Environmental Modelling & Software 46 (2013) 292–298

Contents lists available at SciVerse ScienceDirect

Environmental Modelling & Software

journal homepage: www.elsevier.com/locate/envsoft

Short communication

Using a parallelized MCMC algorithm in R to identify appropriate likelihood functions for SWAT

J.F. Joseph [a,*], J.H.A. Guillaume [b]

[a] Department of Civil and Environmental Engineering, The University of Texas at San Antonio, One UTSA Circle, San Antonio, TX 78249, USA
[b] Integrated Catchment Assessment and Management (iCAM) Centre and National Centre for Groundwater Research and Training, Fenner School of Environment and Society, The Australian National University, Canberra ACT 2600, Australia

ARTICLE INFO

Article history:
Received 17 April 2012
Received in revised form
12 March 2013
Accepted 23 March 2013
Available online 20 April 2013

Keywords:
Likelihood function
Bayesian MCMC analysis
Parallel processing
SWAT
R
Uncertainty analysis
Parameter

ABSTRACT

Markov Chain Monte Carlo (MCMC) algorithms allow the analysis of parameter uncertainty. This analysis can inform the choice of appropriate likelihood functions, thereby advancing hydrologic modeling with improved parameter and quantity estimates and more reliable assessment of uncertainty. For long-running models, the Differential Evolution Adaptive Metropolis (DREAM) algorithm offers spectacular reductions in time required for MCMC analysis. This is partly due to multiple parameter sets being evaluated simultaneously. The ability to use this feature is hindered in models that have a large number of input files, such as SWAT. A conceptually simple, robust method for applying DREAM to SWAT in R is provided. The general approach is transferrable to any executable that reads input files. We provide this approach to reduce barriers to the use of MCMC algorithms and to promote the development of appropriate likelihood functions.

1. Introduction and motivation

The Soil and Water Assessment Tool (SWAT) is a basin-scale, semi-distributed, precipitation-runoff hydrologic model with open source code (Arnold et al., 1998). Over 1000 peer-reviewed publications utilize or review SWAT applications since its development in the early 1990s (http://swatmodel.tamu.edu/publications/peer-reviewed-publications/). Applications of SWAT include but are not limited to modeling water availability and water quality, stream channel erosion, plant growth, climate change impact, and comparing watershed management options.

SWAT contains non-linearities in its equations approximating the complexities of nature, and relies on input that inevitably contains errors. Since the work of Sorooshian and Dracup (1980) and Troutman (1983), the hydrologic community has been growing in awareness that, due to input and model inaccuracies, calibration and uncertainty analysis may lead to substantial bias in parameter and quantity estimates, and unreliable confidence intervals. As a theoretically sound remedy, Kavetski et al. (2006) propose a Markov chain Monte Carlo (MCMC) analysis within a Bayesian framework. Central to this analysis is the development of a likelihood function which realistically models residuals.

The MCMC approach originates in the seminal work by Metropolis et al. (1953) in the field of Physics and was generalized for other fields by Hastings (1970) as the Metropolis–Hastings (M–H) MCMC algorithm. MCMC has been growing in popularity (Diaconis, 2009). Yang et al. applied the traditional M–H MCMC algorithm to SWAT models of the Thur Basin in Switzerland (2007) and the Chaohe Basin in China (2008) with likelihood functions based on rigorous analysis of residuals.

Yet the application of MCMC approaches to SWAT has remained absent from the peer-reviewed literature since the work of Yang et al. (2007, 2008). As SWAT typically requires a fraction of a minute to several minutes per simulation, and traditional MCMC algorithms require a very high number of simulations for convergence, the application of MCMC to SWAT has required prohibitively long computational time. Before the algorithm converges, estimates of the probability distribution of the model parameters are likely to be unreliable. Yang et al. (2007, 2008) needed to use the Shuffled Complex Evolution-University of Arizona (SCE-UA) algorithm (Duan et al., 1992) to find an initial estimate of the global maximum as their starting point before applying M–H MCMC for

* Corresponding author.
E-mail address: john.joseph@utsa.edu (J.F. Joseph).

1364-8152/$ – see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.envsoft.2013.03.012

their SWAT models. With such difficulties in applying the Bayesian MCMC framework, progress has been lacking in the development of appropriate likelihood functions. Smith et al. (2010) have noted particular need for improvement in developing a likelihood function for even a single ephemeral catchment. Joseph (2011) identifies several challenges remaining to develop an appropriate likelihood function for two intermittent catchments.

The problem of the high number of simulations required in the M–H MCMC algorithm is certainly not limited to the SWAT model, and has precipitated various other MCMC algorithms that have appreciably accelerated movement of the MCMC chain towards equilibrium. Nonetheless, the problem of prohibitively high numbers of simulations has yet to be surmounted within the SWAT community. The MCMC algorithm that is most widely available to SWAT practitioners is the M–H MCMC algorithm itself, coded by Yang et al. (2007) for incorporation along with several other algorithms into the SWAT Calibration and Uncertainty Analysis Programs (SWAT-CUP) public domain software available through the Swiss Institute of Aquatic Science and Technology (http://www.eawag.ch/forschung/siam/software/swat/index_EN). SWAT user support group postings relating to the MCMC option in SWAT-CUP are few in number compared to those regarding the other (non-MCMC) algorithms, and these often refer merely to failed attempts at its usage.

The Delayed Rejection and Adaptive Metropolis (DRAM) MCMC algorithm is also now available to SWAT users through the R Flexible Modeling Environment (FME) package (Soetaert and Petzoldt, 2010). Wu and Liu (2012) allow SWAT to be called from R as a linked library and wrap it with FME, and this allows for the application of DRAM to SWAT. DRAM has been shown to have superior performance to that of M–H MCMC (Haario et al., 2006) for an algae growth model. However, a comparison (Vrugt et al., 2009) for the 13-parameter Sacramento-Soil Moisture Accounting (SAC-SMA) hydrologic model (Burnash, 1995) suggests that the improvement of DRAM over the more traditional MCMC is not sufficiently dramatic to attract SWAT users to DRAM, though they may be attracted to FME for its other algorithms and tools.

An MCMC algorithm that does offer a dramatic improvement over M–H MCMC is the Differential Evolution Adaptive Metropolis (DREAM) algorithm (Vrugt et al., 2009), When run in its parallelized version such as that of the MatLab code written by its developer, DREAM generates multiple Markov chains in parallel, which increases the parameter space explored and can speed up convergence. The cross-referencing reduces the number of simulations compared to that of DRAM and, as each chain may be assigned to a node, the total time for the MCMC analysis is spectacularly reduced compared to that of DRAM. DREAM has been run in parallel for a variety of models in the literature (e.g., He et al., 2011; Huisman et al., 2010). Vrugt et al. (2009) illustrate that DREAM identifies a global optimum RMSE more reliably than Random Walk Metropolis MCMC, DRAM, and SCE-UA, and requires fewer simulations. In addition, a near linear speed-up can often be achieved in methods that run the model simultaneously in parallel for parameter values in multiple chains (Vrugt et al., 2006), and MCMC methods may therefore be designed with this capability in mind (Laloy and Vrugt, 2012). Based on our own experience, with an 8-parameter calibration of a SWAT model of subbasin I of the Little River Experimental Watershed in Georgia, USA for an 8-thread desktop computer requiring 15 s per simulation, DREAM is able to reach completion in 12 h. For comparison, SCE-UA requires 18 h, does not provide estimates of posterior densities and therefore does not lend itself to the development of appropriate likelihood functions. Running DREAM without parallelization would have required 4 days, based on linear extrapolation, and was therefore considered impractical.

There are few examples of SWAT being run in parallel. Whittaker (2004) and Whittaker et al. (2010) have applied an unpublished approach to modify SWAT parameters in R by editing and recompiling the SWAT source code with a Linux operating system. Rouholahnejad et al. (2012) developed proprietary software in which a non-MCMC algorithm is parallelized for SWAT for Windows. Confesor and Whittaker (2007) and Zhang et al. (2012) have also applied parallel processing to SWAT. In any case, no application of a parallelized MCMC algorithm for SWAT has been found in the literature, let alone one that is open source and requires no proprietary software. As will be discussed in the following section, the general lack of parallel algorithm application to SWAT may be due to the high number of diverse input files that must be edited with each simulation.

The motivation of this paper is to present a reliable method for applying DREAM to SWAT in R, using tools to which practitioners have immediate access, while leaving them full freedom in specifying the likelihood function. We hope that this free access will aid in the development of appropriate likelihood functions, and thus more reliable confidence intervals and a correspondingly deeper scientific understanding of hydrology-related dynamics. The commonly used Windows 7 operating system was used, but a similar procedure should be possible in other operating systems.

Recently, a modified DREAM algorithm, MT-DREAM$_{(ZS)}$, has been developed and shown to be more efficient than DREAM in at least some contexts (Laloy and Vrugt, 2012), though not compared for SWAT. Additionally, unlike running DREAM in parallel, MT-DREAM$_{(ZS)}$ maintains detailed balance and reversibility throughout the execution process and is thus more correct theoretically (Laloy and Vrugt, 2012). The translation of MT-DREAM$_{(ZS)}$ from MATLAB to R is not yet achieved. However, the only obstacle to applying SWAT to DREAM in R has been the parallelizing aspect of DREAM, and, as this is the only obstacle anticipated for MT-DREAM$_{(ZS)}$ as well, we expect the method presented in this paper to be useful in applying MT-DREAM$_{(ZS)}$ to SWAT in R.

2. A conceptual description of the parallelization approach

DREAM runs a user-defined number of parallel chains and shares information among chains so that the proposal densities of randomized jumps are adjusted, resulting in accelerated movement of the MCMC chains towards equilibrium (Vrugt et al., 2009). Within each iteration, the model must be run for a set of parameters in each chain. These model runs are however entirely independent, which allows them to be run simultaneously, making use of the multiple processors, cores, or threads on a modern computer. To apply DREAM to SWAT in R, we propose establishing a separate folder for each chain, so that each chain has its own execution environment.

Our proposal springs from the fact that a SWAT project typically consists of hundreds or thousands of input parameter and data files to represent the multitude of hydrologic response units (HRUs) in the subbasins spatially distributed throughout the basin, and that these files must be edited and read before each simulation in an autocalibration process. Parameter values may vary in each HRU or subbasin, and, for model calibration and uncertainty analysis, a multiplier or addend or replacement will be associated with each default or best-guess value of the parameter. For example, a multiplier of 1.05 may be applied to the initially estimated permeabilities of all soil layers in each HRU, or to only the top three soil layers in the eastern half of the basin. These multipliers and addends and replacements themselves can thus be thought of as parameters, and throughout the remainder of this paper will be referred to as “adjustment parameters”. These adjustment parameters for the SWAT model include a channel roughness adjustment parameter, a soil hydraulic conductivity adjustment parameter, an adjustment parameter for the maximum volume of rainfall to be intercepted by the canopy, and several dozen others. A more