Leverage more efficient computational resources – parallel computing

Sequential Monte Carlo Approach for Bayesian model comparison

What do we want?

Better estimates at *less computational cost* with *less manual calibration*

Better estimates

Unbiased or almost unbiased

Consistent

Smaller variance or smaller MSE if biased

Less computational cost

Smaller number of samples

Leverage more efficient computational resources – parallel computing

Less manual calibration

Generic approach

Adaptive strategies

Sequential Monte Carlo

Initialization from $\eta_0(X)$ for $\pi_0(X)$

Iterate over intermediate distributions $\{\pi_t(X) = \gamma_t(X)/Z_t\}_{t=1}^{T}$

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo

Draw $\{X_0^{(i)}\}_{i=1}^N$ from η_0

Compute $\{W_0^{(i)}\}_{i=1}^N$

Iterate over intermediate distributions $\{\pi_t(X) = \gamma_t(X)/Z_t\}_{t=1}^T$

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo

Draw $\{X_0^{(i)}\}_{i=1}^N$ from η_0
Compute $\{W_0^{(i)}\}_{i=1}^N$

Resampling if necessary
Draw $X_t^{(i)}$ from $K_t(X_{t-1}^{(i)}, \cdot)$ for $i = 1, \dots, N$
Compute incremental weights $\{\tilde{w}_t^{(i)}(X_{t-1}^{(i)}, X_t^{(i)})\}_{i=1}^N$
Compute normalized weights $\{W_t^{(i)}\}_{i=1}^N$

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo

Draw $\{X_0^{(i)}\}_{i=1}^N$ from η_0

Compute $\{W_0^{(i)}\}_{i=1}^N$

Resampling if necessary

Draw $X_t^{(i)}$ from $K_t(X_{t-1}^{(i)}, \cdot)$ for $i = 1, \dots, N$

Compute incremental weights $\{\tilde{w}_t^{(i)}(X_{t-1}^{(i)}, X_t^{(i)})\}_{i=1}^N$

Compute normalized weights $\{W_t^{(i)}\}_{i=1}^N$

$\pi_t^N(\mathrm{d}\,x) = \sum_{i=1}^N W_t^{(i)} \delta_{X_t^{(i)}}(\mathrm{d}\,x)$ approxiamte $\pi_t(\mathrm{d}\,x)$

$\frac{\hat{Z}_t}{Z_{t-1}} = \sum_{i=1}^N W_{t-1}^{(i)} \tilde{w}_t^{(i)}$ estimates the ratio of normalizing constants recusively

Sequential Monte Carlo – Evaluate $\pi(M_k|\boldsymbol{y})$ directly

Initialization from $\eta_0(X)$ for $\pi_0(X)$

Iterate over intermediate distributions $\{\pi_t(X) = \gamma_t(X)/Z_t\}_{t=1}^{T}$

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo – Evaluate $\pi(M_k|\boldsymbol{y})$ directly

$$\eta_0(\theta_0, M_0) = \pi_0(\theta_0, M_0) \propto \pi(M_0)\pi(\theta_0|M_0)$$

Iterate over intermediate distributions $\{\pi_t(X) = \gamma_t(X)/Z_t\}_{t=1}^{T}$

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo – Evaluate $\pi(M_k|\boldsymbol{y})$ directly

$$\eta_0(\theta_0, M_0) = \pi_0(\theta_0, M_0) \propto \pi(M_0)\pi(\theta_0|M_0)$$

$$\pi_t(\theta_t, M_t) \propto \pi(M_t)\pi(\theta_t|M_t)p(\boldsymbol{y}|\theta_t, M_t)^{\alpha(t/T)}$$

Markov kernel $K_t(X_{t-1}, \cdot)$ requires both within- and inter-model moves

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo – Evaluate $\pi(M_k|\boldsymbol{y})$ directly

$$\eta_0(\theta_0, M_0) = \pi_0(\theta_0, M_0) \propto \pi(M_0)\pi(\theta_0|M_0)$$

$$\pi_t(\theta_t, M_t) \propto \pi(M_t)\pi(\theta_t|M_t)p(\boldsymbol{y}|\theta_t, M_t)^{\alpha(t/T)}$$

Markov kernel $K_t(X_{t-1}, \cdot)$ requires both within- and inter-model moves

Estimate $\pi(M_k|\boldsymbol{y})$ using particle approximation to $\pi_T(\theta_T, M_T)$

Sequential Monte Carlo – Evaluate $\pi(\boldsymbol{y}|M_k)$ individually

Initialization from $\eta_0(X)$ for $\pi_0(X)$

Iterate over intermediate distributions $\{\pi_t(X) = \gamma_t(X)/Z_t\}_{t=1}^{T}$

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo – Evaluate $\pi(\boldsymbol{y}|M_k)$ individually

$$\eta_0(\theta_0) = \pi_0(\theta_0) \propto \pi(\theta_0|M_k)$$

Iterate over intermediate distributions $\{\pi_t(X) = \gamma_t(X)/Z_t\}_{t=1}^{T}$

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo – Evaluate $\pi(\boldsymbol{y}|M_k)$ individually

$$\eta_0(\theta_0) = \pi_0(\theta_0) \propto \pi(\theta_0|M_k)$$

$\pi_t(\theta_t) \propto \pi(\theta_t|M_k)p(\boldsymbol{y}|\theta_t, M_k)^{\alpha(t/T)}$ or

$\pi_t(\theta_t) \propto \pi(\theta_t|M_k)p(\boldsymbol{y}_{1:t}|\theta_t, M_k)$ (Chopin, 2002)

Markov kernel $K_t(X_{t-1}, \cdot)$ only within-model moves

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo – Evaluate $\pi(\boldsymbol{y}|M_k)$ individually

$$\eta_0(\theta_0) = \pi_0(\theta_0) \propto \pi(\theta_0|M_k)$$

$\pi_t(\theta_t) \propto \pi(\theta_t|M_k)p(\boldsymbol{y}|\theta_t, M_k)^{\alpha(t/T)}$ or
$\pi_t(\theta_t) \propto \pi(\theta_t|M_k)p(\boldsymbol{y}_{1:t}|\theta_t, M_k)$ (Chopin, 2002)
Markov kernel $K_t(X_{t-1}, \cdot)$ only within-model moves

Estimate $p(\boldsymbol{y}|M_k)$, the normalizing constant ratio Z_T/Z_0

Sequential Monte Carlo – Evaluate $\frac{p(\boldsymbol{y}|M_{k+1})}{p(\boldsymbol{y}|M_k)}$ sequentially

Initialization from $\eta_0(X)$ for $\pi_0(X)$

Iterate over intermediate distributions $\{\pi_t(X) = \gamma_t(X)/Z_t\}_{t=1}^{T}$

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo – Evaluate $\frac{p(\boldsymbol{y}|M_{k+1})}{p(\boldsymbol{y}|M_k)}$ sequentially

$\pi_0(\theta_0) \propto \pi(\theta_0|M_k)p(\boldsymbol{y}|\theta_0, M_k)$

$\eta_0(\theta_0)$: The particle system of the sampler iterating from model M_{k-1} to M_k.

Iterate over intermediate distributions $\{\pi_t(X) = \gamma_t(X)/Z_t\}_{t=1}^{T}$

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo – Evaluate $\frac{p(\boldsymbol{y}|M_{k+1})}{p(\boldsymbol{y}|M_k)}$ sequentially

$$\pi_0(\theta_0) \propto \pi(\theta_0|M_k)p(\boldsymbol{y}|\theta_0, M_k)$$

$\eta_0(\theta_0)$: The particle system of the sampler iterating from model M_{k-1} to M_k.

$$\pi_t(\theta_t, M_t) \propto \pi_t(M_t)\pi(\theta_t|M_t)p(\boldsymbol{y}|\theta_t, M_t)$$

$$\pi_t(M_t) = \alpha(t/T)$$

Markov kernel $K_t(X_{t-1}, \cdot)$ requires both within- and inter-model moves

Terminiate at $\pi_T(X)$ and estimation

Sequential Monte Carlo – Evaluate $\frac{p(\boldsymbol{y}|M_{k+1})}{p(\boldsymbol{y}|M_k)}$ sequentially

$$\pi_0(\theta_0) \propto \pi(\theta_0|M_k)p(\boldsymbol{y}|\theta_0, M_k)$$

$\eta_0(\theta_0)$: The particle system of the sampler iterating from model M_{k-1} to M_k.

$$\pi_t(\theta_t, M_t) \propto \pi_t(M_t)\pi(\theta_t|M_t)p(\boldsymbol{y}|\theta_t, M_t)$$
$$\pi_t(M_t) = \alpha(t/T)$$

Markov kernel $K_t(X_{t-1}, \cdot)$ requires both within- and inter-model moves

Estimate Bayes factor $B_{k+1,k}$, the normalizing constant ratio Z_T/Z_0

Sequential Monte Carlo – Path sampling estimator

Basic identity

Given a family of distributions $\{\pi_\alpha(x) = q_\alpha(x)/Z_\alpha\}_{\alpha\in[0,1]}$

$$\log\left(\frac{Z_1}{Z_0}\right) = \int_0^1 \mathbb{E}_\alpha\left[\frac{\mathrm{d}\log q_\alpha(X)}{\mathrm{d}\,\alpha}\right]\mathrm{d}\,\alpha$$

Sequential Monte Carlo – Path sampling estimator

Basic identity

Given a family of distributions $\{\pi_\alpha(x) = q_\alpha(x)/Z_\alpha\}_{\alpha\in[0,1]}$

$$\log\left(\frac{Z_1}{Z_0}\right) = \int_0^1 \mathbb{E}_\alpha\left[\frac{\mathrm{d}\log q_\alpha(X)}{\mathrm{d}\,\alpha}\right]\mathrm{d}\,\alpha$$

Using SMC samples

Particle approximations of the expectations from SMC samplers

Numerical integration to approximate the estimator

Sequential Monte Carlo – Path sampling estimator

Basic identity

Given a family of distributions $\{\pi_\alpha(x) = q_\alpha(x)/Z_\alpha\}_{\alpha\in[0,1]}$

$$\log\left(\frac{Z_1}{Z_0}\right) = \int_0^1 \mathbb{E}_\alpha\left[\frac{\mathrm{d}\log q_\alpha(X)}{\mathrm{d}\,\alpha}\right] \mathrm{d}\,\alpha$$

Using SMC samples

Particle approximations of the expectations from SMC samplers

Numerical integration to approximate the estimator

Andrew Gelman and Xiao-Li Meng (1998). "Simulating normalizing constants: From importance sampling to bridge sampling to path sampling". In: *Statistical Science* 13.2, pp. 163–185

Simple illustrative example: Gaussian mixture model

Model

Determine the number of components k, which define the model by

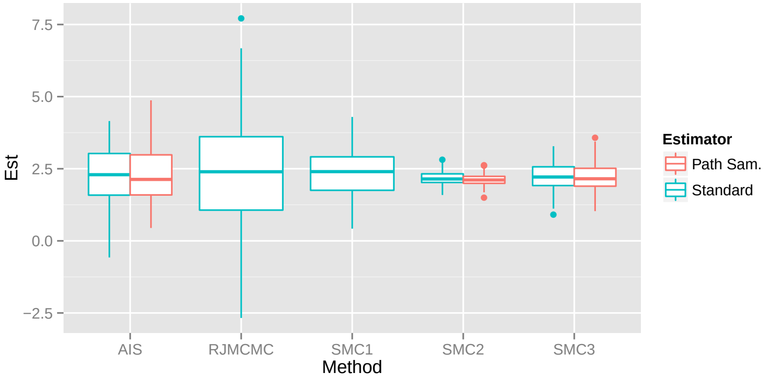
$$y_i|\theta_k \sim \sum_{j=1}^{k} \omega_j \mathcal{N}(\mu_j, \lambda_j^{-1}) \qquad i = 1, \ldots, n$$

$$\mu_j \sim \mathcal{N}(\xi, \kappa^{-1}) \quad \lambda_j \sim \mathcal{G}(\nu, \chi) \quad \omega_{1:k} \sim \mathcal{D}(\rho)$$

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra (2006). "Sequential Monte Carlo samplers". In: *Journal of Royal Statistical Society B* 68.3, pp. 411–436

Simple illustrative example: Gaussian mixture model

Comparison of estimates of Bayes factor $B_{5,4}$



- AIS & SMC: 1,000 particles, 100 time steps, $\alpha(t/T) = (t/T)^2$
- RJMCMC: 100,000 iterations

Adaptive strategies – Specification of distributions

Purpose

Create a *smooth* sequence of distributions that reduces discrepancy between π_{t-1} and π_t

Adaptive strategies – Specification of distributions

Purpose

Create a *smooth* sequence of distributions that reduces discrepancy between π_{t-1} and π_t

Assumption

The criterion for adaptation can be calculated prior to the sampling of next iteration.
For example, $\tilde{w}_t(X_{t-1}, X_t) \propto \pi_t(X_{t-1})/\pi_{t-1}(X_t)$ when $K_t(X_{t-1}, \cdot)$ is π_t invariant

Adaptive strategies – Specification of distributions

Purpose

Create a *smooth* sequence of distributions that reduces discrepancy between π_{t-1} and π_t

Assumption

The criterion for adaptation can be calculated prior to the sampling of next iteration.
For example, $\tilde{w}_t(X_{t-1}, X_t) \propto \pi_t(X_{t-1})/\pi_{t-1}(X_t)$ when $K_t(X_{t-1}, \cdot)$ is π_t invariant

Why does it matter?

"the variance of $(\tilde{w}_t)$ will typically be high if the discrepancy between π_{t-1} and π_t is large *even if the kernel K_t mixes very well*" (Del Moral, Doucet, and Jasra, 2006)
Recall normalizing constants estimator relates directly to $\tilde{w}_t$

Adaptive strategies – Specification of distributions

Purpose

Create a *smooth* sequence of distributions that reduces discrepancy between π_{t-1} and π_t

Assumption

The criterion for adaptation can be calculated prior to the sampling of next iteration.
For example, $\tilde{w}_t(X_{t-1}, X_t) \propto \pi_t(X_{t-1})/\pi_{t-1}(X_t)$ when $K_t(X_{t-1}, \cdot)$ is π_t invariant

Why does it matter?

"the variance of $(\tilde{w}_t)$ will typically be high if the discrepancy between π_{t-1} and π_t is large *even if the kernel K_t mixes very well*" (Del Moral, Doucet, and Jasra, 2006)
Recall normalizing constants estimator relates directly to $\tilde{w}_t$

Desired effect of the adaptive strategy

Independent of resampling strategies – it is a property of the sequence of distributions

Adaptive strategies – Specification of distributions

Using ESS (Jasra et al., 2008; Schäfer and Chopin, 2011)

$$\text{ESS}_t = \frac{\left(\sum_{j=1}^{N} W_{t-1}^{(j)} \tilde{w}_t^{(j)}\right)^2}{\sum_{j=1}^{N} (W_{t-1}^{(j)})^2 (\tilde{w}_t^{(j)})^2}$$

At time $t-1$, find $\alpha(t/T)$ such that ESS_t equal a specific value

Adaptive strategies – Specification of distributions

Using ESS (Jasra et al., 2008; Schäfer and Chopin, 2011)

$$\mathrm{ESS}_t = \frac{(\sum_{j=1}^{N} W_{t-1}^{(j)} \tilde{w}_t^{(j)})^2}{\sum_{j=1}^{N} (W_{t-1}^{(j)})^2 (\tilde{w}_t^{(j)})^2}$$

At time $t-1$, find $\alpha(t/T)$ such that ESS_t equal a specific value

Caveats

The sequence of distributions depends on the resampling strategies

Simple illustrative example: Gaussian mixture model

Consider a SMC sampler on $\{\pi_t(\theta_t)\}_{t=0}^{T}$

$$\pi_t(\theta_t) \propto \pi(\theta_t | M_k) p(\boldsymbol{y} | \theta_t, M_k)^{\alpha(t/T)}$$

Simple illustrative example: Gaussian mixture model

Consider a SMC sampler on $\{\pi_t(\theta_t)\}_{t=0}^{T}$

$$\pi_t(\theta_t) \propto \pi(\theta_t|M_k)p(\boldsymbol{y}|\theta_t, M_k)^{\alpha(t/T)}$$

Problem

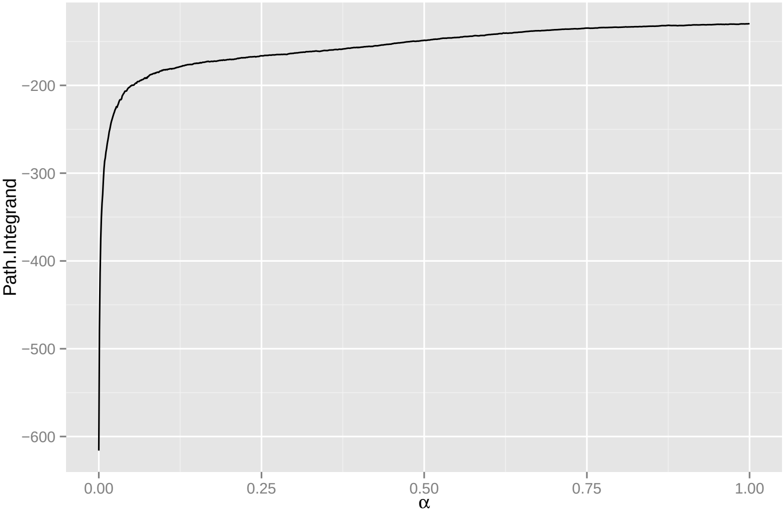
At each $\alpha(t/T)$, find the $\Delta\alpha(t/T) = \alpha((t+1)/T) - \alpha(t/T)$. How does $\Delta\alpha$ evolve along with α?
Does the adaptive specification of the sequence of distributions, $\alpha(t/T)$ improve the normalizing constant estimator?

Benchmark

Comparison to $\alpha(t/T)$ = t/T

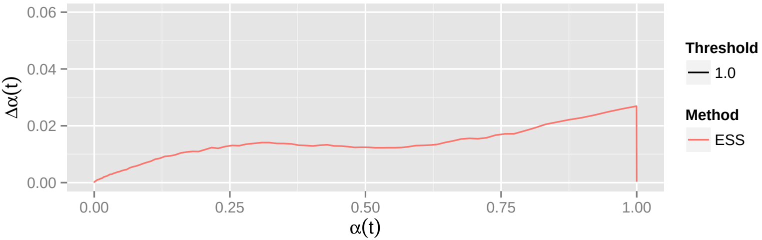
Simple illustrative example: Gaussian mixture model

Change of path sampling integrands ($\log p(\boldsymbol{y}|\theta_t, M_k)$)



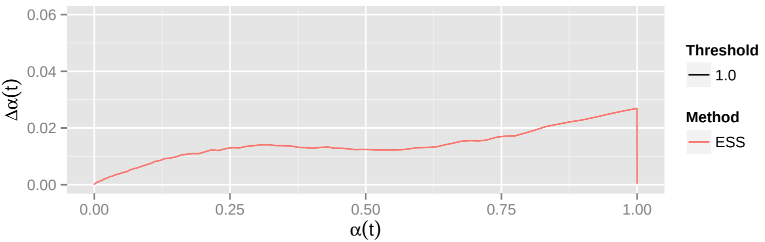
Adaptive strategies – Specification of distributions

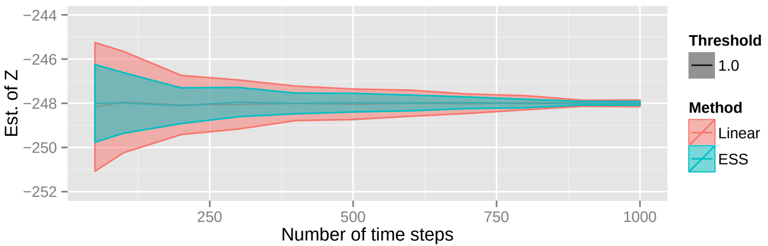
Using ESS– Resampling every iteration (Schäfer and Chopin, 2011)



Adaptive strategies – Specification of distributions

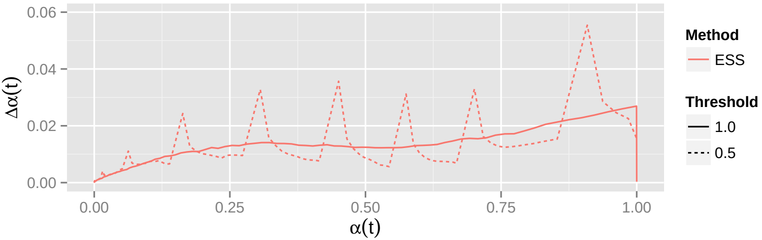
Using ESS– Resampling every iteration (Schäfer and Chopin, 2011)





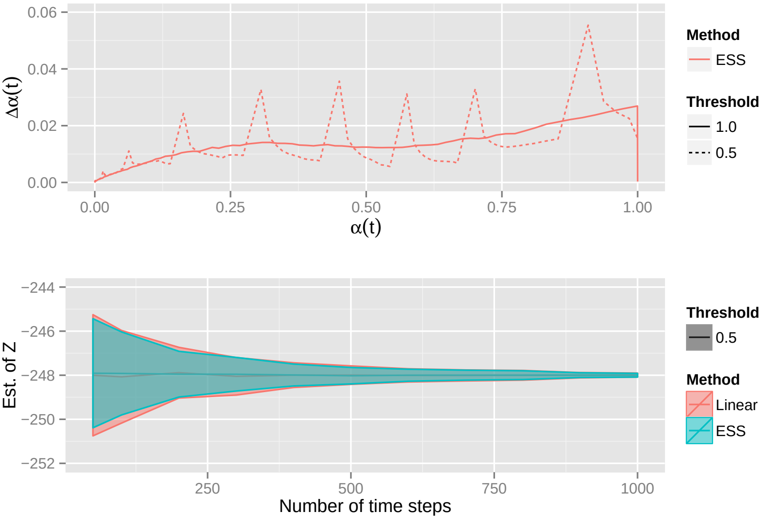
Adaptive strategies – Specification of distributions

Using ESS– Resampling only when ESS $< N/2$ (Jasra et al., 2008)



Adaptive strategies – Specification of distributions

Using ESS– Resampling only when ESS $< N/2$ (Jasra et al., 2008)



Adaptive strategies – Specification of distributions

A new approach: CESS– Conditional ESS

$$\text{CESS}_t = \frac{\left(\sum_{j=1}^{N} W_{t-1}^{(j)} \tilde{w}_t^{(j)}\right)^2}{\sum_{j=1}^{N} \frac{1}{N} W_{t-1}^{(j)} (\tilde{w}_t^{(j)})^2}$$

Adaptive strategies – Specification of distributions

A new approach: CESS– Conditional ESS

$$\mathrm{CESS}_t = \frac{\left(\sum_{j=1}^{N} W_{t-1}^{(j)} \tilde{w}_t^{(j)}\right)^2}{\sum_{j=1}^{N} \frac{1}{N} W_{t-1}^{(j)} \left(\tilde{w}_t^{(j)}\right)^2}$$

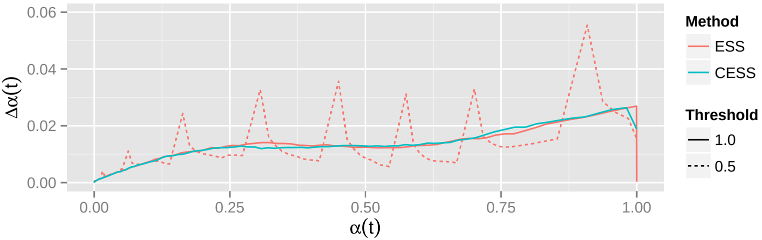
Properties

Approximate the ESS as if resampling at time $t-1$ without actually doing it

Produce the same sequence regardless of resampling strategy

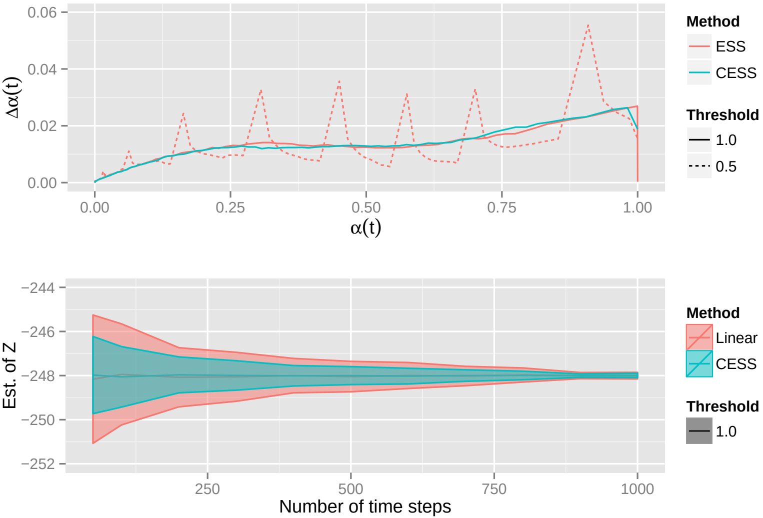
Adaptive strategies – Specification of distributions

Using CESS– Resampling every iteration



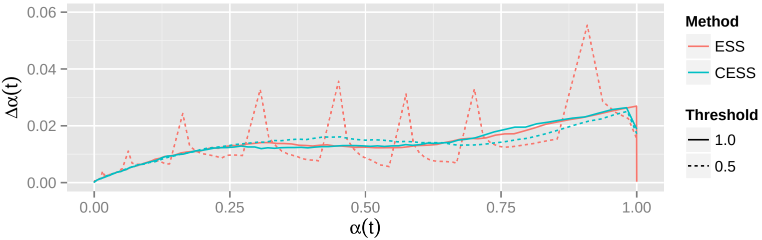
Adaptive strategies – Specification of distributions

Using CESS– Resampling every iteration



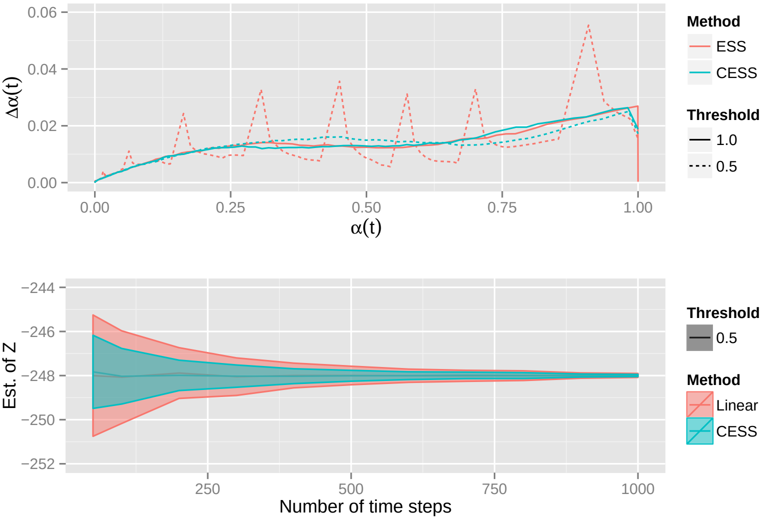
Adaptive strategies – Specification of distributions

Using CESS– Resampling only when ESS $< N/2$



Adaptive strategies – Specification of distributions

Using CESS– Resampling only when ESS $< N/2$



Adaptive strategies – Calibrating RWM or MALA proposal scales

Estimating moments of parameters from particle approximations

Performance: SMC vs AIS

Annealed importance resampling

SMC without resampling

Some argues SMC does not improve results for normalizing constant estimates

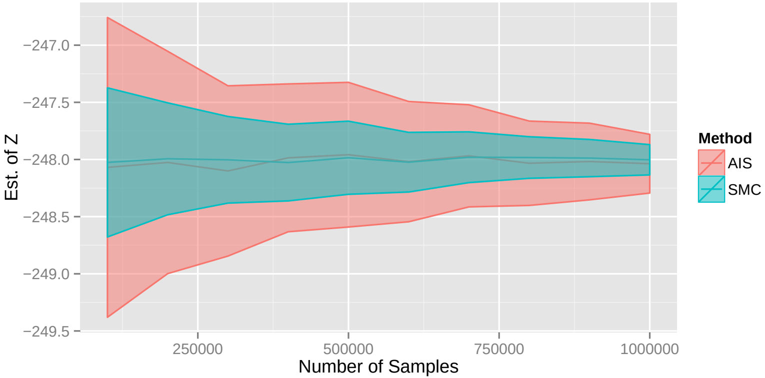
Performance: SMC vs AIS

Annealed importance resampling

SMC without resampling

Some argues SMC does not improve results for normalizing constant estimates

Effects of resampling in estimating normalizing constants



Performance: Path sampling using SMC vs Population-MCMC

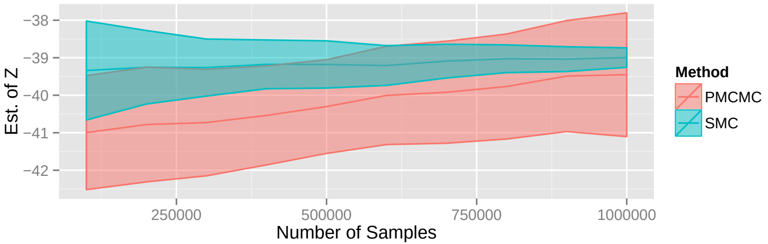
Population-MCMC with path sampling estimator (Calderhead and Girolami, 2009)

Sampling parallel MCMC chains for $\pi(X_{0:T}) = \prod_{t=0}^{T} \pi_t(X_t)$, with local mixing and global swap/crossover moves

Performance: Path sampling using SMC vs Population-MCMC

Population-MCMC with path sampling estimator (Calderhead and Girolami, 2009)

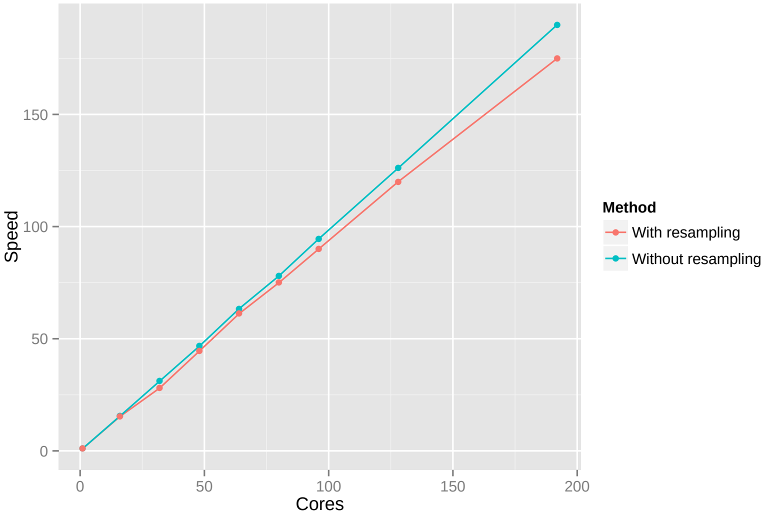
Sampling parallel MCMC chains for $\pi(X_{0:T}) = \prod_{t=0}^{T} \pi_t(X_t)$, with local mixing and global swap/crossover moves



SMC: Fix number of particles $N = 1000$; Population-MCMC: Fix number of iterations $N = 10000$

Performance: Scalability on GPU parallelization

Implemented with OpenCL on NVIDIA Quadro 2000



Summary

Bayesian model comparison via Sequential Mote Carlo

- Can be used as drop-in replacement where conventional MCMC, RJMCMC, etc., were used
- Requires minimal manual calibration
- Can provide better and more robust performance
- Can be parallelized straightforwardly

Thank You!