Hydrol. Earth Syst. Sci. Discuss., 8, C2191–C2194, 2011
www.hydrol-earth-syst-sci-discuss.net/8/C2191/2011/

Hydrology and Earth System Sciences Discussions

Interactive comment on "DREAM$_{(\mathrm{D})}$: an adaptive markov chain monte carlo simulation algorithm to solve discrete, noncontinuous, posterior parameter estimation problems" *by* J. A. Vrugt

J. Vrugt

jasper@uci.edu

Received and published: 15 June 2011

DREAM$_{(\mathrm{D})}$: an adaptive markov chain monte carlo simulation algorithm to solve discrete, noncontinuous, posterior parameter estimation problems

I highly appreciate the constructive and useful feedback of the reviewers. Their suggestions and comments will certainly be beneficial to further improve this paper.

Let me provide some general responses to the comments of the reviewers.

In the past few years, I have received a few emails from users of the DREAM / AMAL-

GAM software with questions how to use these methods to solve non-continuous (integer) optimization problems. I have responded to these emails and questions by giving some ideas on how to modify the codes to accomplish this. A few months ago, I finally decided to do this implementation myself. I was curious to see whether these initial ideas would work in practice, and also wanted to add some new options to the DREAM suite of codes.

Solving discrete optimization problems is by no means easy unless the underlying parameter and objective (likelihood) function space is deterministic with a clear direction of improvement. The DREAM$_{(\mathrm{D})}$ code developed herein is a simple straightforward extension of the DREAM software suite, and includes only a single alteration. The proposal distribution has been adopted so that only integer values are being sampled. In such a way that reversibility of the Markov chains is ensured. This simple modification works surprisingly well for a range of different problems! And this is what is being presented in the paper. I don't claim however, that DREAM$_{(\mathrm{D})}$ is the most efficient or effective to solve an integer optimization problem; Indeed for the sudoku puzzle considered in the paper branch and bound optimization algorithms are way more efficient; but do not provide an estimate of the resulting parameter uncertainty. The code developed herein, DREAM$_{(\mathrm{D})}$ does so, and I therefore suffice to say that DREAM$_{(\mathrm{D})}$ is to the best of my knowledge the first MCMC code to solve discontinuous parameter estimation problems and provide an estimate the of the underlying posterior distribution. This is useful in the context of optimal experimental design. Access to the complete posterior distribution would convey which measurements are extremely important (very well constrained marginal posterior distribution) and which ones are not (wide posterior). Often we resort to a single experimental design, whereas in reality many different designs can yield very similar results for similar operational costs.

I would not argue, however that DREAM$_{(\mathrm{D})}$ is the way to go to solve very large dimensional discrete spaces. Indeed, existing optimization methods can do this perhaps faster, and more efficiently. The thrust of the current paper is just to show how some easy adaptations to the DREAM suite of methods enable solving discontinuous parameter estimation problems!! Obviously, improvements can be made, although this is not straightforward given the constraints posed by detailed balance.

Reversibility of the sampled Markov chains severely reduces the degrees of freedom of the proposal distribution, and dictates the heart of the algorithm. This is most important when evaluating the current manuscript. I believe that the Sudoku puzzle nicely illustrates the ability of $\text{DREAM}_{(\text{D})}$ to solve relatively high-dimensional (50 unknown variables) integer optimization problems. In the revision I will include the likelihood function used to solve the Sudoku. This is simply a combination of the constraint violations used to evaluate the correctness and plausibility of the Sudoku. In principle one can replace the Sudoku with a hydrologic model and setup an experimental design problem, in which the goal is to locate a set of optimum measurement locations. The essence of the approach is similar, but computationally (run time) the Sudoku has some advantages. I will investigate the advantages of a discrete space distance over a linear distance as suggested by reviewer 1.

If of eminent importance I can include a case study where the goal is to find a set of measurement locations that maximize information retrieval for a given hydrologic or environmental system. The last case study (rainfall – runoff transformation) served this purpose, and illustrates why one needs a MCMC algorithm. It provides the underlying posterior for a given sampling grid. This posterior is very similar to a case in which the parameters are assumed continuous.

I can make a better differentiation between cases that use a formal likelihood function (hydrologic modeling example) and an informal likelihood function (Sudoku example). The second class includes examples for which the model itself produces integers.

Then going back to some technical concerns about the algorithm raised by reviewer 1: I disagree that $\text{DREAM}_{(\text{D})}$ is not adaptive. The proposal distribution is of fixed form (like Haario et al.), and the orientation and scale of this distribution is derived from a

number of chains running in parallel (Haario et al. estimates this using the covariance of the history of the sampled points). The crossover value (distribution) is automatically tuned during sampling in such a way that the average squared jumping distance is maximized. This approach was detailed in previous papers, but I will explicitly include this part in the revision. Also, I will further comment on the selection of d', the effective dimensionality used in the jump.

Interactive comment on Hydrol. Earth Syst. Sci. Discuss., 8, 4025, 2011.