Ecological Modelling 360 (2017) 270–278

Contents lists available at ScienceDirect

Ecological Modelling

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

A Markov Chain Monte Carlo technique for parameter estimation and inference in pesticide fate and transport modeling

Julien Boulange*, Hirozumi Watanabe, Shinpei Akai

Tokyo University of Agriculture and Technology, Tokyo, Japan

ARTICLE INFO

Article history:
Received 10 January 2017
Received in revised form 14 July 2017
Accepted 14 July 2017
Available online 27 July 2017

Keywords:
Rice paddy
Pesticide fate and transport
Markov Chain Monte Carlo (MCMC)
Inverse modeling
PCPF-1 model

ABSTRACT

A Bayesian method involving Markov Chain Monte Carlo (MCMC) technique was implemented into a pesticide fate and transport model to estimate the best input parameter ranges while considering uncertainties included in both the observed pesticide concentrations and in the model.

The methodology used for integrating the MCMC technique into a pollutant fate and transport models was detailed. The uncertainties encompassed in the dissolution rate and in the adsorption coefficient of the herbicide mefenacet were greatly reduced by the MCMC simulations. In addition, an optimal set of input parameters extracted from the MCMC simulations accurately reproduced mefenacet concentrations in paddy water and paddy soil as compared to the original published dataset. Consequently, by simultaneously optimizing multiple parameters of environmental models and conducting uncertainty analysis, MCMC technique exhibits powerful capability for improving the reliability and accuracy of computer models.

The main strengths of the MCMC methodology are: (1) the consideration of uncertainties from both input parameters and observations and (2) the prior distributions of the input parameters which can be reformulate when additional knowledge is available.

1. Introduction

The contamination of water bodies as a result of pesticide used in agricultural fields, including rice paddies, was indicated in the literature (Iwafune et al., 2010). While monitoring is crucial to identify vulnerable areas to prioritize mitigation measures, it is costly and tedious. Simulation models are useful tools that, once validated, can help decision makers. Indeed, providing continuous information through simulation model and forecast assessments is a mandatory step for the registration of pesticide in Europe and in the U.S. (ter Horst et al., 2013). However, a major difficulty in predicting accurate pesticide concentrations arises from the uncertainties incorporated in the input parameters of computer models and in the observed or monitored datasets. A number of factors contribute to the uncertainty of forecasts, including parameter uncertainty, spatial variability, conceptual uncertainty, and boundary uncertainty (Görlitz et al., 2011; Hassan et al., 2009). In addition, since some processes involved in pesticide fate and transport are simplified or ignored by some computer models, the parameters related to such processes cannot be gather through direct measurements in the field, but can only be derived using a calibration procedure (Gallagher and Doherty, 2007; Vrugt et al., 2013). The effects of uncertainty on the predictions of a computer model are usually considered using analytical or numerical tools that spreads the uncertainty's attributes from the inputs to the final outputs of the model. A representative quantification of parameter's uncertainty remains nonetheless a challenge during the calibration and validation phases.

Model performance is typically quantitatively reported using statistical indices, such as the goodness of fit (R^2), which can be maximized by manual or automatic adjustments of the input parameters of the model (Hassan et al., 2009). Most optimization methods are however limited since they do not: (1) estimate the significance of the so-called optimal parameter set and (2) realistically quantify the uncertainty encompassed in the model (Kanso et al., 2006). Thus, it is arguable whether the so-called "optimized" input parameters established during the calibration of a model are appropriate. Indeed, due to non-uniqueness problems, two very different set of input parameters can often produce similar responses of the model (Abbaspour, 2015).

Markov Chain Monte Carlo (MCMC) techniques provide a captivating methodology to conduct the optimization of models while considering uncertainty encompassed in both input parameters and observations. However due their relative complexity, these

* Corresponding author.
E-mail address: boulange.julien@gmail.com (J. Boulange).

http://dx.doi.org/10.1016/j.ecolmodel.2017.07.011
0304-3800/

techniques have not yet been demonstrated in a pesticide fate and transport context. Thus, the aim of this research was to implement MCMC techniques to a rice pesticide fate and transport model, the PCPF-1 model. Specific objectives were defined as: (1) implement a MCMC algorithm (Metropolis-Hasting) into the PCPF-1 model, (2) develop the methodology to analyze the outputs of the MCMC simulation, and (3) validate the method by comparing the concentrations of the herbicide mefenacet predicted by the PCPF-1 model using the original validated dataset and the concentrations predicted using the optimized dataset generated through the MCMC output analysis.

2. Materials and method

2.1. PCPF-1 model

The PCPF-1 is a deterministic pesticide fate and transport model used to forecast pesticide concentrations in paddy water (PW) and in 1 cm deep paddy soil layer (PSL) (Watanabe and Takagi, 2000b; Watanabe et al., 2006). The model considers the transport of pesticide with paddy water (through irrigation, percolation, surface runoff and drainage) and the fate of pesticide by computing various degradation, desorption, dilution and dissolution processes. The best management practices for reducing pesticide discharge from paddy fields were identified by analyzing the model forecasts (Phong et al., 2011). The model is currently validated with six herbicides in Japan (Boulange et al., 2015; Takagi et al., 2012; Watanabe and Takagi, 2000a; Watanabe et al., 2006) and is pertinent for rice paddies in Europe (Karpouzas et al., 2006) and in California, USA (Luo et al., 2011). The inputs of the model consist of over 40 parameters which can be categorized into (1) climatic parameter, (2) water balance parameters, (3) soil parameters and (4) pesticides characteristics. A sensitivity and uncertainty analysis of the model stated that the accuracy of the forecasted pesticide concentrations in PW and PSL are extremely dependent on the accuracy of the input parameters (Boulange et al., 2012; Kondo et al., 2012). While improving the accuracy of some input parameters can be achieved with better experimental designs and a re-parametrization of the model, in most instances, the simulations need to be conducted using available data due to cost and time considerations (Malve et al., 2005). Four input parameters were included in the MCMC framework: the pesticide dissolution rate (k_{diss}), the first-order degradation rate of the pesticide in paddy soil (k_{bio}), the desorption rate of the pesticide (k_{des}), and the pesticide partitioning coefficient (k_d). All these parameters were reported to significantly impact the accuracy of the predicted pesticide concentrations (Boulange et al., 2012). A similar approach was adopted by Iizumi et al. (2009) who removed from the optimization process the most robust (certain) parameters even though some of them were empirical.

2.2. Bayesian inference

Bayesian inference prolong the use of probability theory by representing the uncertainty of a system (Malve et al., 2005; Reichert and Omlin, 1997). In a modeling application context, Bayesian inference is applied to estimate the values of θ unknown parameters of a model about which some prior information may be available (Gallagher and Doherty, 2007; Harmon and Challenor, 1997; Qian et al., 2003; Van Oijen et al., 2005). Using Bayesian inference, parameter uncertainty can be realistically implemented as the methodology distinguishes two sources of information for learning about unknown parameters: (1) pre-existing knowledge about parameters of a model, and (2) data collected via experimentation and observation (Bates and Campbell, 2001; Campbell et al., 1999; Hassan et al., 2009).

The prior probability distributions of the θ parameters are then updated to a new, posterior distributions, using the data collected via experimentations and observations (Hartig et al., 2011; Reichert and Omlin, 1997). The updating process is based on Bayes' theorem (Bayes, 1763):

$$P(\theta|d) = \frac{P(\theta)\cdot P(d|\theta)}{\int_{\theta} P(\theta)\cdot P(d|\theta)\cdot d\theta} \propto P(\theta)\cdot P(d|\theta) \quad (1)$$

where $P(\theta|d)$ is the posterior probability density of p model parameters $\theta = (\theta_1, \theta_2 \ldots \theta_p)$ given additional data $d = (d_1, d_2 \ldots d_n)$. $P(\theta)$ is the prior probability density of θ and captures all available knowledge about θ (Campbell et al., 1999; Paulo et al., 2005). $P(d|\theta)$ is the conditional probability density for the measured data d given the parameters θ. It is often referred to as the likelihood function and incorporates the statistical as well as the mechanistic relationships among the predictors and variables (Liu et al., 2008).

Typically, it is difficult to analytically summarize the posterior distributions which limits the practical implementation of Bayesian inference. However, an alternative approach is to use a Markov Chain Monte Carlo (MCMC) algorithm to obtain the numerical summarization of the posterior distribution (Liu et al., 2008). The process of collecting data, and thus acquiring knowledge about θ, is typically reflected in a reduction of uncertainty so that the posterior density will be more concentrated, more informative, than the prior density (Campbell et al., 1999).

2.3. Markov Chain Monte Carlo (MCMC) method

Bayesian inference often produce a posterior probability function that is difficult to compute using conventional numerical methods (Bates and Campbell, 2001). MCMC provides a general methodology of computing the posterior without having to perform integration over it (Harmon and Challenor, 1997; Qian et al., 2003). By generating a large enough sample from the posterior distribution, $P(\theta|d)$, any desired features (expectation value, median or maximum of the distribution) of the posterior distribution may be accurately summarized (Campbell et al., 1999; Chib and Greenberg, 1995; Hassan et al., 2009; Luke, 1994). Monte Carlo Markov Chain sampling methods involve three major steps: (1) formulation of the prior distributions of the selected parameters, (2) specification of the likelihood function, and (3) MCMC sampling to generate the posterior probability distributions of the selected parameters.

The Metropolis-Hastings (M−H) method describes a category of Monte Carlo methods which construct a Markov Chain in steps by randomly sampling from the posterior distribution described by Eq. (1) (Mathé and Novak, 2007; Wang and Chen, 2013). Although the M−H algorithm is not the most efficient Markov Chain sampler, it is extensively used in computer modeling applications due to the simplicity of its implementation, and its generality (Gallagher and Doherty, 2007; Kanso et al., 2006).

In practice, M−H algorithms begin by defining an initial value θ^0, of the model parameters θ. Then, by specifying a proposal density, $P(\theta^*|\theta^{t-1})$, a candidate value θ^* is selected, and the ratio R can be computed (Eq. (2)):

$$R = \frac{P(\theta^*|d)P(\theta^{t-1}|\theta^*)}{P(\theta^{t-1}|d)P(\theta^*|\theta^{t-1})} \quad (2)$$

where $P(\theta^*|d)$ and $P(\theta^{t-1}|d)$ were previously defined as the posterior probability densities of model parameters θ^* and θ^{t-1} given the data d, respectively (see Eq. (1)). The ratio R is compared to a random sample ζ taken from a uniform distribution (0,1). When $R > \zeta$, the candidate value θ^* is accepted as the next value in the sequence ($\theta^t = \theta^*$). In contrast, when $R < \zeta$, the candidate value θ^* is rejected as the next value in the sequence ($\theta^t = \theta^{t-1}$). The sequence