

Interactive comment on “Precision Annealing Monte Carlo Methods for Statistical Data Assimilation: Metropolis-Hastings Procedures” by Adrian S. et al.

Adrian S. et al.

asw012@ucsd.edu

Received and published: 17 May 2019

L18:

What is the difference in definition between data assimilation and statistical data assimilation?

They are the same.

C1

L 105:

Perhaps it is simply semantics, but I interpret equation (4) as being built from propagating forward in time from time t_0 to t_F . Is there a reason for the interpretation of moving backwards in time?

Forward or backward here are both equivalent in this formulation. Remember that time is not sequential in this formulation, but (in the language of graphs) is more like a vertex between two nodes on an un-directed graph.

L 115:

Maybe I misinterpreted the notation, but it seemed based on equation (2) that $x(n+1)$ and $f(x(n), p)$ were equal. If this is supposed to represent a measure of model error, it would help if this was explained in more detail.

If we are using the perfect state, then $x(n+1) = f(x(n), p)$. However, we propose that the **estimated** state evolve as $x(n+1) = f(x(n), p) + N(0, R_f^{-1})$, where N is the normal distribution. We will add this in the paper.

Equation (7):

Since the argument to the action A is $\cap X$, and this doesn't appear in the right hand side, it is a bit confusing at first sight. Could the authors make the relationship more clear. Again, it would help if somewhere it is clarified how the terms $x(n+1)$ and $f(x(n), p)$ differ. Which is the known quantity and which is the control variable

C2

corresponding to X ? I'd like a little more explanation of the quantities in the second term and where they come from.

We have restated/reiterated X for clarity.

L 184-186:

it should be acknowledged that this will encounter problems if the model has significant systemic errors relative to the 'true' system used to generate the observed data. (The systemic model errors may be more severe than a mis-specified model parameter, e.g. a process that was oversimplified in the formation of the model).

We expect there should be some limitations depending on the number of observations that are used. For example, with $D = 5$ and $L = 1$, there is still a significant portion of the state estimate that will be unconstrained when $R_f = 0$, and could lead to a quite poor starting condition.

Also, if the artificial model is completely unusable, in the sense that it produces exponentially large errors, then this method will inevitably fail, as would any method. But as long as the error are bounded in the L_2 sense, then the method should work. That is not to say that it will predict well, or that the results will look appealing, but it is the best one could do with the given (faulty) model and measurements.

C3

L 217-218:

"However, we substitute for $x_l(t_0 + k[n\tau])$, whenever it occurs in the equations Eq. (9), ..." I assume then that you're relying on a degree of synchronization so that the unobserved variables become dynamically consistent with the observed variables. With respect to this process, I have two questions: (1) how long does it take for the unobserved variables to reach a dynamic equilibrium with the observed variables so the full states are balanced. Is this performed before the DA analysis window $[t_0, t_F]$, or within this window? If the latter, is sufficient time given in the ensemble generation phase to achieve sufficient 'spin up'? (2) Assuming there is noise in the observed variables, this could potentially produce states that are not dynamically consistent. What is the sensitivity of this procedure to increasing noise in the observations?

1) We don't know how much time it takes for the synchronization to occur, but this is missing the point. We just generate these paths within the window $[t_0, t_F]$, and they may or may not be 'good' relative to the true solution, but almost always 'good' compared to a random initial guess. We are simply trying to be a bit better than random. 2) This is a very interesting question, and one that we are aware of and will look into deeper. We have tried adding a generous amount of noise. So far, this method seems pretty resilient to noise. This method of initialization will be tackled in much greater detail in a separate paper.

Equation (10):

This looks like an ensemble mean of the N_A accepted paths for each of the N_I initial paths. From the description in lines 227-230, it's not clear to me how these paths are generated.

C4

We use standard Metropolis-Hastings Monte Carlo sampling to generate the N_A paths. We will defer the full M-H algorithm to some good review papers.

L 242-245:

Could you clarify how many integrations of the full nonlinear model are required at each iteration of increasing R_f values? Is it one per each of the N_I ensemble members?

There are a total of M^2 full model integration steps.

L 282-283:

It seems like you will eventually reach a point of 'overfitting' a model with systematic errors (e.g. slightly incorrect fundamental equations, not just parameter errors). Is there a stopping criterion to avoid overfitting? Going back to my earlier question, I wonder if there is an analogous part of the algorithm to the 'validation' phase of the deep learning machine learning process that could help identify overfitting.

In fact, the point of the algorithm is to "over-fit" the model to some extent. However, this is not quite the same as overfitting or memorizing data. Moreover, if the fundamental equations are wrong, then most methods we are aware of will fail too. We can always stop the algorithm as soon as $\frac{dA}{dR_f} = 0$, so that R_f does not get too large. From our experiments, we have not seen evidence/consequence for anything resembling as

C5

overfitting. Hopefully, this addressed the concern of overfitting.

In the machine learning language, the estimation window is equivalent to the test set. And it seems, conceptually, that the prediction window is the equivalent of the validation set. As far as we can infer, there is no test set equivalent, nor is there a good reason for one.

L 293:

What is the length of the time window? (i.e. what is t_0 and t_F ?)

t_0 is 0, and t_F is 5, given $\Delta t = 0.025$. (We should be adding this to the manuscript somewhere).

Figure (2):

I wonder if you can limit the ensemble size N_I equal to the number of positive (+neutral) Lyapunov exponents, similarly to the minimum required ensemble size for the EnKF. Have you tried reducing the ensemble size N_I and testing the sensitivity of convergence to this size? For reference, see Bocquet and Carrasi 2017: <https://www.tandfonline.com/doi/full/10.1080/16000870.2017.1304504> I assume this figure illustrates that the system cannot be observed with as few as 5 observed model grid points per analysis time step. It might be worth clarifying the authors' interpretation in the figure caption. Could you describe which are the observed variables, as in the Figure (3) caption?

C6

We have not tried anything like what you have referenced. We don't know the answer to this question, since we don't know if this is comparable to the ensemble size for the EnKF.

Figure (6):

$dt = 5.0$ for the L96 model likely has pretty nonlinear error growth. (1) An estimate of the error growth over this window (e.g. the FTLEs) could be useful to set the context for how you might expect errors to grow during the forecast period. (2) With a systematic model error this might experience even greater sensitivity. I wonder if the authors could attempt a similar experiment with a shorter time window (e.g. with more approximately linear error growth), but cycle the process over multiple time windows like a realistic forecasting application.

We do not quite understand this comment and are unable to respond to it, at this time, without further clarification.

Interactive comment on Nonlin. Processes Geophys. Discuss., <https://doi.org/10.5194/npg-2019-1>, 2019.