Global Consensus Monte Carlo

https://doi.org/10.1080/10618600.2020.1811105

Lewis Rendell*, **Adam M. Johansen***,
Anthony Lee** and Nick Whiteley**

* University of Warwick; ** University of Bristol

e-mail: a.m.johansen@warwick.ac.uk
slides: https://go.warwick.ac.uk/amjohansen/talks/

JSM: August 10th, 2022

Outline

- Introduction and the Global Consensus Approach
- Global Consensus Markov chain Monte Carlo
- Global Consensus Sequential Monte Carlo
- Conclusions

Introduction

For problems involving large data sets, it may be convenient or necessary to distribute the data across multiple processors.

We consider a target probability density function given by

$$\pi(z) \propto \mu(z) \prod_{j=1}^{b} f_j(z)$$

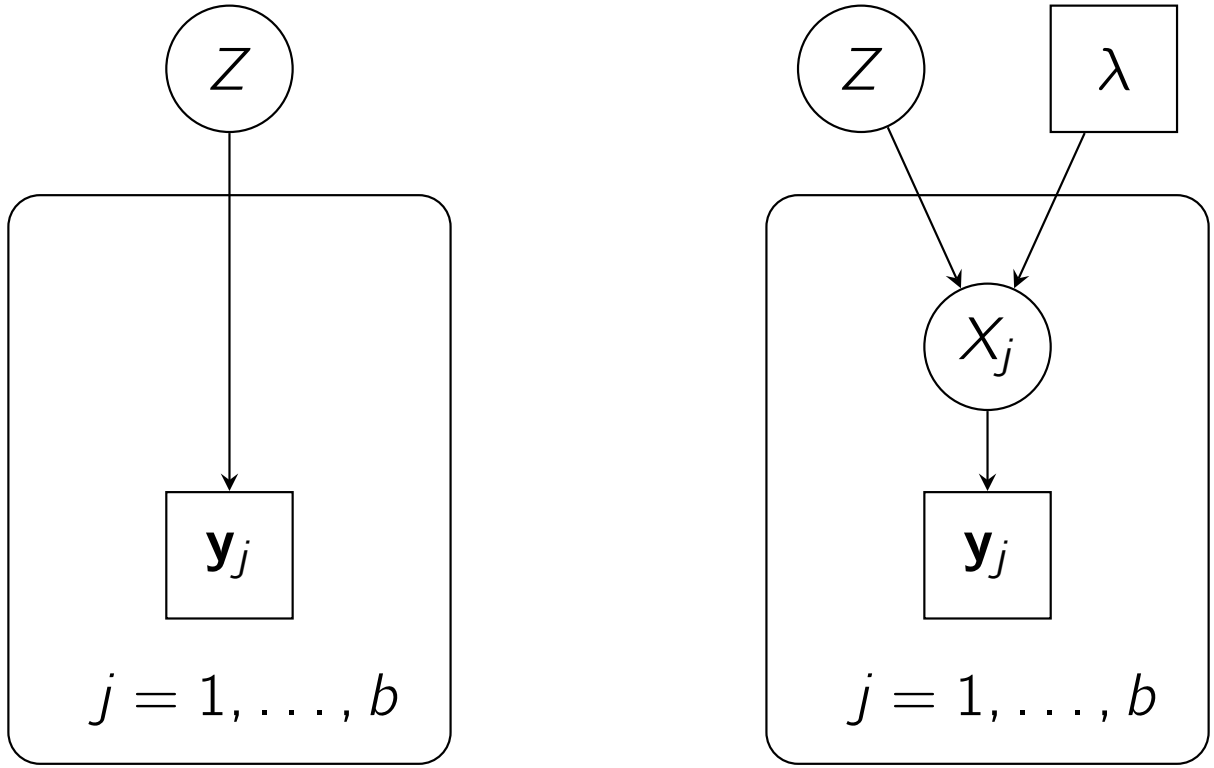
where f_j is computable on processor j, requiring consideration of $\mathbf{y}_j$, the jth subset of the full data set.

Earlier approaches

Target density: $\pi(z) \propto \mu(z) \prod_{j=1}^{b} f_j(z)$

- 'Embarrassingly parallel' approaches run b separate MCMC chains in parallel, followed by some final processing step.
  - Consensus Monte Carlo (Scott et al., 2016) requires chains with target densities proportional to $\mu(z)^{1/b} f_j(z)$. The samples are combined in a way that implicitly assumes approximate Gaussianity.
- Xu et al. (2014) employ expectation propagation, approximating each f_j by a density belonging to an exponential family.

An Instrumental Statistical Model



Introduce an instrumental hierarchical model (*above right*):

- Maintain the global variable z
- Introduce a top-level parameter λ
- Associate an instrumental variable x_j with each subset of the data — a local 'proxy' for the global variable
- Inspiration: (Global variable) Consensus Optimization (*not Consensus Monte Carlo*)

An instrumental model

Target density and Its Proxy

Target density: $$\pi(z) \propto \mu(z) \prod_{j=1}^{b} f_j(z)$$

Proxy: $$\tilde{\pi}_\lambda(z, x_{1:b}) \propto \mu(z) \prod_{j=1}^{b} K_j^\lambda(z, x_j) f_j(x_j)$$

We assume that f_j is bounded, and assume that this family satisfies $\int K_j^\lambda(z, x) f_j(x) \mathrm{d}x \to f_j(z)$ pointwise as $\lambda \to 0$.

Then the z-marginal of $\tilde{\pi}_\lambda$ converges in total variation to π, and so for bounded functions φ,

$$\int \varphi(z) \tilde{\pi}_\lambda(z) \mathrm{d}z \to \int \varphi(z) \pi(z) \mathrm{d}z.$$

MCMC algorithm

For given λ, a $\tilde{\pi}_\lambda$-reversible Markov chain is obtained via

Full conditional densities

$$\tilde{\pi}_\lambda(z \mid x_{1:b}) \propto \mu(z) \prod_{j=1}^{b} K_j^\lambda(z, x_j),$$

$$\tilde{\pi}_\lambda(x_j \mid z) \propto K_j^\lambda(z, x_j) f_j(x_j).$$

A two-variable Gibbs sampler may be constructed, where the two variables are z and $x_{1:b}$: providing approximations of $\int \varphi(z)\tilde{\pi}_\lambda(z)\mathrm{d}z$.

Same construction proposed for a different purpose in contemporaneous work by Vono et al. (2019).

GCMC MCMC Algorithm

Fix $\lambda > 0$. Set initial state $(Z^0, X^0_{1:b})$; choose chain length N.

For $i = 1, \ldots, N$:

- For $j \in \{1, \ldots, b\}$, sample $X^i_j \sim P^{(\lambda)}_{j, Z^{i-1}}(X^{i-1}_j, \cdot)$.
- Sample $Z^i \sim P^{(\lambda)}_{X^i_{1:b}}(Z^{i-1}, \cdot)$.

Return $(Z^i, X^i_{1:b})_{i=1}^N$.

Where:

- $P^{(\lambda)}_{j, Z^{i-1}}(X^{i-1}_j, \cdot)$ is $\tilde{\pi}_\lambda(x_j | Z^{i-1})$-invariant;
- $P^{(\lambda)}_{X^i_{1:b}}(Z^{i-1}, \cdot)$ is $\tilde{\pi}_\lambda(z \mid X^i_{1:b})$-invariant.

In practice Metropolis-within-Gibbs may be used: allows for architecture-based tuning.

The regularisation parameter λ

In practice, λ takes the role of a tuning parameter.

- λ too large
  $\Rightarrow \tilde{\pi}_\lambda(z)$ may form a poor approximation of $\pi(z)$
  $\Rightarrow$ estimators have a high bias.
- λ too small
  $\Rightarrow$ Markov chains may have high auto-correlation, poor mixing
  $\Rightarrow$ estimators have a high variance.

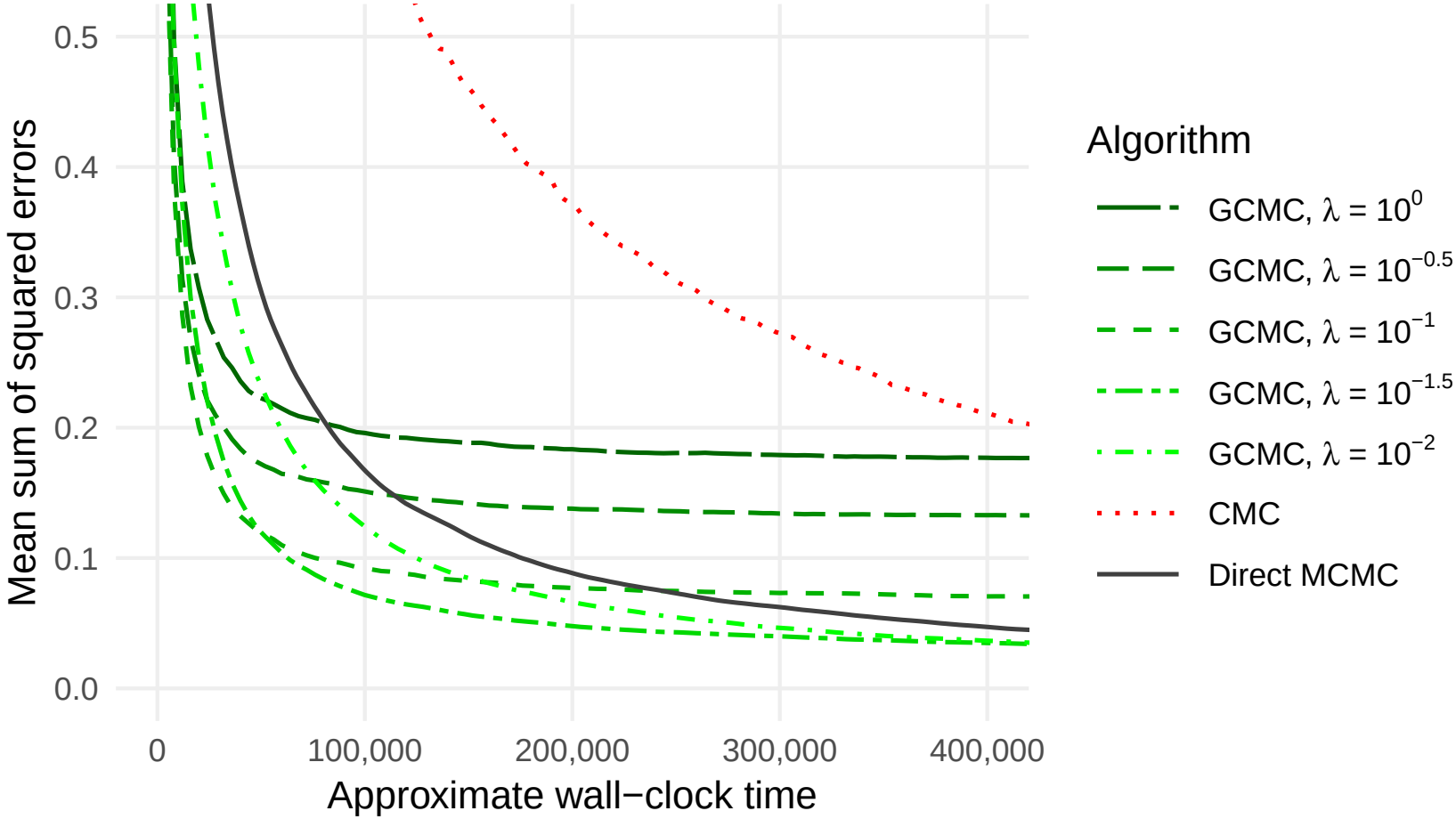
Choose λ to balance these, in a **bias–variance trade-off**.

An Example: Logistic regression

Data set formed of responses $\eta_i \in \{-1, 1\}$ and vectors $\xi_i \in \mathbb{R}^{20}$ of centred binary covariates.

- $d = 211$ coefficients:
  intercept $+20$ effect terms $+\binom{20}{2} = 190$ interaction terms.
- The $n = 80,000$ data are split into $b = 8$ subsets;
  $f_j(z) = \prod_i \sigma(\eta_i z^{\mathsf{T}} \xi_i)$, where the product is taken over those indices i included in the jth data subset, and σ is the logistic function.
- Prior: $\mu \sim \mathcal{N}(0, 20^2 I)$.
- For GCMC, we use normal transition kernels:
  $K_j^{\lambda}(z, x) = \mathcal{N}(x; z, \lambda I)$.
- MCMC steps: Z Gibbs Sampler
  X 20 iterates of random walk Metropolis.

Logistic regression MSE



- MSSE over all d components of poterior mean estimates.
- Idealised abstraction in which we assume latency is $10\times$ partial-likelihood evaluation time. Time is relative to the time taken to compute a single partial likelihood term.
- All values computed over 25 replicates.

SMC sampler (cf. Del Moral et al. (2006))

Instead of using the distribution $\tilde{\pi}_\lambda$ corresponding to a single λ value, use a sequence of such distributions: $\pi_{\lambda_0}, \pi_{\lambda_1}, \ldots, \pi_{\lambda_n}$.

To approximate such a sequence, use an SMC sampler:

- At time $p = 0$:
  - Draw N particles from π_{λ_0}
- At time $p = 1, \ldots, n$:
  - Importance weight the particles to target π_{λ_p}
  - Resample the particles (if necessary)
  - Apply a Markov kernel invariant with respect to π_{λ_p}

Such procedures can result in better approximations of each π_{λ_p} than would be obtained by a single π_{λ_p}-invariant MCMC chain.

Bias correction

Suppose we wish to estimate $\int \varphi(z)\pi(z)\mathrm{d}z$.
Recall that $\int \varphi(z)\tilde{\pi}_\lambda(z)\mathrm{d}z$ converges to this integral as $\lambda \to 0$.

Using output of SMC sampler, for some decreasing sequence $\lambda_0, \lambda_1, \ldots, \lambda_n$ we obtain estimates of

$$\int \varphi(z)\pi_{\lambda_0}(z)\mathrm{d}z, \int \varphi(z)\pi_{\lambda_1}(z)\mathrm{d}z, \ldots, \int \varphi(z)\pi_{\lambda_n}(z)\mathrm{d}z.$$

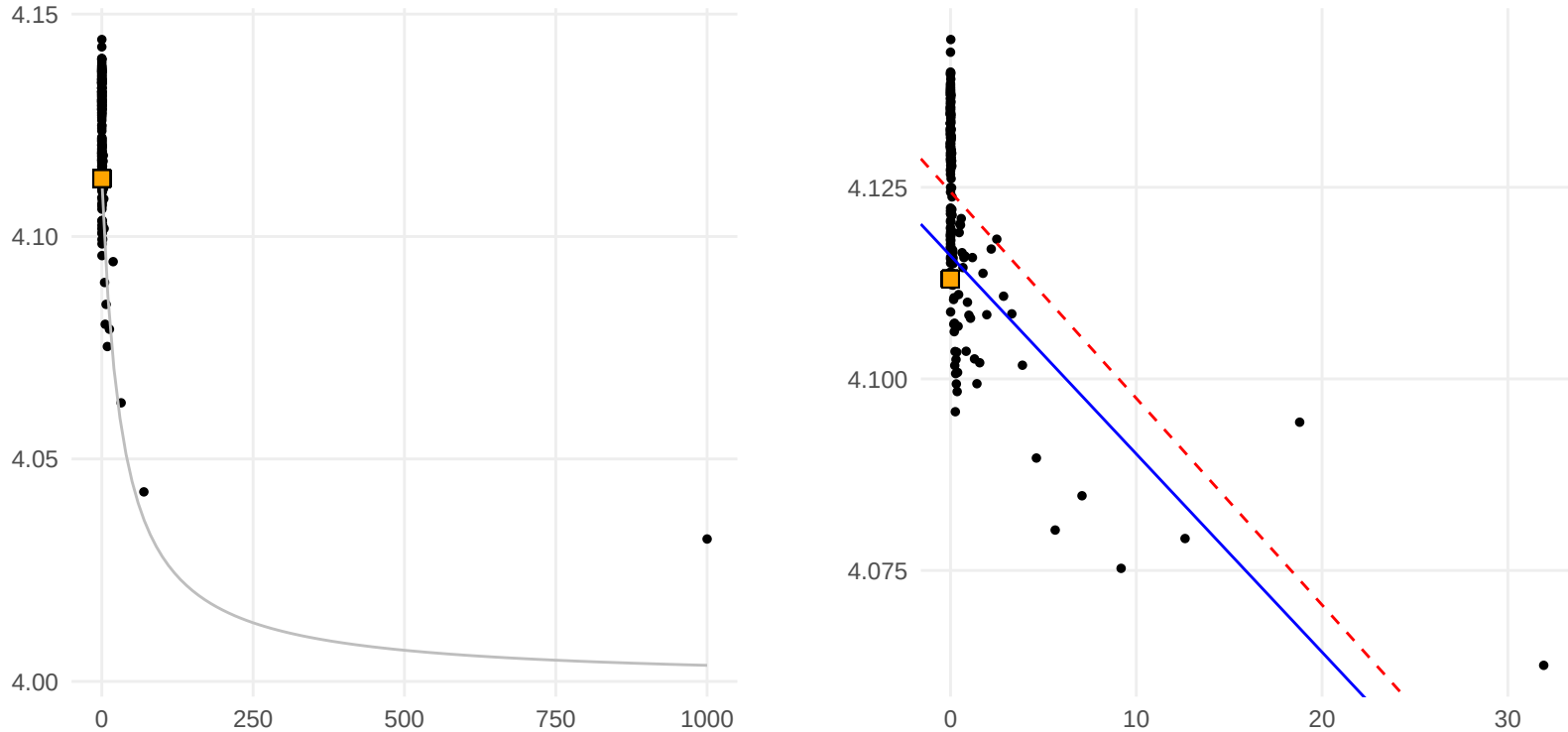
Idea: regress these estimates on λ, to obtain a bias-corrected estimate of the desired integral. We suggest local linear regression using weighted least squares.

An automated procedure

- Initialise the SMC sampler at some large value of λ; use an adaptive procedure (e.g. Zhou et al. (2016) [in JCGS]) to determine each successive λ value in a decreasing sequence.
- At each stage, compute an estimate of $\int \varphi(z)\tilde{\pi}_\lambda(z)\mathrm{d}z$, and estimate the variance of this estimate (using e.g. Lee and Whiteley (2018)).
- Adaptively determine a subset of these estimates for which λ is small enough that the dependence on λ is approximately linear.
- Use weighted least squares on this subset, extrapolating to obtain a bias-corrected estimate at $\lambda = 0$.

A Gaussian toy example

Gaussian prior density, Gaussian likelihood contributions.
We look to estimate $\int z\pi(z)\mathrm{d}z \approx 4.113$ (orange square).



LHS: Estimates and the true value vs. λ (solid grey).
RHS: Estimates used in regression vs. λ. Weighted (solid blue) and unweighted (dashed red) regression lines.

Returning to Logistic regression

- We look to estimate $\int z\pi(z)\mathrm{d}z \in \mathbb{R}^{211}$. We aim to minimise the sum of the mean squared errors of the posterior mean estimate of each component.
- For the MCMC approach (with a single value of λ), the smallest such 'total MSE' obtained was 0.0478 (for $\lambda = 10^{-1.5}$), though this was sensitive to the choice of λ.
- A comparable value of 0.0367 was obtained by the bias-corrected estimate obtained from SMC, at similar computational cost, while avoiding the difficulty in specifying a single λ value.
- Further improvements described in paper.

Conclusion

- Framework for sampling in distributed settings.
  - Pro: few distributional assumptions.
  - Pro: An automated SMC approach to tuning parameter specification.
  - Con: Requires more regular communication between computing nodes than some competitors.
  - Pro: Very amenable to incorporation of node-level random effects (Rendell, 2020, Section 7.5)
  - Local linear regression suggestion for bias correction is simple; other approaches are also possible.

- Another approach to distributed SMC first proposed by Lindsten et al. (2017) [in JCGS], with a theoretical analysis in Kuntz et al. (2021), has been used to unify inferences exactly with few distributional assumptions (Chan et al., 2021).

References

R. Chan, M. Pollock, A. M. Johansen, and G. O. Roberts. Divide-and-conquer Monte Carlo fusion. e-print 2110.07265, arXiv, 2021.

P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society B*, 63(3):411–436, 2006. doi:10.1111/j.1467-9868.2006.00553.x.

J. Kuntz, F. R. Crucinio, and A. M. Johansen. Divide-and-conquer sequential Monte Carlo: Properties and limit theorems. e-print 2110.15782, arXiv, 2021.

A. Lee and N. Whiteley. Variance estimation in the particle filter. *Biometrika*, 105(1):609–625, 2018.

F. Lindsten, A. M. Johansen, C. A. Naesseth, B. Kirkpatrick, T. Schön, J. A. D. Aston, and A. Bouchard-Côté. Divide and conquer with sequential Monte Carlo samplers. *Journal of Computational and Graphical Statistics*, 26(2):445–458, 2017. doi:10.1080/10618600.2016.1237363.

L. J. Rendell. *Sequential Monte Carlo variance estimators and global consensus*. Ph.D. thesis, University of Warwick, 2020. URL `http://webcat.warwick.ac.uk/record=b3684525~S15`.

L. J. Rendell, A. M. Johansen, A. Lee, and N. Whiteley. Global consensus Monte Carlo. *Journal of Computational and Graphical Statistics*, 30(2):249–259, 2021. doi:10.1080/10618600.2020.1811105.

S. L. Scott, A. W. Blocker, F. V. Bonassi, H. A. Chipman, E. I. George, and R. E. McCulloch. Bayes and big data: The consensus Monte Carlo algorithm. *International Journal of Management Science and Engineering Management*, 11(2):78–88, 2016. doi:10.1080/17509653.2016.1142191.

M. Vono, N. Dobigeon, and P. Chainais. Split-and-augmented Gibbs sampler — application to large-scale inference problems. *IEEE Transactions on Signal Processing*, 67:1648–1661, 2019. doi:10.1109/TSP.2019.2894825.

M. Xu, B. Lakshminarayanan, Y.-W. Teh, J. Zhu, and B. Zhang. Distributed Bayesian posterior sampling via moment sharing. In *Advances in Neural Information Processing Systems*, pages 3356–3364, 2014. URL `https://proceedings.neurips.cc/paper/2014/hash/a941493eeea57ede8214fd77d41806bc-Abstract.html`.

Y. Zhou, A. M. Johansen, and J. A. D. Aston. Towards automatic model comparison: An adaptive sequential Monte Carlo approach. *Journal of Computational and Graphical Statistics*, 25(3):701–726, 2016. doi:10.1080/10618600.2015.1060885.

The authors gratefully acknowledge the support of The Alan Turing Institute under the EPSRC grant EP/N510129/1, the Lloyd's Register Foundation–Alan Turing Institute Programme on Data-Centric Engineering, and the EPSRC under grants EP/M508184/1, EP/R034710/1 and EP/T004134/1.