Hydrol. Earth Syst. Sci. Discuss., 8, 4025–4052, 2011 www.hydrol-earth-syst-sci-discuss.net/8/4025/2011/ doi:10.5194/hessd-8-4025-2011 © Author(s) 2011. CC Attribution 3.0 License.



This discussion paper is/has been under review for the journal Hydrology and Earth System Sciences (HESS). Please refer to the corresponding final paper in HESS if available.

DREAM_(D): an adaptive markov chain monte carlo simulation algorithm to solve discrete, noncontinuous, posterior parameter estimation problems

J. A. Vrugt^{1,2}

¹Department of Civil and Environmental Engineering, University of California, Irvine, 4130 Engineering Gateway, Irvine, CA 92697-2175, USA ²Institute for Biodiversity and Ecosystem Dynamics, University of Amsterdam, Amsterdam, The Netherlands

Received: 19 April 2011 - Accepted: 20 April 2011 - Published: 26 April 2011

Correspondence to: J. A. Vrugt (jasper@uci.edu)

Published by Copernicus Publications on behalf of the European Geosciences Union.



Abstract

Formal and informal Bayesian approaches are increasingly being used to treat forcing, model structural, parameter and calibration data uncertainty, and summarize hydrologic prediction uncertainty. This requires posterior sampling methods that approximate the (evolving) posterior distribution. We recently introduced the DiffeRential Evolution Adaptive Metropolis (DREAM) algorithm, an adaptive Markov Chain Monte Carlo (MCMC) method that is especially designed to solve complex, high-dimensional and multimodal posterior probability density functions. The method runs multiple chains in parallel, and maintains detailed balance and ergodicity. Here, I present the latest algorithmic developments, and introduce a discrete sampling variant of DREAM that 10 samples the parameter space at fixed points. The development of this new code, DREAM_(D), has been inspired by the existing class of integer optimization problems, and emerging class of experimental design problems. Such non-continuous parameter estimation problems are of considerable theoretical and practical interest. The theory developed herein is applicable to DREAM_(ZS) (Vrugt et al., 2011) and MT-DREAM_(ZS) 15

(Laloy and Vrugt, 2011) as well. Two case studies involving a sudoku puzzle and rainfall – runoff model calibration problem are used to illustrate DREAM(D).

1 Introduction

Formal and informal Bayesian methods have found widespread application and use to
 ²⁰ summarize parameter and model predictive uncertainty in hydrologic modeling. These parameters generally represent model dynamics, but could also include rainfall multipliers (Kavetski et al., 2006; Kuczera et al., 2006; Vrugt et al., 2008), error model variables (Smith et al., 2008; Schoups and Vrugt, 2010), and calibration data measurement errors (Sorooshian and Dracup, 1980; Schaefli et al., 2007; Vrugt et al., 2008). Monte
 ²⁵ Carlo methods are admirably suited to generate samples from the posterior parameter distribution, but generally inefficient when confronted with complex, multimodal,



and high-dimensional model-data synthesis problems. This has stimulated the development of Markov Chain Monte Carlo (MCMC) methods that generate a random walk through the search (parameter space) and iteratively visit solutions with stable frequencies stemming from an invariant probability distribution. If well designed, such MCMC methods should be more efficient than brute force Monte Carlo or importance sampling methods.

5

To visit configurations with a stable frequency, an MCMC algorithm generates trial moves from the current position of the Markov chain x_{t-1} to a new state z. The earliest and most general MCMC approach is the random walk Metropolis (RWM) algorithm (Metropolis et al., 1953). Assume that we have already sampled points $\{x_0, ..., x_{t-1}\}$ this algorithms proceeds in the following three steps. First, a candidate point z is sampled from a proposal distribution q that depends on the present location, x_{t-1} and is symmetric, $q(x_{t-1}, z) = q(z, x_{t-1})$. Next, the candidate point is either accepted or rejected using the Metropolis acceptance probability:

15
$$\alpha(\mathbf{x}_{t-1}, \mathbf{z}) = \begin{cases} \min\left(\frac{\pi(\mathbf{z})}{\pi(\mathbf{x}_{t-1})}, 1\right) & \text{if } \pi(\mathbf{x}_{t-1}) > 0\\ 1 & \text{if } \pi(\mathbf{x}_{t-1}) = 0 \end{cases}$$

where $\pi(\cdot)$ denotes the probability density function (pdf) of the target distribution. Finally, if the proposal is accepted the chain moves to *z*, otherwise the chain remains at its current location x_{t-1} . The result is a Markov chain which under some regularity conditions has a unique stationary distribution with pdf $\pi(\cdot)$.

The standard RWM algorithm has been designed to maintain detailed balance with respect to $\pi(\cdot)$ at each step in the chain:

$$\pi(\mathbf{x}_{t-1})\rho(\mathbf{x}_{t-1} \to \mathbf{z}) = \pi(\mathbf{z})\rho(\mathbf{z} \to \mathbf{x}_{t-1})$$
(2)

where $\pi(\mathbf{x}_{t-1})$ ($\pi(\mathbf{z})$) denotes the probability of finding the system in state $\mathbf{x}_{t-1}(\mathbf{z})$, and $p(\mathbf{x}_{t-1} \rightarrow \mathbf{z})(p(\mathbf{z} \rightarrow \mathbf{x}_{t-1}))$ denotes the conditional probability to perform a trial move ²⁵ from \mathbf{x}_{t-1} to \mathbf{z} (\mathbf{z} to \mathbf{x}_{t-1}). Hastings (Hastings, 1970) extended Eq. (1) to include nonsymmetrical proposal distributions, i.e. $q(\mathbf{x}_{t-1}, \mathbf{z}) \neq q(\mathbf{z}, \mathbf{x}_{t-1})$ in which a proposal jump



(1)

to z and the reverse jump do not have equal probability. This extension is called the Metropolis Hastings algorithm (MH), and has become the basic building block of many existing MCMC sampling schemes.

- Existing theory and experiments prove convergence of well-constructed MCMC schemes to the appropriate limiting distribution under a variety of different conditions. In practice, this convergence is often observed to be impractically slow. This deficiency is frequently caused by an inappropriate selection of $q(\cdot)$ used to generate trial moves in the Markov Chain. This inspired Vrugt et al. (2008, 2009) to develop a simple adaptive RWM algorithm called Differential Evolution Adaptive Metropolis (DREAM) that runs multiple chains simultaneously for global exploration, and automatically tunes the scale and orientation of the proposal distribution during the evolution to the posterior distribution. This scheme is an adaptation of the Shuffled Complex Evolution Metropolis (Vrugt et al., 2003) global optimization algorithm and has the advantage of maintaining
- detailed balance and ergodicity while showing excellent efficiency on complex, highly nonlinear, and multimodal target distributions (Vrugt et al., 2008, 2009).

20

25

In DREAM, *N* different Markov Chains are run simultaneously in parallel. If the state of a single chain is given by a single *d*-dimensional vector *x*, then at each generation the *N* chains in DREAM define a population *X*, which corresponds to an *N* x *d* matrix, with each chain as a row. Jumps in each chain $i = \{1, ..., N\}$ are generated by taking a fixed multiple of the difference of the states of randomly chosen pairs of chains of *X*:

$$\boldsymbol{z}^{i} = \boldsymbol{x}^{i} + (\boldsymbol{1}_{d} + \boldsymbol{e}_{d})\boldsymbol{\gamma}(\delta, d') \left(\sum_{j=1}^{\delta} \boldsymbol{x}^{r_{1}(j)} - \sum_{n=1}^{\delta} \boldsymbol{x}^{r_{2}(n)}\right) + \boldsymbol{\epsilon}_{d}$$
(3)

where δ signifies the number of pairs used to generate the proposal, $\mathbf{x}^{r_1(j)}$ and $\mathbf{x}^{r_2(n)}$ are randomly selected without replacement from the population \mathbf{X}_{t-1}^{-i} (the population without \mathbf{x}_{t-1}^i); $r_1(j), r_2(n) \in \{1, ..., N\}$ and $r_1(j) \neq r_2(n)$. The values of \mathbf{e}_d and \mathbf{e}_d are drawn from $U_d(-b,b)$ and $N_d(0,b^*)$ with b and b^* small compared to the width of the target distribution, respectively, and the value of the jump-size, γ depends on δ and



d', the number of dimensions that will be updated jointly. By comparison with RWM, a good choice for $\gamma = 2.4/\sqrt{2\delta d'}$ (Roberts and Rosenthal, 2001; Ter Braak, 2006). This choice is expected, for Gaussian and Student target distributions, to yield an acceptance probability of 0.44 for d' = 1, 0.28 for d' = 5 and 0.23 for large d'. Every 5th generation $\gamma = 1.0$ to facilitate jumping between disconnected posterior modes (Vrugt et al., 2008).

5

The difference vector in Eq. (3) contains the desired information about the scale and orientation of the target distribution, $\pi(\mathbf{x}|\cdot)$. By accepting each jump with the Metropolis ratio min $\left\{\pi(\mathbf{z}^i|\cdot)/\pi(\mathbf{x}_{t-1}^i|\cdot), 1\right\}$, a Markov chain is obtained, the stationary or limiting

- distribution of which is the posterior distribution. The proof of this is given in Ter Braak and Vrugt (2008) and Vrugt et al. (2008, 2009). Because the joint pdf of the *N* chains factorizes to $\pi(x^1|\cdot) \times \ldots \times \pi(x^N|\cdot)$, the states $x^1 \ldots x^N$ of the individual chains are independent at any generation after DREAM has become independent of its initial value. After this burn-in period, the convergence of DREAM can thus be monitored with the
- $\overset{15}{R}$ -statistic of Gelman and Rubin (1992). This convergence diagnostic compares the within and in-between variances of the *N* different chains.

Various recent contributions have shown the utility of MCMC simulation with DREAM to treat different error sources, and help quantify and analyze parameter, model structural, forcing, and calibration data uncertainty (Vrugt et al., 2008, 2009b; Dekker et

- al., 2010; He et al., 2010; Huisman et al., 2010; Keating et al., 2010; Minasny et al., 2011; Schoups and Vrugt, 2010; Vrugt et al., 2011). These, and most other model-data synthesis studies in the earth sciences, typically involve continuous variables (parameters and probability density functions) that can take on any numerical value within their prior ranges defined by the user. Yet, relatively little attention has been given to pos-
- terior sampling problems involving discrete variables. The existing DREAM framework has been developed based on the assumption of continuity of the parameter space, $\lim_{x \to c} \pi(x) = \pi(c)$, and thus will exhibit some practical problems when confronted with discrete variables. Such non-continuous parameter estimation problems are abundant



in many fields of study, and therefore of considerable theoretical and practical interest. For instance, in hydrology there is increasing interest in using optimization approaches to help find optimal experimental design strategies that attempt to minimize cost, parameter and model predictive uncertainty, or combinations thereof. This involves selecting one or multiple different measurement locations amongst a discrete set of possibilities. The solution of this problem lies in an adaptation of DREAM, and its various extensions including $\text{DREAM}_{(ZS)}$ (Vrugt et al., 2011), and MT-DREAM_(ZS) (Laloy and Vrugt, 2011).

In this paper we present a discrete implementation of DREAM, that is especially designed to solve noncontinuous search and optimization problems. This new code, DREAM_(D) uses DREAM as its main building block, yet implements integer search to facilitate solving discrete sampling problems. The DREAM_(D) algorithm maintains detailed balance and ergodicity, and achieves excellent performance across a range of noncontinuous posterior sampling problems. The DREAM_(D) code is perhaps the only method available to solve discontinuous parameter estimation problems, and simultanocurbu provide an estimate of upportainty.

neously provide an estimate of uncertainty. The remainder of this paper is organized as follows. Section 2 presents a short introduction to MCMC, followed by a detailed description of $DREAM_{(D)}$ in Sect. 3. Section 4 demonstrates the performance of $DREAM_{(D)}$ using two different case studies involving

²⁰ a simple 52-dimensional sudoku puzzle, and a rainfall-runoff model calibration problem. These results are illustrated in detail, and used to highlight further possible improvements. Finally, in Sect. 5 we summarize the methodology and discuss the results.

2 Nonlinear optimization involving discrete variables

5

 Discrete optimization problems are abundant in many fields of study, and have begun
 to appear in the hydrologic literature (Furman et al., 2004; Harmancioglu et al., 2004;
 Perrin et al., 2008; Kleidorfer et al., 2008; Neuman et al., 2011). To illustrate a noncontinuous parameter estimation problem, please consider Fig. 1 that presents a tile



puzzle consisting of sixteen different surfaces. Each tile contains a different letter from the alphabet. The goal is to get the letters in the appropriate order. The solution to this problem is immediately obvious to a human, but not immediately clear to a computer. A search algorithm is therefore required to solve this problem.

⁵ If we assign numbers to each letter, a = 1, b = 2 and so forth, we could measure the distance from our initial guess to the actual solution. This constitutes an integer optimization problem. The tiles can only take on integer values, between 1 and 16. This results in a d = 16 dimensional search problem with each dimension $x \in [1, 2, ..., 16]$. Figure 1a illustrates an initial guess, and in a series of panels (B \Rightarrow D) it is shown how DREAM_(D) translates this guess into the final solution.

Other problems that require integer search involve finding optimal experimental design strategies. Usually this consists of finding the best measurement locations among a prior defined and often restricted set of possibilities. This requires integer optimization in a similar way as done in the tile puzzle.

15 3 $DREAM_{(D)} \Rightarrow$ differential evolution adaptive metropolis with discrete sampling

We now describe our new code, entitled $\mathsf{DREAM}_{(\mathsf{D})},$ which uses DREAM as main building block.

Let X be a $N \times d$ -matrix defining the N initialized starting positions, x', i = 1, ..., N of the different Markov chains by drawing samples from $p_d(x)$, the prior distribution.

²⁰ Similarly, let **Z** be an external archive of points that periodically appends the elements of **X** at regular intervals. The initial population $[X_t; t = 0]$ is translated into a sample from the posterior target distribution using the following pseudo code:

1. Set $T \leftarrow 1$ FOR $m \leftarrow 1, ..., K$ DO (POPULATION EVOLUTION) FOR $i \leftarrow 1, ..., N$ DO (CHAIN EVOLUTION)



25

a. Generate a candidate point, z^i in chain *i*,

$$\boldsymbol{z}^{i} = \boldsymbol{x}^{i} + \left\| (\boldsymbol{1}_{d} + \boldsymbol{e}_{d}) \boldsymbol{\gamma}(\delta, d') \left(\sum_{j=1}^{\delta} \boldsymbol{x}^{r_{1}(j)} - \sum_{n=1}^{\delta} \boldsymbol{x}^{r_{2}(n)} \right) + \boldsymbol{\varepsilon}_{d} \right\|_{d}$$
(4)

where the function $\|\cdot\|_d$ rounds each element j = 1, ..., d of the jump vector to the nearest integer.

b. Replace each element (j = 1, ..., d) of the proposal z_j^i with z_j^i using a binomial scheme with probability 1 - CR,

$$z_{j}^{i} = \begin{cases} x_{j}^{i} \text{ if } U \le 1 - CR, \quad d' = d' - 1\\ z_{j}^{i} \text{ otherwise} \end{cases}$$
(5)

where *CR* denotes the crossover probability, and $U \in [0, 1]$ is a draw from a uniform distribution.

c. Compute $\pi(z^i)$ and accept the candidate points with Metropolis acceptance probability, $\alpha(\mathbf{x}^i, \mathbf{z}^i)$,

$$\alpha(\boldsymbol{x}^{i}, \boldsymbol{z}^{i}) = \begin{cases} \min\left(\frac{\pi(\boldsymbol{z}^{i})}{\pi(\boldsymbol{x}^{i})}, 1\right) & \text{if } \pi(\boldsymbol{x}^{i}) > 0\\ 1 & \text{if } \pi(\boldsymbol{x}^{i}) = 0 \end{cases}$$
(6)

d. If accepted, move the chain to the candidate point, $x^i = z^i$, otherwise remain at the old location, x^i .

15

10

5

END FOR (CHAIN EVOLUTION) END FOR (POPULATION EVOLUTION)

- 2. Append *X* to *Z*.
- 3. Compute the Gelman-Rubin convergence diagnostic, \hat{R}_j (Gelman and Rubin, 1992) for each dimension j = 1, ..., d using the last 50% of the samples of Z.



The transition kernel in DREAM(D) is especially designed to handle discrete search

4. If $\hat{R}_j \leq 1.2$ for j = 1, ..., d or $T > T_{max}$, stop and go to step 5, otherwise set $T \leftarrow T + 1$

5. Summarize the posterior pdf using Z after discarding the initial and burn-in sam-

The DREAM(D) algorithm presented herein is similar as DREAM, but especially de-

signed to solve discrete search and optimization problems. The function $\left[\cdot\right]_{\alpha}$ in Eq. (4) forces the jumps to maintain integer values, and maintains detailed balance (as demon-

strated later). Compared to the original DREAM sampling scheme, DREAM(D) contains two additional algorithmic parameters. These are T_{max} , the maximum number of generations, and K, the thinning rate used to periodically add samples to Z. Based on

recommendations in (Vrugt et al., 2011), we set K = 10, and assign a large value for

 T_{max} , thus DREAM_(D) automatically stops after convergence has been achieved. To speed up convergence to the target distribution, DREAM estimates a distribution of

and go to POPULATION EVOLUTION.

ples.

5

15

CR values during burn-in that favors large jumps over smaller ones in each of the N chains. Details of this can be found in (Vrugt et al., 2008, 2009, 2011).

and optimization problems, yet only samples integer values. This is by no means a limitation, but a simple computational trick is therefore required to solve non-integer problems. For such problems, we discretize the range of each individual parameter into

- equidistant intervals, and rewrite the actual problem using integers only. The posterior 20 distribution of these integers is subsequently derived with DREAM(D). For example, consider a discrete parameter that can only take on the values $x_1 \in [0, 0.2, 0.4, \dots, 5]$. We can rewrite this problem as: $x_1 = \min(x_1) + 0.2j$ and sample $j \in [0, 1, \dots, 25]$ using DREAM(D). This approach, if deemed necessary, allows for a different discretization
- of each individual parameter. Also, the simultaneous use of DREAM and DREAM_(D), 25 enables joint sampling of continuous and discrete parameters.



3.1 DREAM_(D) \Rightarrow Detailed Balance?

We are now left with a proof that $DREAM_{(D)}$ yields an invariant distribution that is identical to the posterior distribution. For this we need to demonstrate that the transition kernel in DREAM maintains detailed balance, and thus results in a reversible Markov chain. This essentially means that the forward $(p(x^i \rightarrow z^i))$ and backward $(p(z^i \rightarrow x^i))$ jump have equal probability at every single step in the chain. This is easy to proof for standard RWM algorithms that use a fixed proposal distribution. Yet, the jumping distribution used in DREAM_(D) is adaptive, and continuously changes scale and orientation en route to the posterior distribution. This significantly enhances efficiency, but it is not

¹⁰ immediately clear whether Eq. (4) also satisfies the detailed balance condition.

In previous papers, we have given formal proofs of convergence of DREAM (Vrugt et al., 2008), DREAM_(ZS) (Vrugt et al., 2011), and MT-DREAM_(ZS) (Laloy and Vrugt, 2011) to the appropriate limiting distribution. For simplicity, we use a hypothetical example to illustrate that DREAM_(D) maintains detailed balance. Lets consider a twodimensional, (*d* = 2) discrete sampling problem with *N* = 5 different chains, δ = 1, and $\gamma = 2.4/\sqrt{2\delta d'} = 2.4/\sqrt{2 \times 1 \times 2} = 1.2$. The initial states of these chains are color coded and depicted in Fig. 2, and used to generate candidate points. Lets assume that the jump in the first chain (purple) uses the states of the 2nd (r_1 : blue), and 4th (r_2 : green) chain, and that $e_j = e_j = 0$; j = 1, ..., 2. Following Eq. (4), the forward jump and proposal point then becomes:

$$z^{i} = [22] + \left\| 1.2([44] - [51]) \right\| = [22] + \left\| [-1.23.6] \right\| = [16]$$
(7)

This point is indicated in Fig. 2 with the red square. Lets assume that we accept this candidate point, and chain 1 transitions to this new state.

We are now left with studying the probability of the backward jump. This requires the green chain to be r_1 , and the blue chain to be r_2 , thus selected in opposite order from the previous jump. The chance of this backward jump is equal to the chance of the forward jump because of the uniform random number generator used to select the



respective chain pairs for the candidate point. Hence, the chance to select r_1 as chain 2, and r_2 as chain 4 is equal to drawing $r_2 = 4$, and $r_2 = 2$. Reversibility is thus ensured, yet is is not directly obvious that this proof also holds for DREAM(D) that uses an explicit integer rounding function in the transition kernel. If we proceed with our new selection ⁵ of r_1 and r_2 then the next candidate point becomes:

$$z^{i} = [16] + ||1.2([51] - [44])|| = [16] + ||[1.2 - 3.6]|| = [22]$$
(8)

Indeed, the reverse jump results in a proposal point that is identical to the initial state of chain 1 (purple point), and reversibility is ensured. Apparently, the function, $\|\cdot\|_d$, used to round the *d*-dimensional jumping vector to the closest integers does not violate detailed balance. This proof also holds when e_d , and e_d are drawn from their 10 respective symmetric probability distributions, and when $\delta > 1$ and d > 2. We leave this up to the reader. This concludes the proof of detailed balance.

A final remark is appropriate. In theory it is possible that at least one of the d arguments, y_d of the rounding function, $||y_d||$ has a fractional part of .5. Or in mathematical notation, $\exists y_i \in Y_i$; $y_i - ||y_i|| = 0.5$; j = 1, ..., d, where Y_i denotes the feasible jump space 15 of the *i*th dimension, generated with Eqs. (3) or (4). If this (mathematical) statement is true, then the respective argument(s) is (are) rounded down (floor) to the closest integer. This directed rounding violates detailed balance and introduces a possible bias in the proposal jump. A simple one-dimensional example will immediately illustrate this, but the chance this bias happens in practice is virtually zero. If nothing else, the 20 stochastic nature of $e_d \sim U_d(-b,b)$, and $e_d \sim N_d(0,b^*)$ will eliminate this possibility. In the rare event of $y_i - ||y_i|| = 0.5$ we implement stochastic rounding, and choice among $y_i = 0.5$ and $y_i = 0.5$ with equal probability. This ensures reversibility of the Markov chains generated with DREAM_(D). Note that the size of Y_i essentially depends on the choice of the prior distribution.



4 Case studies

I now present two different case studies with increasing complexity. The first study consists of a typical integer estimation problem, and involves a sudoku puzzle. This puzzle has become quite popular in the past 10 years, and many newspapers, jour-

⁵ nals, and magazines around the world publish sudokus for entertainment. This synthetic study illustrates the ability of DREAM_(D) to help solve a relatively difficult integer optimization problem. The second study is concerned with estimating parameters in a mildly complex lumped watershed model using observed daily discharge data from the Guadalupe River in Texas. The parameter estimation problem is posed in discrete
 ¹⁰ form, and solved using DREAM_(D). This results in a discrete The posterior parameter distribution is compared against a classical continuous formulation of the model calibration inverse problem separately inferred using DREAM_(ZS), and used to highlight the advantages of DREAM_(D) and discrete sampling.

4.1 The daily sudoku

The first case study considers a synthetic integer parameter estimation problem, involving a sudoku puzzle. The objective is to fill a 9 × 9 grid with digits so that each column, each row, and each of the nine 3 × 3 sub-grids that make up the total square contains values of 1 to 9. The same single integer may not appear twice in the same 9 × 9 playing board row or column or in any of the nine 3 × 3 subregions. The puzzle setter provides a partially completed grid, which typically has a single (unique) solution. The puzzle was popularized in 1986 by the Japanese puzzle company Nikoli, under the name Sudoku, meaning single number (Hayes, 2006). Nowadays, Sudoku puzzles are very popular, and widely practised by many millions of people throughout the world.

I consider the Sudoku puzzle in Fig. 3, taken from Wikipedia (http://en.wikipedia. org/wiki/Sudoku). The initial grid is depicted at the left-hand side, whereas the final solution is presented at the right-hand side. In this particular puzzle, the solution at 29 different cells is known, leaving us with d = 81 - 29 = 52 parameter values to be



estimated. These parameters can take on values between 1 to 9. Figure 4 illustrates the sampled values of $DREAM_{(D)}$ at various stages during the integer search. A total of N = 25 different chains were used to search the parameter space, and the initial sample was created using Latin Hypercube Sampling. I did not impose any constraints on this initial solution, and thus each value of 1 to 9 can appear in any number in the grid. The log-likelihood function measures the constraint violation, details of which are

outside the scope of this publication. The first grid, at the left-hand side of Fig. 4 presents an example of a possible starting solution used in one of the chains. Obviously, this solution is rather bad, and violates each of the three different constraints discussed previously (1 to 9 in each column and row, and each 3 × 3 subregion). The second grid, Fig. 4b, shows considerable improvements, and better resemblances the final solution. The third panel in Fig. 4c is a further refinement, but with a few noticeable deviations from the true solution. Finally,

after about 1 million sudoku function evaluations, the puzzle is successfully solved (Fig. 4d).

Obviously, it is inspiring that $DREAM_{(D)}$ is able to successfully solve a Sudoku puzzle, but the required number of function evaluations (computational time) to do so is rather large. Indeed, a branch and bound optimization approach would be about 3 orders of magnitude more efficient. Such an approach shuffles the d = 52 integer values until all constraint are met. We could implement this procedure in $DREAM_{(D)}$, and change the proposal distribution in such a way that jumps in each individual chain are no longer created by taking the difference of the states of two (or more) other chains but simply generated by shuffling selected parameter values of the same chain. Yet, it is not directly obvious how to make such modification and ensure reversibility of the

²⁵ Markov chain. This is an interesting problem, and deserves additional investigation, and theoretical development.



4.2 Watershed model calibration using discrete parameter estimation

The final case study involves flood forecasting, and consists of the calibration of a mildly complex lumped watershed model using historical data from the Guadalupe River at Spring Branch, Texas. This is the driest of the 12 MOPEX basins described in the study

- of Duan et al. (2006). The model structure and hydrologic process representations are found in (Schoups and Vrugt, 2010). The model transforms rainfall into runoff at the watershed outlet using explicit process descriptions of interception, throughfall, evaporation, runoff generation, percolation, and surface and subsurface routing. Table 1 summarizes the seven different model parameters, and their prior uncertainty ranges.
- Each parameter is discretized equidistantly in 250 intervals with respective step size listed in the last column at the right hand side. This gridding is necessary to create a non-continuous, discrete, parameter estimation problem. Unlike the previous case study in which integer values are sampled only, this particular study (mostly) involves non-integer values.
- Daily discharge, mean areal precipitation, and mean areal potential evapotranspiration were derived from (Duan et al., 2006) and used for model calibration. Details about the basin, experimental data, and likelihood function can be found there, and will not be discussed herein. The same model and data was used in a previous study (Schoups and Vrugt, 2010), and used to introduce a generalized likelihood function for
 heteroscedastic, non-Gaussian, and correlated (streamflow) prediction errors.

Figure 5 presents histograms of the marginal distribution of a selected few hydrologic model parameters using five years of observed daily discharge data. The top panel presents the results of DREAM_(D), whereas the bottom panel presents the results for a continuous parameter space. These histograms were derived by separately running DREAM for the same data set and model. Notice the close agreement between the histograms derived with both MCMC methods. This is a testament to the ability of DREAM_(D) to successfully solve discrete posterior parameter estimation problems. The influence of gridding is hardly noticeable, but becomes apparent if we use at least 25 bins to represent the marginal density (not shown herein).



To better illustrate the effect of discretization, please consider Figure 6 that presents two-dimensional scatter plots of the $\mathsf{DREAM}_{(D)}$ derived posterior samples for a few selected parameter pairs. The bottom panel shows similar plots but then assuming continuity of the parameter space. The effect of gridding is immediately apparent. Whereas

- the original bivariate scatter plots sample the parameter space in a (bloodstain) spatter pattern, two-dimensional plots of the posterior samples derived with DREAM_(D) exhibit an obvious grid pattern with horizontally and vertically aligned points. The posterior samples take on discrete values with a distance between subsequent points that is similar to the intervals listed in Table 1. Despite this difference in sampling pattern, the
- ¹⁰ shape of the DREAM and DREAM_(D) derived bivariate scatter plots are very similar, commensurate with the covariance structure of the posterior distribution. The results presented in Fig. 5 inspire confidence in the ability of DREAM_(D) to solve noncontinuous posterior sampling problems.

The excellent correspondence of the posterior parameter distributions derived with DREAM and DREAM_(D) results in marginal differences of the resulting streamflow predictions. I therefore do not show any times series plots of model predictions, and corresponding 95% uncertainty ranges. Such plots can be found in (Schoups and Vrugt, 2010) and details can be found in that publication. It is interesting to observe that the maximum log-likelihood value of 543 found with DREAM_(D) is somewhat larger than its
 counterpart estimated with DREAM (540). This difference was consistently observed for multiple different trials with both MCMC algorithms.

To provide better insights into the efficiency of $DREAM_{(D)}$, Fig. 7a plots the evolution of the \hat{R} -statistic of Gelman and Rubin (Gelman and Rubin, 1992) for each of the different parameters. To benchmark these results, the bottom panel (Fig. 7b) illustrates the convergence results of DREAM assuming continuity of the parameter space. The presented traces represent an average of 25 different MCMC trials. The convergence speed for both algorithms is strikingly similar. Both MCMC methods need about 30 000 model evaluations to converge to a stationary (limiting) distribution (indicated with the dashed blue horizontal line). Although gridding significantly reduces the size of the



feasible parameter space, this has no immediate effect on sampling efficiency. An approximately similar number of model runs is required to converge to and explore the posterior target distribution. Yet, this by no means is a universal finding. My initial results to date for more complex models with much higher parameter dimensionality demonstrate considerable enhancements in efficiency (sometimes dramatically) when

demonstrate considerable enhancements in efficiency (sometimes dramatically) when solving the calibration problem in the discretized rather than continuous space. Such transformation is particularly useful for insensitive parameters as gridding significantly reduces the space of feasible solutions. Further research on this topic is warranted.

5 Conclusions

- ¹⁰ In the past decade much progress has been made in the development of sampling algorithms for statistical inference of the posterior parameter distribution. The typical assumption in this work is that the parameters are continuous and can take on any value within their upper and lower bounds. Unfortunately, such problems typically do not work for discrete parameter estimation problems. Such problems are abundant
- in many fields of study, and therefore of considerable theoretical and practical interest. Examples include selecting among different measurement locations in the design of optimal experimental strategies, finding the best members of an ensemble of predictors, and more generally discrete model calibration problems. Here, I have introduced a discrete MCMC simulation algorithm that is especially designed to solve non-continuous
- posterior sampling problems. This method, entitled DREAM_(D) uses DREAM as its main building block, yet uses a modified proposal distribution to facilitate solve discrete sampling problems. The DREAM_(D) algorithm maintains detailed balance and ergodicity, and receives good performance across a range of problems involving a sudoku puzzle, and discrete rainfall – runoff model calibration problem.
- The theory developed herein is easily implemented in DREAM_(ZS) (Vrugt et al., 2011) and MT-DREAM_(ZS) (Laloy and Vrugt, 2011), which provides a venue to further increase the efficiency of MCMC simulation. The DREAM_(D) code is written in MATLAB and is available upon request (jasper@uci.edu).



Acknowledgements. I gratefully acknowledge the insightful comments of the three different reviewers. Computer support, provided by the SARA center for parallel computing at the University of Amsterdam, The Netherlands, is highly appreciated.

References

- ⁵ Box, G. E. P. and Tiao, G. C.: Bayesian Inference in Statistical Analyses Addison-Wesley-Longman, Reading, Massachusetts, 1973.
 - Dekker, S. C., Vrugt, J. A., and Elkington, R. J.: Significant variation in vegetation characteristics and dynamics from ecohydrologic optimality of net carbon profit, Ecohydrology, doi:10.1002/eco.177, 2010. 4029
- ¹⁰ Duan, Q., Gupta, V. K., and Sorooshian, S.: Effective and efficient global optimization for conceptual rainfall-runoff models, Water Resour. Res., 28, 1015–1031, 1992.
 - Duan, Q., Schaake, J., Andrassian, V., Franks, S., Goteti, G., Gupta, H. V., Gusev, Y. M., Habets, F., Hall, A., Hay, L., Hogue, T., Huang, M., Leavesley, G., Liang, X., Nasonova, O. N., Noilhan, J., Oudin, L., Sorooshian, S., Wagener, T., and Wood, E. F.: Model Parameter
- Estimation Experiment (MOPEX): An overview of science strategy and major results from the second and third workshops, J. Hydrol., 320, 3–17, 2006. 4038

Furman, A., Ferré, T. P. A., and Warrick, A. W.: Optimization of ERT surveys for monitoring transient hydrological events using perturbation sensitivity and genetic algorithms, Vadose Zone Journal, 3, 1230–1239, 2004. 4030

- Gelfand, A. E. and Smith, A. F.: Sampling based approaches to calculating marginal densities, J. Am. Stat. Assoc., 85, 398–409, 1990.
 - Gelman, A. G. and Rubin, D. B.: Inference from iterative simulation using multiple sequences, Stat. Sci., 7, 457–472, 1992. 4029, 4032, 4039, 4052

Gilks, W. R., Roberts, G. O., and George, E. I.: Adaptive direction sampling, Statistician, 43, 179–189, 1994.

25

30

Gilks, W. R. and Roberts, G. O.: Strategies for improving MCMC, in Markov Chain Monte Carlo in Practice, edited by: Gilks, W. R., Richardson, S., and Spiegelhalter, D. J., Chapman & Hall, London, 89–114, 1996.

Hastings, W. K.: Monte Carlo sampling methods using Markov chains and their applications, Biometrika, 57, 97–109, 1970. 4027



4042

- Haario, H., Saksman, E., and Tamminen, J.: Adaptive proposal distribution for random walk Metropolis algorithm, Computational Statistics, 14, 375–395, 1999.
- Haario, H., Saksman, E., and Tamminen, J.: An adaptive Metropolis algorithm, Bernoulli, 7, 223–242, 2001.
- ⁵ Haario, H., Saksman, E., and Tamminen, J.: Componentwise adaptation for high dimensional MCMC, Computation. Stat., 20, 265–274, 2005.
 - Haario, H., Laine, M., Mira, A., and Saksman, E.: DRAM: Efficient adaptive MCMC, Stat. Comput., 16, 339–354, 2006.
 - Harmancioglu, N. B., Icaga, Y., and Gul, A.: The use of an optimization method in assessment of water guality sampling sites, European Water, 5/6, 25–35, 2004. 4030
- Hayes, B.: Unwed numbers, American Scientist, 94(1), 12–15, doi:10.1511/2006.1.12, 2006. 4036
 - He, M., Hogue, T. S., Franz, K. J., Margulis, S. A., and Vrugt, J. A.: Characterizing parameter sensitivity and uncertainty for a snow model across hydroclimatic regimes, Adv.n Water
- ¹⁵ Resour., 34, 114–127, doi:10.1016/j.advwatres.2010.10.002, 2010. 4029

10

30

Huisman, J. A., Rings, J., Vrugt, J. A., Sorg, J., and Vereecken, H.: Hydraulic properties of a model dike from coupled Bayesian and multi-criteria hydrogeophyliscal inversion, J. Hydrol., 380(1–2), 62–73, doi:10.1016/j.jhydrol.2009.10.023, 2010. 4029

Kavetski, D., Kuczera, G., and Franks, S. W.: Bayesian analysis of input uncer-

- tainty in hydrologic modeling: 2. Application, Water Resour. Res., 42, W03408, doi:10.1029/2005WR004376, 2006. 4026
 - Keating, E., Doherty, J., Vrugt, J. A., and Kang, Q.: Optimization and uncertainty assessment of strongly non-linear groundwater models with high parameter dimensionality, Water Resour. Res., 46, W10517, doi:10.1029/2009WR008584, 2010. 4029
- Kleidorfer, M., Möderl, M., Fach, S., and Rauch, W.: Optimization of measurement campaigns for calibration of a hydrological model, Proceedings of the 11th International Conference on Urban Drainage, Edinburgh, Scotland, UK, 2008. 4030
 - Kuczera, G., Kavetski, D., Franks, F. W., and Thyer, M. T.: Towards a Bayesian total error analysis of conceptual rainfall-runoff models: Characterising model error using storm-dependent parameters, J. Hydrol., 331(1–2), 161–177, 2006. 4026
- Laloy, E. and Vrugt, J. A.: High-dimensional posterior exploration of hydrologic models using multi-try DREAM_(ZS) and high performance computing, Water Resour. Res., in review, 2001. 4030, 4034, 4040



- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E.: Equation of state calculations by fast computing machines, J. Chem. Phys., 21, 1087–1092, 1953. 4027
- Minasny, B., Vrugt, J. A., and McBratney, A. B.: Confronting uncertainty in model-based geostatistics using Markov chain Monte Carlo simulation, Geoderma, in press, 2001. 4029
- Neuman, S. P., Xue, L., Ye, M., and Lu, D.: Bayesian analysis of data-worth considering model and parameter uncertainties, Adv. Water Resour., doi:10.1016/j.advwatres.2011.02.007, in press, 2011. 4030
 - Perrin, C., Andréassian, V., Rojas Serna, C., Mathevet, T., and Le Moine, N.: Discrete parameterization of hydrological models: Evaluating the use of parameter sets libraries over 900
- catchments, Water Resour. Res., 44, W08447, doi:10.1029/2007WR006579, 2008. 4030 Price, K. V., Storn, R. M., and Lampinen, J. A.: Differential Evolution, A Practical Approach to Global Optimization, Springer, Berlin, 2005.
 - Roberts, G. O., and Gilks, W. R.: Convergence of adaptive direction sampling, J. Multivariate Anal., 49, 287–298, 1994.
- ¹⁵ Roberts, G. O. and Rosenthal, J. S.: Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms, J. Appl. Probab., 44, 458–475, 2007.
 - Roberts, G. O. and Rosenthal, J. S.: Examples of adaptive MCMC, online manucript, 2008.
 - Roberts, G. O. and Rosenthal, J. S.: Optimal scaling for various Metropolis-Hastings algorithms, Stat. Sci., 16, 351–367, 2001. 4029
- Schaefli, B., Talamba, D. B., and Musy, A.: Quantifying hydrological modeling errors through a mixture of normal distributions, J. Hydrol., 332, 303–315, 2007. 4026
 - Schoups, G. and Vrugt, J. A.: A formal likelihood function for parameter and predictive inference of hydrologic models with correlated, heteroscedastic and non-gaussian errors, Water Resour. Res., 46, W10531, doi:10.1029/2009WR008933, 2010. 4026, 4029, 4038, 4039
- Smith, T. J. and Marshall, L. A.: Bayesian methods in hydrologic modeling: A study of recent advancements in Markov chain Monte Carlo techniques, Water Resour. Res., 44, W00B05, doi:10.1029/2007WR006705, 2008. 4026
 - Sorooshian, S. and Dracup, J. A.: Stochastic parameter estimation procedures for hydrologic rainfall-runoff models: correlated and heteroscedastic error cases, Water Resour. Res.,
- ³⁰ 16(2), 430–442, doi:10.1029/WR016i002p00430, 1980. 4026 Storn, R. and Price, K.: Differential evolutiona simple and efficient heuristic for global optimization over continuous spaces, Global Optimization, 11, 341–359, 1997.

Ter Braak, C. J. F.: A Markov chain Monte Carlo version of the genetic algorithm differential



evolution: easy Bayesian computing for real parameter spaces, Stat. Comput., 16, 239–249, 2006. 4029

- Ter Braak, C. J. F. and Vrugt, J. A.: Differential evolution Markov chain with snooker updater and fewer chains, Stat. Comput., 16(3), 239–249, doi:10.1007/s11222-008-9104-9, 2008. 4029
- ⁵ Vrugt, J. A., ter Braak, C. J. F., Diks, C. G. H., Higdon, D., Robinson, B. A., and Hyman, J. M.: Accelerating Markov chain Monte Carlo simulation by differential evolution with self-adaptive randomized subspace sampling, Int. J. Nonlin. Sci. Num., 10, 271–288, 2009. 4028, 4029, 4033

Vrugt, J. A., ter Braak, C. J. F., and Diks, C. G. G.: A particle DREAM filter for posterior tracking

- of hydrologic model parameters and states, Water Resour. Res., in review, 2011. 4029, 4030, 4033, 4034
 - Vrugt, J. A., ter Braak, C. J. F., Gupta, H. V., and Robinson, B. A.: Equifinality of formal (DREAM) and informal (GLUE) Bayesian approaches in hydrologic modeling?, Stoch. Env. Rese. Risk A., 23(7), 1011–1026, doi:10.1007/s00477-008-0274-y, 2009. 4029
- ¹⁵ Vrugt, J. A., ter Braak, C. J. F., Clark, M. P., Hyman, J. M., and Robinson, B. A.: Treatment of input uncertainty in hydrologic modeling: Doing hydrology backward with Markov chain Monte Carlo simulation, Water Resour. Res., 44, W00B09, doi:10.1029/2007WR006720, 2008. 4026, 4028, 4029, 4033, 4034

Vrugt, J. A., Gupta, H. V., Bouten, W., and Sorooshian, S.: A shuffled complex evolution

- ²⁰ metropolis algorithm for optimization and uncertainty assessment of hydrologic model parameters, Water Resour. Res., 39, 1–14, doi:10.1029/2002WR001642, 2003. 4028
 - Vrugt, J. A., Nualláin, B. O, Robinson, B. A., Bouten, W., Dekker, S. C., and Sloot, P. M. A.: Application of parallel computing to stochastic parameter estimation in environmental models, Comput. Geosci., 32(8), doi:10.1016/j.cageo.2005.10.015, 1139–1155, 2006.
- ²⁵ Vrugt, J. A., Laloy, E., and ter Braak, C. J. F.: Differential evolution adaptive Metropolis with sampling from past states, SIAM J. Optimiz., in review, 2011. 4040

lier leeinn Da	HESSD 8, 4025–4052, 2011								
DDr	DREAM _(D) MCMC s	DREAM _(D) → discrete MCMC simulation							
	J. A.	Vrugt							
	Title	Page							
none	Abstract	Introduction							
	Conclusions	References							
	Tables	Figures							
	14	►I.							
כ		•							
DDr	Back	Close							
_	Full Scr	een / Esc							
	Printer-frie	ndly Version Discussion							
Dunor	œ	O BY							

Discussion Pap	HE 8, 4025–4	SSD 4052, 2011
ēr	DREAM _(D) MCMC s	→ discrete imulation
Discu	J. A.	Vrugt
Ission P	Title	Page
aper	Abstract	Introduction
	Conclusions	References
Discu	Tables	Figures
ISSIO	۱۹	►I.
n Pal	•	•
oer	Back	Close
—	Full Scr	een / Esc
Discussion	Printer-frie Interactive	ndly Version Discussion
Paper	C	D BY

Table 1. Prior Uncertainty Ranges of Hydrologic and Error Model Parameters.

Parameter	Symbol	Minimum	Maximum	Units	Step size
Maximum interception	I _{max}	0	10	mm	0.02
Soil water storage capacity	S_{max}	10	1000	mm	1.98
Maximum percolation rate	Q_{\max}	0	100	$\rm mmd^{-1}$	0.20
Evaporation parameter	α_{E}	0	100	_	0.20
Runoff parameter	α_{F}	-10	10	_	0.04
Time constant, fast reservoir	K_{F}	0	10	days	0.02
Time constant, slow reservoir	Ks	0	150	days	0.30
Heteroscedasticity intercept	σ_{0}	0	1	mm/d	0.002
Heteroscedasticity slope	σ_1	0	1	—	0.002
Autocorrelation coefficient	ϕ_1	0	1	_	0.002
Kurtosis parameter	β	-1	1	-	0.004

	Discussion Paper 8, 4025–4052, 2011 DREAM _(D) \rightarrow discrete MCMC simulation
	J. A. Vrugt
(D) $C D$ $G H$ $K L$ $P N$ (D) $A B C D$ $E F G H$ $I J K L$ $M N O P$ The goal is to get the letters in	Title Page Abstract Introduction Conclusions References Tables Figures
ger value, and the resulting inverse ow the various steps of DREAM _(D)	Sion Paper Back Close Full Screen / Esc
	Printer-friendly Version Interactive Discussion Paper

(A)					(B)					(C)					(I
	Н	В	Ρ	Ν		Е	В	С	D		А	В	С	D	
	G	Μ	0	D		F	Α	G	н		F	Е	G	Н	
	I	Е	С	L		Ι	J	М	L		Ι	J	К	L	
	Κ	А	F	J		К	0	Ν	Ρ		М	0	Ρ	Ν	

Fig. 1. A 4 × 4 square tile puzzle with 16 different letters. The goal is to get the letters in order of the alphabet. Each letter is assigned a different integer value, and the resulting inverse problem is solved using $DREAM_{(D)}$. The panels **(B)–(D)** show the various steps of $DREAM_{(D)}$ to the target solution.





Fig. 2. Illustration of detailed balance: two-dimensional parameter estimation problem with $x_1 \in [0,7]$ and $x_2 \in [0,7]$. The color dots denote the starting points of the N = 5 different chains. The red square signifies the candidate point of the first chain and is created by selecting the blue chain as r_1 and green chain as r_2 . One step later, after moving to the red square, the backward jump has equal probability, as there is an equal chance of drawing r_1 (green) and r_2 (blue) in reversed order.

Discussion Paper	HES 8, 4025–4 DREAM _(D) - MCMC si	HESSD 8, 4025–4052, 2011 DREAM _(D) → discrete MCMC simulation						
Discus	J. A. '	Vrugt						
ssion P	Title	Page						
aper	Abstract	Introduction						
_	Conclusions	References						
Discu	Tables	Figures						
Issio	[◀	►I						
n Pap	•	•						
oer	Back	Close						
—	Full Scre	en / Esc						
Discussion	Printer-frien Interactive	Idly Version						
Paper	e	BY						

(A)

5 3

9 8

1 9

4 | 1 | 9

2 8

7 9

(B)

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	З	4	2	5	6	7
8	5	9	7	6	1	4	2	З
4	2	6	8	5	З	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

From: Wikipedia

Fig. 3. The daily sudoku: (A) Initial solution, and (B) Final solution. The black numbers were given, whereas the solution numbers are marked in red.







Discussion Paper



Fig. 5. Histogram of the DREAM_(D) derived marginal posterior distributions of the **(A)** I_{max} , **(B)** K_{s} , and **(C)** ϕ_{1} rainfall – runoff and error model parameters (in red). For convenience, only a few parameters are plotted. To benchmark the results of DREAM_(D) the bottom panel illustrates the results for DREAM (in blue), with the common assumption of a continuous parameter space.





Fig. 6. Two-dimensional scatter plots of the $DREAM_{(D)}$ derived posterior samples (top panel: red), and corresponding bivariate samples estimated with DREAM (bottom panel: blue). Only a few parameter pairs are shown as this is sufficient to illustrate the similarity of the sampled distributions.





Fig. 7. Simulated traces of the \hat{R} -statistic of Gelman and Rubin (Gelman and Rubin, 1992) using the **(A)** DREAM_(D) (top panel)(discrete sampling), and **(B)** DREAM (bottom panel) (continuous sampling) MCMC algorithms. Each parameter is coded with a different color. A similar convergence speed is observed for both algorithms.

