Random thoughts about scalable Bayesian computing

Matti Vihola (Jyväskylä)

Colloquium of the Department of Mathematics and Statistics @ University of Helsinki
29 March 2023

ACADEMY OF FINLAND

FINNISH CENTRE OF EXCELLENCE IN RANDOMNESS AND STRUCTURES 2022–2029

UNIVERSITY OF JYVÄSKYLÄ
DEPARTMENT OF MATHEMATICS AND STATISTICS

Introduction

- Computing limits the use of Bayesian methods 😐
- Important to develop scalable methods for challenging problems 😃
- What scalability means and which methods are scalable? 🤔

Bayesian inference problem

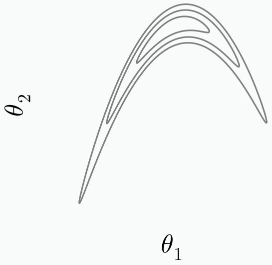
- Posterior density of the form:

$$\boxed{\pi(\boldsymbol{\theta}) := p(\boldsymbol{\theta} \mid \boldsymbol{y}) \propto \mathrm{pr}(\boldsymbol{\theta}) L(\boldsymbol{\theta}; \boldsymbol{y}) =: \pi_u(\boldsymbol{\theta}),}$$

  with prior $\mathrm{pr}(\boldsymbol{\theta})$, likelihood $L(\boldsymbol{\theta}; \boldsymbol{y})$ and
  - p unknowns: $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$
  - n data records: $\boldsymbol{y} = (y_1, \ldots, y_n)$
- 'Inference' = calculating probabilities/expectations

$$\mathbb{E}_\pi[\phi(\boldsymbol{\Theta})] = \int_{\mathbb{R}^p} \pi(x)\phi(x)\mathrm{d}x, \quad \phi : \mathbb{R}^p \to \mathbb{R} \text{ test function(s)}$$

- Can point-wise evaluate $\pi_u(\boldsymbol{\theta})$ (and $\nabla \log \pi_u(\boldsymbol{\theta}) = \nabla \log \pi(\boldsymbol{\theta})$)



Bayesian inference problem

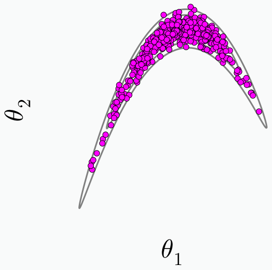
- Posterior density of the form:

$$\boxed{\pi(\boldsymbol{\theta}) := p(\boldsymbol{\theta} \mid \boldsymbol{y}) \propto \mathrm{pr}(\boldsymbol{\theta}) L(\boldsymbol{\theta}; \boldsymbol{y}) =: \pi_u(\boldsymbol{\theta}),}$$

  with prior $\mathrm{pr}(\boldsymbol{\theta})$, likelihood $L(\boldsymbol{\theta}; \boldsymbol{y})$ and
  - p unknowns: $\boldsymbol{\theta} = (\theta_1, \dots, \theta_p)$
  - n data records: $\boldsymbol{y} = (y_1, \dots, y_n)$
- 'Inference' = calculating probabilities/expectations

$$\mathbb{E}_\pi[\phi(\boldsymbol{\Theta})] = \int_{\mathbb{R}^p} \pi(x)\phi(x)\mathrm{d}x, \quad \phi : \mathbb{R}^p \to \mathbb{R} \text{ test function(s)}$$

- Can point-wise evaluate $\pi_u(\boldsymbol{\theta})$ (and $\nabla \log \pi_u(\boldsymbol{\theta}) = \nabla \log \pi(\boldsymbol{\theta})$)
- Monte Carlo: $\frac{1}{m} \sum_{k=1}^m \phi(\boldsymbol{\Theta}_k) \approx \mathbb{E}_\pi[\phi(\boldsymbol{\Theta})]$



Case 1: Big data scalability

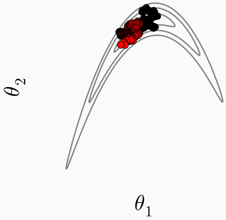
A lot of data but moderate number of unknowns

- Data $\boldsymbol{y} = (y_1, \ldots, y_n)$ where $n \to \infty$ (very large)
- Unknowns $\boldsymbol{\theta} \in \mathbb{R}^p$ where $p = \mathrm{constant}$ (small or moderate)

A lot of data but moderate number of unknowns

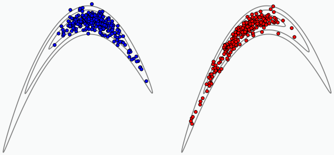
- Data $\boldsymbol{y} = (y_1, \ldots, y_n)$ where $n \to \infty$ (very large)
- Unknowns $\boldsymbol{\theta} \in \mathbb{R}^p$ where $p = \text{constant}$ (small or moderate)

- Good old Markov chain Monte Carlo (MCMC)?
  - ✓ Applicable (in principle)
  - × Too slow: computing a posterior value costs $O(n)$
    $\rightsquigarrow$ possible to calculate only few times



Big data scalable MCMC?

At least following 'scalable MCMC' have been suggested:



- Run MCMC targeting 'sub-posteriors' of batches[1]
  - ✓ Simple: just use standard MCMC for each compute node
  - × Combination of sub-posteriors relies on approximation
- Approximate accept/reject decision in Metropolis-Hastings[2]
  - ✓ Can have substantial speed-up of single iteration
  - × Tradeoff: accuracy, which is difficult to quantify

[1]Scott et al. (*Intern. J. Managem. Sci. Eng. Managem.*, 2016)
[2]Bardenet, Doucet & Holmes (*JMLR*, 2017); Korattikara, Chen & Welling (*PMLR*, 2014)

Unadjusted Langevin algorithm

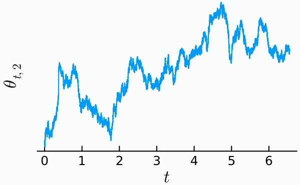
- Let $U(\boldsymbol{\theta}) = -\log\big(\mathrm{pr}(\boldsymbol{\theta})L(\boldsymbol{\theta};\boldsymbol{y})\big)$, so that

$$\boxed{\pi(\boldsymbol{\theta}) \propto e^{-U(\boldsymbol{\theta})}}$$

- The (overdamped) Langevin diffusion

$$\mathrm{d}\boldsymbol{\theta}_t = -\frac{1}{2}\nabla U(\boldsymbol{\theta}_t)\mathrm{d}t + \mathrm{d}\boldsymbol{B}_t$$

has π as stationary distribution (mild cond. on U)



Unadjusted Langevin algorithm

- Let $U(\boldsymbol{\theta}) = -\log\left(\mathrm{pr}(\boldsymbol{\theta})L(\boldsymbol{\theta};\boldsymbol{y})\right)$, so that

$$\boxed{\pi(\boldsymbol{\theta}) \propto e^{-U(\boldsymbol{\theta})}}$$

- The (overdamped) Langevin diffusion

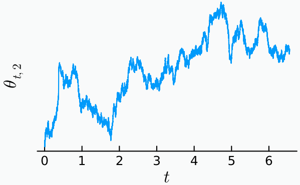
$$\mathrm{d}\boldsymbol{\theta}_t = -\frac{1}{2}\nabla U(\boldsymbol{\theta}_t)\mathrm{d}t + \mathrm{d}\boldsymbol{B}_t$$

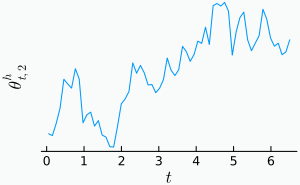
has π as stationary distribution (mild cond. on U)

$\rightsquigarrow$ Time-discretised Langevin process

$$\boldsymbol{\theta}^h_{t+h} = \boldsymbol{\theta}^h_t - \frac{h}{2}\nabla U(\boldsymbol{\theta}^h_t) + \sqrt{h}\boldsymbol{Z}, \qquad \boldsymbol{Z} \sim N(\mathbf{0}, \mathrm{I})$$

has stationary distribution $\pi_h \approx \pi$





Unadjusted Langevin with stochastic gradients

- Assuming i.i.d. data, $L(\boldsymbol{\theta}; \boldsymbol{y}) = \prod_{i=1}^{n} L_i(\boldsymbol{\theta}; y_i)$ and so

$$\nabla U(\boldsymbol{\theta}) = \nabla \log \mathrm{pr}(\boldsymbol{\theta}) + \sum_{i=1}^{n} \nabla \log L_i(\boldsymbol{\theta}; y_i)$$

- Let $I \subset \{1, \ldots, n\}$ be random with size $|I| = m$, then

$$G(\boldsymbol{\theta}) = \nabla \log \mathrm{pr}(\boldsymbol{\theta}) + \frac{n}{m} \sum_{i \in I} \nabla \log L_i(\boldsymbol{\theta}; y_i)$$

  satisfies $\mathbb{E}[G(\boldsymbol{\theta})] = \nabla U(\boldsymbol{\theta})$ and costs only $O(m)$
- Stochastic gradient Langevin dynamics (SGLD)[3]

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \frac{h_k}{2} G(\boldsymbol{\theta}_t) + \sqrt{h_k} \boldsymbol{Z}, \qquad \boldsymbol{Z} \sim N(\mathbf{0}, \mathrm{I})$$

  with suitably chosen $h_k \searrow 0$ samples from π...

[3] Welling & Teh (*ICML*, 2011); Nemeth & Fearnhead (*JASA*, 2021)

Unbiased gradients (cont.)

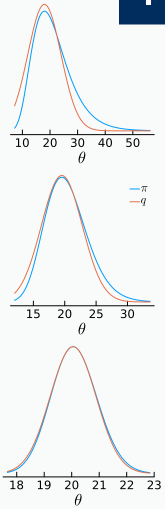
- SGLD convergence rate slower than with 'standard' MCMC
  - Can still be useful non-asymptotically
  - Further enhancements, such as control variates for $G(\boldsymbol{\theta})$...
- Recent wave of 'piecewise deterministic Markov processes' are continuous-time processes like Langevin, and based on gradients[4]
  - ✓ In principle, valid MCMC (provably target the posterior)
  - ✓ Gradients can potentially be replaced by unbiased estimators
  - × Difficult to implement in practice (further information about model required, or a difficult-to-quantify approximation error...)

[4] e.g.Bierkens, Fearnhead & Roberts (*Ann. Statist.*, 2019); Bouchard-Côté, Vollmer & Doucet (*JASA*, 2018).

Bernstein-von-Mises & good old Laplace approximation?

- Bernstein-von-Mises theorem for ‘well-identifiable’ models and large n:
  - Posterior nearly normal & asymptotically equivalent to maximum likelihood...

$\therefore$ Laplace approximation good for large n:
  - Find maximum-a-posteriori $\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \log \pi_u(\boldsymbol{\theta})$
  - Calculate Hessian H of $\log \pi_u(\boldsymbol{\theta})$
  - Approximate $\pi(\boldsymbol{\theta}) \approx q(\boldsymbol{\theta})$ where $q = N(\boldsymbol{\theta}^*, H^{-1})$
- The effect of prior vanishes as $n \to \infty$
  - Is Bayes really necessary? Stick with maximum likelihood?



Case 2: Refined model fidelity

A lot of unknowns but limited data

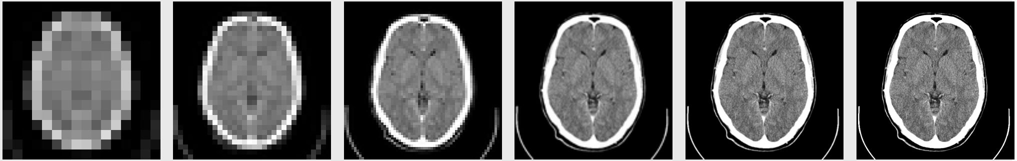
- Data $\boldsymbol{y} = (y_1, \ldots, y_n)$ where n constant (moderate)
- Unknowns $\boldsymbol{\theta} \in \mathbb{R}^p$ where $p \to \infty$ (very large)

A lot of unknowns but limited data

- Data $\boldsymbol{y} = (y_1, \ldots, y_n)$ where n constant (moderate)
- Unknowns $\boldsymbol{\theta} \in \mathbb{R}^p$ where $p \to \infty$ (very large)

Example: Computed tomography

$p = 16^2$ $p = 32^2$ $p = 64^2$ $p = 128^2$ $p = 256^2$ $p = 512^2$

4 

Typical properties

- Often refined discretisation of a continuous time/space model

$$\pi_p(\boldsymbol{\theta}) \propto \mathrm{pr}_p(\boldsymbol{\theta}) L_p(\boldsymbol{\theta}; \boldsymbol{y}), \qquad \mathrm{pr}_p \to \mathrm{pr}_\infty \;\&\; L_p \to L_\infty$$

- The likelihood does not concentrate as $p \to \infty$
  $\rightsquigarrow$ the effect of prior remains substantial as $p \to \infty$
  $\rightsquigarrow$ $\pi_p(\boldsymbol{\theta})$ can be clearly non-Gaussian

× Generic MCMCs break down in high dimension p

- Can be exponentially bad in p...

✓ The posterior does not *really* get 'more difficult' when p increases...

Pre-conditioned Crank-Nicolson (pCN) algorithm[5]

- Assume prior $\mathrm{pr}(\boldsymbol{\theta}) = N(\boldsymbol{\theta}; \mathbf{0}, \Sigma)$
- Ornstein-Uhlenbeck/AR(1) proposal:

$$\boldsymbol{\theta}' = (1-\epsilon^2)^{1/2}\boldsymbol{\theta} + \epsilon \boldsymbol{Z}, \qquad \boldsymbol{Z} \sim N(\mathbf{0}, \Sigma),$$

  where $\epsilon \in (0,1)$ is a tuning parameter
- Proposal $q(\boldsymbol{\theta}, \boldsymbol{\theta'})$ admits detailed balance for $\mathrm{pr}(\boldsymbol{\theta})$
- Metropolis acceptance probability

$$\min\left\{1, \frac{L(\boldsymbol{\theta}'; \boldsymbol{y})}{L(\boldsymbol{\theta}; \boldsymbol{y})}\right\}$$

- pCN is generalisation of Metropolis algorithm:
  - Metropolis: q symmetric $\iff$ reversible wrt. Lebesgue $\lambda(\mathrm{d}\boldsymbol{\theta})$
  - pCN: q reversible wrt. $\mu(\mathrm{d}\boldsymbol{\theta}) = \mathrm{pr}(\boldsymbol{\theta})\lambda(\mathrm{d}\boldsymbol{\theta})$

[5] Cotter, Roberts, Stuart & White (*Statist. Sci.*, 2013)

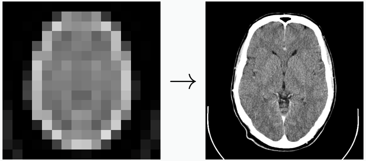
Post-correction of mismatch

- Theoretically, pCN can be valid even in ‘infinite dimension’
  - Practical effectiveness of pCN depends on how informative $L(\boldsymbol{\theta}; \boldsymbol{y})$ is
- If discrepancy is small enough, importance sampling can be sufficient, too
  - Draw $\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_m \sim \mathrm{pr}$
  - Estimate $\mathbb{E}_\pi[\phi(\Theta)] \approx \sum_{k=1}^m W_k \phi(\boldsymbol{\Theta}_k)$ where $W_k = \frac{L(\boldsymbol{\Theta}_k)}{\sum_{i=1}^m L(\boldsymbol{\Theta}_i)}$
  - This is practical only if $L(\,\cdot\,; \boldsymbol{y})$ is weakly informative

[6] e.g. V, Helske & Franks (*Scand. J. Statist.*, 2020)

Post-correction of mismatch

- Theoretically, pCN can be valid even in ‘infinite dimension’
  - Practical effectiveness of pCN depends on how informative $L(\boldsymbol{\theta}; \boldsymbol{y})$ is
- If discrepancy is small enough, importance sampling can be sufficient, too
  - Draw $\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_m \sim \mathrm{pr}$
  - Estimate $\mathbb{E}_\pi[\phi(\Theta)] \approx \sum_{k=1}^m W_k \phi(\boldsymbol{\Theta}_k)$ where $W_k = \frac{L(\boldsymbol{\Theta}_k)}{\sum_{i=1}^m L(\boldsymbol{\Theta}_i)}$
  - This is practical only if $L(\,\cdot\,; \boldsymbol{y})$ is weakly informative
- Also possible to balance between MCMC and IS[6]
  - MCMC inference for π_{p_0} with p_0 small(ish)
  - IS post-correct $\pi_{p_0} \to \pi_p$ where $p \gg p_0$ ‘fine enough’



[6] e.g. V, Helske & Franks (*Scand. J. Statist.*, 2020)

Case 3: Large model & a lot of data

The grand challenge: both p and n large

- The more data, the more challenging questions one can ask
  $\rightsquigarrow$ models with large p and n
- Example: Latent variable model with random effect for each datum $p = O(n)$
  $\rightsquigarrow$ substantial uncertainty about (some) unknowns
- In principle, Bayesian modelling and inference can be very useful!

Questions:

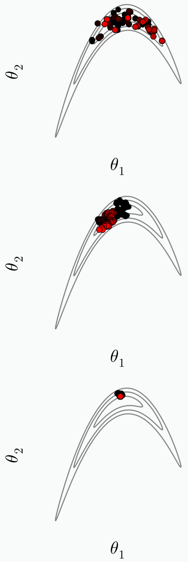
1. How far can we push ‘generic’ MCMC? Dimension scalability?
2. More model specific methods?

Dimension scalability of random-walk Metropolis

- Diffusion limits[a]: proposal variance $O(1/p) \rightsquigarrow$ mixing in $O(p)$ time
- Recent findings[b] consolidate this
  - $\frac{m}{2}\|\boldsymbol{h}\|^2 \leq U(\boldsymbol{\theta}+\boldsymbol{h}) - U(\boldsymbol{\theta}) - \boldsymbol{h}^T \nabla U(\boldsymbol{\theta}) \leq \frac{L}{2}\|\boldsymbol{h}\|^2$, condition number $\kappa = \frac{m}{L}$
  - e.g. $\pi = N(\mathbf{0}, \Sigma) \implies \kappa = \mathrm{cond}(\Sigma)$
  - Gaussian proposal with increments $\sigma^2 = \frac{1}{2}L^{-1}p^{-1}$
  - L_2 spectral gap $\geq C\kappa p^{-1}$ where C is universal

∴ Metropolis algorithm can scale reasonably well to regular targets

  - Assuming well-tuned proposal (and/or reparameterised target) so that κ small!



[a] Roberts, Gelman & Gilks (*Ann. Appl. Probab.*, 1997)
[b] Andrieu, Lee, Power & Wang (*arXiv*, 2022)

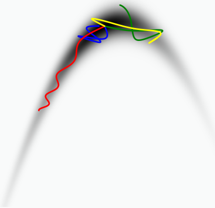
Hamiltonian Monte Carlo algorithms

- Hamiltonian Monte Carlo (HMC) proposals based on ODE

$$\dot{\boldsymbol{\theta}} = S^{-1}\boldsymbol{m} \qquad \boldsymbol{\theta}_0 = \boldsymbol{\theta}$$

$$\dot{\boldsymbol{m}} = -U(\boldsymbol{\theta}) \qquad \boldsymbol{m}_0 \sim N(\mathbf{0}, S)$$

  and $\boldsymbol{\theta}' = \boldsymbol{\theta}_t$ — leaves $\pi(\boldsymbol{\theta}) \propto e^{-U(\boldsymbol{\theta})}$ invariant



- In practice (e.g. Stan[a]):
  - Leapfrog time-discretisation h & Metropolis correction
  - Dynamic integration time: the no-U-turn sampler (NUTS)
  - Heuristics to choose step size h & mass S automatically
- Diffusion limit result[b]: $h = O(p^{-1/4}) \rightsquigarrow$ mixing in $O(p^{1/4})$
  - Not much precise theory, at least for dynamic HMC
  - HMC algorithms are notoriously sensitive to tuning...

[a] `https://mc-stan.org/`, Hoffman & Gelman (*JMLR*, 2014)
[b] Beskos, Pillai, Roberts, Sanz-Serna & Stuart (*Bernoulli*, 2013)

✓ Good old Gibbs sampling can be superior to HMC!

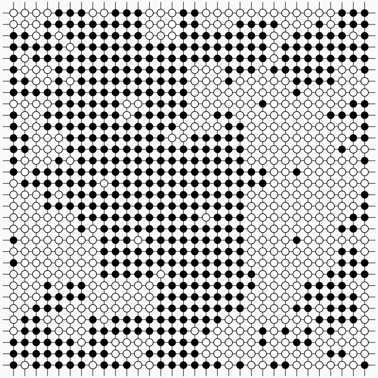
- Gibbs sampling used to be the practical MCMC (WinBUGS/OpenBUGS/JAGS)
  - ✓ Tuning-free algorithm
  - ✓ Can be scalable

Example: Random scan Gibbs on Ising model

$\boldsymbol{\theta} \in \{\pm 1\}^p$, $\mathrm{Ne} \subset \{1, \ldots, p\}^2$ set of neighbours and Δ maximal degree

$$\pi(\boldsymbol{\theta}) \propto \exp\left(\beta \sum_{i,j \in \mathrm{Ne}} \theta_i \theta_j\right)$$

If $\tanh(\Delta)\beta < 1$, then the mixing time is $O(p \log p)$.[7]



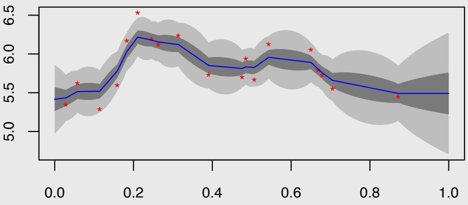
× Depends on the model — with strong dependencies, Gibbs can be very bad...

[7] Theorem 15.1 of Levin, Peres & Wilmer (2009)

Hidden Markov model (a.k.a. general state space model)

- Hidden Markov models have $p = dn$, and Markovian (sequential) structure
  - $L(\boldsymbol{\theta}; \boldsymbol{y}) = \prod_{k=1}^{n} G_k(\theta_k; y_k)$
  - $\mathrm{pr}(\boldsymbol{\theta}) = M_1(\theta_1) \prod_{k=2}^{n} M_k(\theta_{k-1}, \theta_k)$

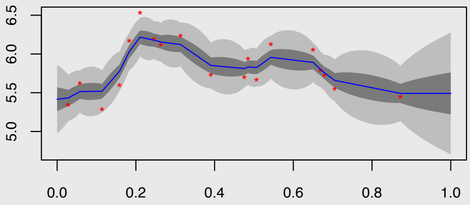
Example: Noisy observations of BM



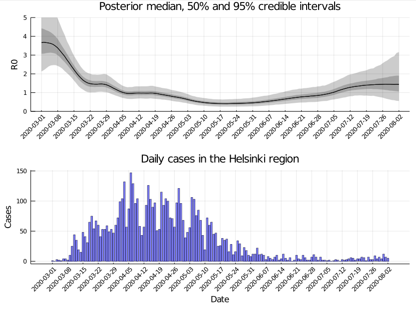
Hidden Markov model (a.k.a. general state space model)

- Hidden Markov models have $p = dn$, and Markovian (sequential) structure
  - $L(\boldsymbol{\theta}; \boldsymbol{y}) = \prod_{k=1}^{n} G_k(\theta_k; y_k)$
  - $\mathrm{pr}(\boldsymbol{\theta}) = M_1(\theta_1) \prod_{k=2}^{n} M_k(\theta_{k-1}, \theta_k)$

Example: Noisy observations of BM



Example: A stochastic SEIR



× Strong dependencies $\implies$ Gibbs and random-walk Metropolis bad

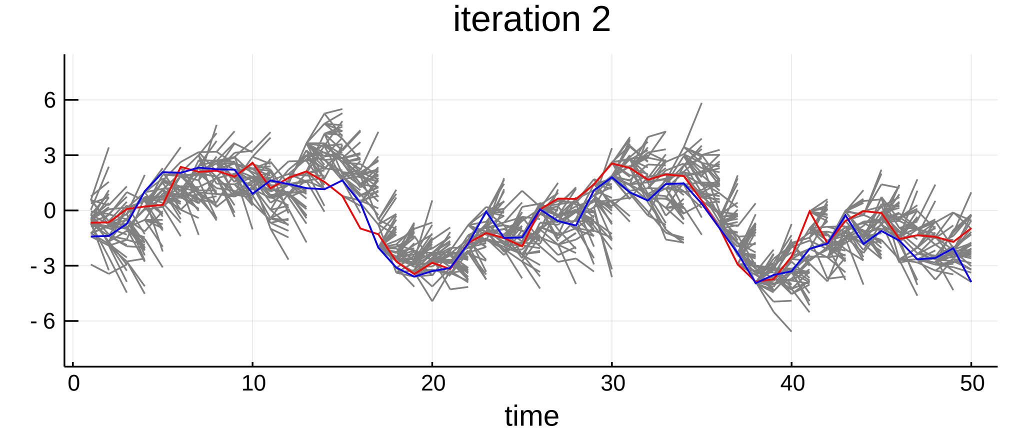
Conditional particle filter with backward sampling (CPF-BS)

Efficient and valid MCMC transition for HMMs $\theta^*_{1:T} \to (\Theta_1^{B_1}, \ldots, \Theta_n^{B_n})$[8]

1. Forward pass: 'tree search'
   - Place reference path $\theta^*_{1:T}$
   - Sample remaining $N-1$ auxiliary 'particles' sequentially from M_1, M_2, ..., M_n
   - Resample proportional to $G_1(\,\cdot\,)$, ..., $G_n(\,\cdot\,)$
2. Backward pass: 'selection'
   - Pick particle B_n at last time n
   - Backward sample $B_{n-1}, \ldots, B_1$
   - Output $(\Theta_1^{B_1}, \ldots, \Theta_n^{B_n})$

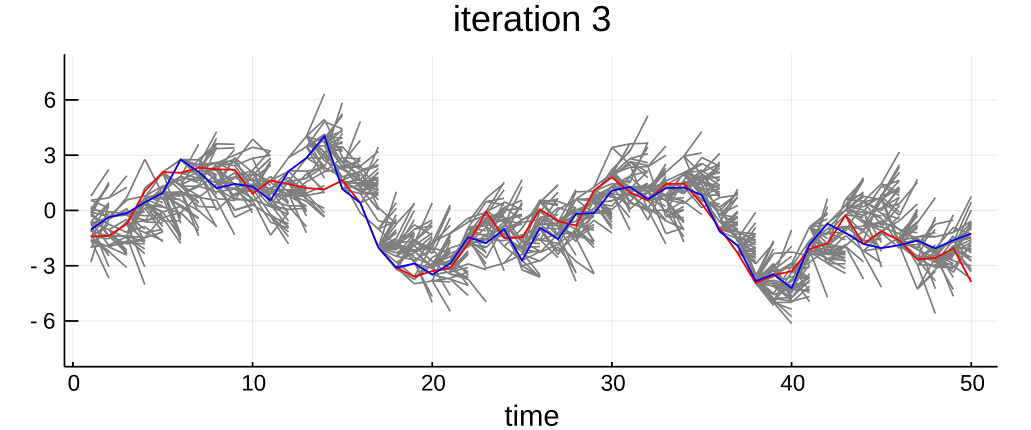
[8] Andrieu, Doucet & Holenstein; and Whiteley (*J. Roy. Statist. Soc. Ser. B.*, 2010); Algorithmic variant: Lindsten, Jordan & Schön (*J. Mach. Learn. Res.*, 2014) ancestor sampling CPF

Iterated CPF-BS on noisy AR(1)



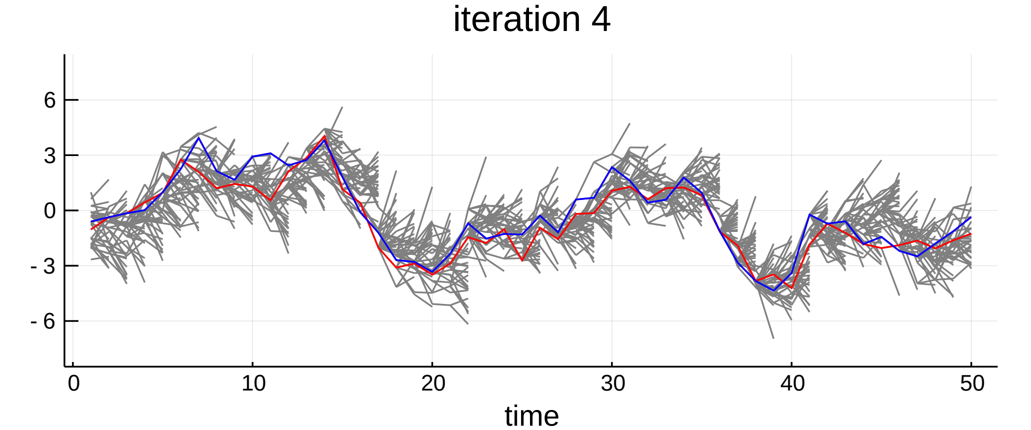
- Reference $\theta^*_{1:n}$, Output $\Theta^{B_{1:n}}_{1:n}$

Iterated CPF-BS on noisy AR(1)



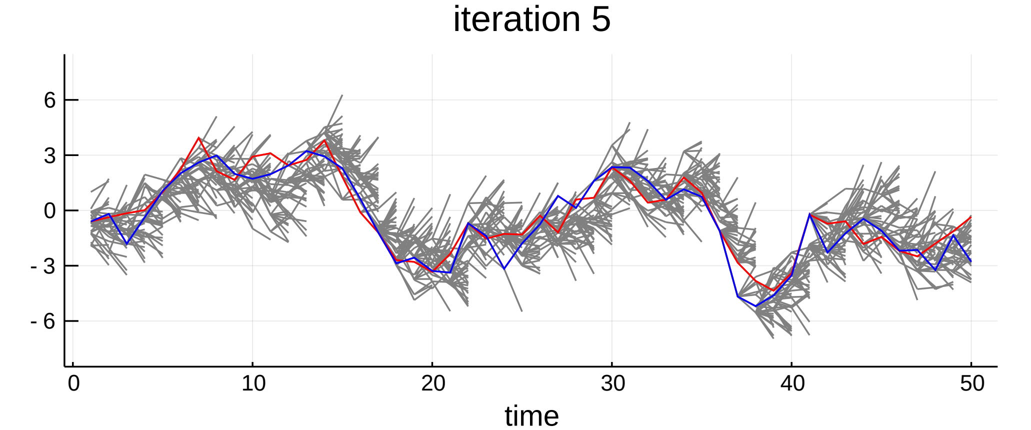
- Reference $\theta^*_{1:n}$, Output $\Theta^{B_{1:n}}_{1:n}$

Iterated CPF-BS on noisy AR(1)



- Reference $\theta^*_{1:n}$, Output $\Theta^{B_{1:n}}_{1:n}$

Iterated CPF-BS on noisy AR(1)



- Reference $\theta^*_{1:n}$, Output $\Theta^{B_{1:n}}_{1:n}$

CPF-BS is really useful!

- ✓ Can be used (almost) off-the-shelf (only N must be chosen)
- ✓ Has impressive empirical performance even with very large n
- ✓ Theoretical support, too:

Scalability result for CPF-BS[9]

If $G_* := \sup_k \|G_k\|_\infty < \infty$ and $M_* := \sup_k \frac{\|M_k(\,\cdot\,)\|_\infty}{\inf_{\theta,\theta'} M_k(\theta'|\theta)} < \infty$, then there exists $N_0 = N_0(G_*, M_*) < \infty$ such that for any $N \geq N_0$ and any $n \geq 1$ the mixing time satisfies

$$\tau_n \leq O(n)$$

- × Specific to suitable (sequential) models

[9] Lee, Singh & V (*Ann. Statist.*, 2020)

Discussion

Role of parallel and/or distributed computing

- MCMC is sequential
- Increase in computing power is increasing distributed and parallel
- Can use other than MCMC methods
- Or try to parallelise MCMC
  - Naive combination of short MCMC runs suffers from bias
  - Recent interest on unbiased MCMC estimators[10]
    $\rightsquigarrow$ run two coupled copies and wait until they meet
- Unbiased multilevel Monte Carlo for refined models[11]
  - Based estimates of differences of nested discretisations...

[10] Glynn & Rhee (*J. Appl. Probab.*, 2014); Jacob, O'Leary & Atchadé (*JRSS B*, 2020)
[11] Rhee & Glynn (*Oper. Res*, 2015); V (*Oper. Res.*, 2018)

Conclusions

- Scalability comes in different forms
    - Solutions are different, too
    - Many methods involve tuning $\rightsquigarrow$ scalable adaptation
- Features of the model can be useful in
    - Refined discretisations $\rightsquigarrow$ prior-informed moves / post-correction
    - Sequential structure $\rightsquigarrow$ particle MCMC

Conclusions

- Scalability comes in different forms
  - Solutions are different, too
  - Many methods involve tuning $\rightsquigarrow$ scalable adaptation
- Features of the model can be useful in
  - Refined discretisations $\rightsquigarrow$ prior-informed moves / post-correction
  - Sequential structure $\rightsquigarrow$ particle MCMC

🌈 Diversity in Bayesian computing!

  - There is no such thing as "state-of-the-art MCMC/Bayesian computing method"
  - SOTA depends on problem type
  - MCMC is not always slow!
  - (Variational) approximations can also be very useful!

Some references

- **Nemeth, C., & Fearnhead, P. (2021).**
  Stochastic gradient Markov chain Monte Carlo.
  JASA, 116(533), 433-450.
- **Bierkens, J., Fearnhead, P. & Roberts, G. (2019).**
  The Zig-Zag process and super-efficient sampling for Bayesian analysis of big data.
  Ann. Statist. 47(3): 1288-1320.
- **Cotter, S. L., Roberts, G. O., Stuart, A. M., & White, D. (2013).**
  MCMC methods for functions: modifying old algorithms to make them faster.
  Statist. Sci. 28(3): 424–446.
- **Vihola, M., Helske, J. & Franks, J. (2020).**
  Importance sampling type estimators based on approximate marginal Markov chain Monte Carlo.
  Scand. J. Statist., 47(4), 1339-1376.
- **Andrieu, C., Lee, A., Power, S. & Wang, A.Q. (2022).**
  Explicit convergence bounds for Metropolis Markov chains: isoperimetry, spectral gaps and profiles.
  Preprint *arXiv:2211.08959*.
- **Andrieu, C., Doucet, A. & Holenstein, R. (2010).**
  Particle Markov chain Monte Carlo methods.
  J. R. Stat. Soc. Ser. B. Stat. Methodol., 72(3), 269-342.
- **Lee, A., Singh, S. S. & Vihola, M. (2020).**
  Coupled conditional backward sampling particle filter.
  Ann. Statist., 48 (5), 3066-3089.
- **Glynn, P. W. & Rhee, C.-H. (2014).**
  Exact estimation for Markov chain equilibrium expectations.
  J. Appl. Probab., 51(A):377–389.

Some more references

- **Bardenet, R., Doucet, A., & Holmes, C. C. (2017).**
  On Markov chain Monte Carlo methods for tall data.
  J. Machin. Learn. Res., 18(47).
- **Scott, S. L., Blocker, A. W., Bonassi, F. V., Chipman, H. A., George, E. I., & McCulloch, R. E. (2016).**
  Bayes and big data: The consensus Monte Carlo algorithm.
  Intern. J. Managem. Sci. Eng. Managem., 11(2), 78-88.
- **Welling, M., & Teh, Y. W. (2011).**
  Bayesian learning via stochastic gradient Langevin dynamics.
  Proc. Mach. Learn. Res. (pp. 681–688).
- **Korattikara, A., Chen, Y., & Welling, M. (2014).**
  Austerity in MCMC land: Cutting the Metropolis-Hastings budget.
  Proc. Mach. Learn. Res.
- **Bouchard-Côté, A., Vollmer, S. J., & Doucet, A. (2018).**
  The bouncy particle sampler: A nonreversible rejection-free Markov chain Monte Carlo method.
  JASA, 113(522), 855-867.
- **Hoffman, M. D., & Gelman, A. (2014).**
  The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo.
  J. Mach. Learn. Res., 15(1), 1593-1623.
- **Wilmer, E. L., Levin, D. A., & Peres, Y. (2009).**
  Markov chains and mixing times.
  American Mathematical Soc., Providence.
- **N. Whiteley.**
  Discussion on "*Particle Markov chain Monte Carlo methods*".
  J. R. Stat. Soc. Ser. B. Stat. Methodol., 2010.
- **F. Lindsten, M. I. Jordan and T. B. Schön.**
  Particle Gibbs with ancestor sampling.
  J. Mach. Learn. Res., 2014.
- **Jacob, P. E., O'Leary, J., & Atchadé, Y. F. (2020).**
  Unbiased Markov chain Monte Carlo methods with couplings.
  J. R. Stat. Soc. Ser. B. Stat. Methodol. 82(3).
- **Rhee, C. H., & Glynn, P. W. (2015).**
  Unbiased estimation with square root convergence for SDE models.
  Oper. Res., 63(5), 1026-1043.
- **Vihola, M. (2018).**
  Unbiased estimators and multilevel Monte Carlo.
  Oper. Res.,66(2), 448-462.