Geosci. Model Dev. Discuss.,
https://doi.org/10.5194/gmd-2020-350-RC1, 2020

Geoscientific Model Development Discussions

Interactive comment on "Efficient Bayesian inference for large chaotic dynamical systems" *by* Sebastian Springer et al.

Anonymous Referee #1

Received and published: 21 December 2020

*** General comments ***

This paper combines and applies two other previously published methods to enable robust Bayesian inference of parameters for chaotic dynamical systems via Markov chain Monte Carlo (MCMC): the correlation integral likelihood (CIL) makes probabilistic inference on chaotic systems reliable, while local approximation MCMC (LA-MCMC) makes it efficient. This opens up MCMC sampling methods to work on numerically expensive chaotic models such as climate models, for which they would otherwise remain intractable.

The use of emulators as stand-ins for computationally expensive models under MCMC is a well-established area. The use of Bayesian techniques on chaotic systems seems to be less well trod, but potentially very high impact as our understanding of these

systems is pivotal to our understanding of the environment and to decision making for resource management and environmental stewardship. The correlation integral likelihood has received relatively little attention compared to synthetic likelihood approaches such as Wood 2010, but the computational efficiency gains for this type of system are clearly demonstrated. The value of the paper thus lies in a proof of principle for applying these techniques together to solve more challenging problems. The CIL might be enough on its own to handle the Lorentz attractor and simple population dynamics models, and LA-MCMC has been demonstrated on inversions involving partial differential equations. For complex nonlinear PDEs such as those describing weather systems, both are probably needed.

Since this paper presents applications rather than all-new methods, it makes sense to submit to an earth science journal rather than an applied statistics journal. Chaotic systems such as weather and climate models are within scope for GMD, and Bayesian inference could be viewed as a means for comparing models to data. The results are general and don't focus on any single geoscientific forward model, though the CIL is a likelihood and thus is an important part of a *statistical* model for the data. If all the forward models were coded with a specific library or package, or if the complete method were coded in a single forward-model-agnostic library, it would be natural to include the code name and version number in the title, but that doesn't seem to be the case; I'll defer to the editor's judgment in this matter.

The paper is overall well-structured and clearly and concisely demonstrates the practical application of the methodology. Model setups are clearly described. The first two experiments are useful to demonstrate that LA-MCMC recovers the true posterior for chaotic systems using CIL, and the reduction of two years' worth of conventional MCMC to three hours' walltime with the quasi-geostrophic weather model is impressive. The description of CIL is detailed and would enable anyone to code the metric for themselves, and the authors cite a good representative sample of competing methods for CIL, and convincingly demonstrate that their method gives sensible results.

I found the paper interesting and informative and believe it merits publication. The main class of clarifications I would like to see concern how various choices were made for the setup and parameters of the CIL and LA-MCMC algorithms used to attack these problems, which will be of interest to any practitioners who want to adopt this method for their problems.

*** Specific comments ***

– Section 2.2 (Local Approximation MCMC)

The discussion of LA-MCMC is shorter and more schematic than the discussion of CIL in the preceding subsection, and relies heavily on citations of Davis et al 2020. Repeating the entire algorithm setup may be overkill, but this level of description is probably not enough for the user to understand and reproduce how the algorithm works, in contrast to the CIL section. There is some discussion about the benefits of using emulators in general, where I think it's appropriate to cite some key papers (e.g. Sacks et al 1989; Kennedy & O'Hagan 2001) introducing the idea of emulators for model calibration and efficiency in Bayesian inference. The specific advantage of LA-MCMC as a meta-method – not the fact of being an emulator method, nor its functional form, but the theoretical guarantee of optimal convergence as the number of samples and forward model evaluations increases – is reasonably well made here. Some discussion about the local support of the approximation and refinement criteria, as in Conrad et al 2016, would also be welcome in this subsection.

Why were the particular parameters to LA-MCMC chosen here? Were there any problem-specific considerations, or are these reasonable default settings? For example, was the quadratic approximation chosen for any particular reason over either linear or Gaussian-Process-based local approximations? How would different choices impact the performance?

What Metropolis-Hastings proposal density is actually used for these problems and how was it tuned? The algorithm as outlined in Davis et al 2020 is a meta-method

that specifies only that the proposal density should be constant in time, which seems on its face to rule out many adaptive and gradient-based proposals. The number of parameters being inferred is always small in this paper, so a Gaussian random walk might work fine, but if so please state that.

– Section 3 (Numerical Experiments)

What considerations are used to pick the number M of correlation scale bins? Clear criteria are given for R_0 and R_M, and b is fixed once these parameters and M are known. The specific choices of M for each problem are given in the text, but not why they are optimal or even appropriate. How will performance of the CIL vary if M is suboptimal?

Similarly, how is the observation interval Delta_t chosen? It seems in all application cases to be deliberately longer than the "predictable" interval, presumably to ensure that adjacent observations yield new information and have settled into the attractor geometry. This becomes important specifically when simulating many short chaotic trajectories rather than one long one (to parallelize the problem and achieve desired efficiency gains). These points are alluded to in the subsequent subsections for individual experiments, but they are central enough points to bring up or reiterate alongside other general information in the opening part of the section.

This would also be a good place to talk about the computing architecture used for these experiments, since it's hard to see what level of computational effort this wall time signifies otherwise. (Mention of a NVIDIA 1070 GPU is made once, and only for the K-S model, though comments about the degree of parallelization are mentioned for each problem.) This is more for completeness given that the choice of processor won't make a factor of 100 difference to the results, but practitioners may want to know this. If any particular libraries were used e.g. fast solvers or integrators, those dependencies should also be made clear.

– Section 3.2 (Quasi-geostrophic model)

For readers unfamiliar with this model, are H1 and H2 only mean thicknesses, or is this more of a 2.5-dimensional model where H1 and H2 are spatially constant thicknesses of two interacting vertical zones?

What was the value of f0 used? How do the beta coefficients depend upon it (dependencies of all other parameters on f0 are given in the text)? Was f0 inferred alongside H1 and H2? If not, would sampling over it be just as straightforward as sampling over H1 and H2, or would it be likely to create new challenges due to its correlations with several other parameters of the problem – as for example sampling in hierarchical Bayesian models where a global variance scale can induce strong and inconvenient posterior curvature?

– Section 4 (Conclusions)

Are there any specific inference problems currently faced by the weather and climate modeling community that might not have been previously feasible, but could perhaps now be attempted with this method?

– Supplementary Material

The authors have expressed their wish to include the code for these examples as supplementary material. Ideally, as a reader I'd prefer to see a Git repository where the code can be downloaded and installed at will, tagged with a given version, and given an external persistent DOI, even if it is not meant to be maintained as a general open-source package for the community. If the authors are going to upload the code as a supplement, it becomes much more important to clearly indicate all environment parameters and code dependencies, and to provide the supplement in a form that can simply be downloaded into a directory and run by the user to reproduce their key results.

*** Technical comments ***

Fig 1: The layout of this figure could be expanded to make it more readable. The font

for the labels seems quite small, and the vertical arrangement of the graphs is crowded. All content is appropriate, though.

Figs 2, 4, 6, 8, 9: Similarly expand label fonts a bit for easier reading, and adjust tick spacing if necessary to prevent overlap.

Fig 3: There isn't much information in this figure and I believe it could be safely omitted. Using a linear y-axis makes the effort needed for full MCMC look dramatic compared to LA-MCMC, but this point is adequately made in the text. The figure would make more sense if the paper were about comparing the efficiencies of different emulator methods.

Fig 7: This is a useful cartoon; I'd like to see some of the other geometric parameters of the system included on this figure, if possible, given that the actual version solved becomes non-dimensional.

Interactive comment on Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2020-350, 2020.