EPA Public Access

Author manuscript

Trans ASABE. Author manuscript; available in PMC 2018 November 08.

About author manuscripts

Submit a manuscript

Published in final edited form as:

Trans ASABE. 2017 ; 60(4): 1259–1269. doi:10.13031/trans.12179.

MOESHA: A Genetic Algorithm for Automatic Calibration and Estimation of Parameter Uncertainty and Sensitivity of Hydrologic Models

Bradley L. Barnhart [Postdoctoral Fellow],

Oak Ridge Institute for Science and Education (ORISE) Post-Doctoral Appointee, U.S. EPA National Health and Environmental Effects Research Laboratory, Western Ecology Division, Corvallis, Oregon

Keith A. Sawicz,

Oak Ridge Institute for Science and Education (ORISE) Post-Doctoral Appointee, U.S. EPA National Health and Environmental Effects Research Laboratory, Western Ecology Division, Corvallis, Oregon

Darren L. Ficklin [Assistant Professor], and

Department of Geography, Indiana University, Bloomington, Indiana

Gerald W. Whittaker [Researcher]

Department of Applied Economics, Oregon State University, Corvallis, Oregon.

Abstract

Characterization of the uncertainty and sensitivity of model parameters is an essential facet of hydrologic modeling. This article introduces the multi-objective evolutionary sensitivity handling algorithm (MOESHA) that combines input parameter uncertainty and sensitivity analyses with a genetic algorithm calibration routine to dynamically sample the parameter space. This novel algorithm serves as an alternative to traditional static space-sampling methods, such as stratified sampling or Latin hypercube sampling. In addition to calibrating model parameters to a hydrologic model, MOESHA determines the optimal distribution of model parameters that maximizes model robustness and minimizes error, and the results provide an estimate for model uncertainty due to the uncertainty in model parameters. Subsequently, we compare the model parameter distributions to the distribution of a dummy variable (i.e., a variable that does not affect model output) to differentiate between impactful (i.e., sensitive) and non-impactful parameters. In this way, an optimally calibrated model is produced, and estimations of model uncertainty as well as the relative impact of model parameters on model output (i.e., sensitivity) are determined. A case study using a single-cell hydrologic model (EXP-HYDRO) is used to test the method using river discharge data from the Dee River catchment in Wales. We compare the results of MOESHA with Sobol's global sensitivity analysis method and demonstrate that the algorithm is able to pinpoint non-impactful parameters, demonstrate the uncertainty of model results with respect to uncertainties in model parameters, and achieve excellent calibration results. A major drawback of

Corresponding author: Bradley Barnhart, U.S. EPA National Health and Environmental Effects Research Laboratory, Western Ecology Division, 200 SW 35th St., Corvallis, OR 97333; phone: 541-754-4517; barnhart.brad@epa.gov and bradleybarnhart@gmail.com.

the algorithm is that it is computationally expensive; therefore, parallelized methods should be used to reduce the computational burden.

Keywords

Genetic algorithm; Hydrologic modeling; Model calibration; Sensitivity analysis; Uncertainty

Hydrologic models are widely used to simulate surface and subsurface water flow within urban, agricultural, and mixed watersheds. A wide variety of hydrologic models attempt to simulate individual hydrologic processes and describe how these processes are impacted by climate and future climate change, land characteristics (e.g., topography, soil types), and human behavior (e.g., land use change, water resources management). However, regardless of the intended application, nearly all hydrologic models require calibration, and they benefit from a description of model uncertainty with respect to, among other things, model parameter uncertainty. Calibration is a process that tunes the model parameters to match observed data, such as streamflow measurements, soil moisture, and subsurface/groundwater flow. Manual calibration is tedious and impractical for large, heterogeneous watersheds with many parameters; therefore, numerous calibration methods have been developed to adjust tunable model parameters within user-specified parameter limits to match simulated output to observed hydrologic data (Abbaspour et al., 2015; Gupta et al., 1999; Larsen et al., 2016; Petrucci and Bonhomme, 2014; Vrugt et al., 2003).

To automate the calibration process, evolutionary algorithms, specifically genetic algorithms, have been widely used for hydrologic models (Tigkas et al., 2016; Wang, 1991; Yapo et al., 1998). Genetic algorithms mimic biological evolution by evaluating a large number of input parameter possibilities (called individuals). The fittest individuals (i.e., the parameter sets that produce solutions that best match the observed data) are computationally combined (i.e., mated) to produce new parameter sets by using crossover and mutation operators while also maintaining diversity within the population. Through evolving individuals over many generations, the overall goal of a genetic algorithm calibration routine is to find the optimal set of input parameters that best match the simulated results with the observed data.

All automatic calibration routines have their own merits, and their utility is largely dependent on the complexity of both the hydrologic model and the watershed, as well as the computing resources. For example, Arsenault et al. (2014) compared ten optimization algorithms for ten different basin test cases with different levels of hydrologic model complexity. They found that optimization methods varied in their effectiveness and depended on the model dimensionality and complexity. In addition, as the number of objectives increases, the required sorting calculations become more complicated. The most widely used genetic algorithm is the non-dominated sorting genetic algorithm II (NSGA-II) (Deb et al., 2002), which sorts individuals using a domination scheme in which one individual is not dominated if it is better than all other individuals in all objectives. NSGA-II maintains diversity in the population through a crowding distance parameter, which specifies

the necessary distance between individuals in objective space. Note that NSGA-III was recently developed for many-objective problems (Deb and Jain, 2014).

Other commonly used evolutionary algorithms include the strength Pareto evolutionary algorithm (SPEA2) (Zitzler et al., 2002), the eNSGA-II algorithm (Laumanns et al., 2002), and the many variants of particle swarm optimization (PSO) (Bratton and Kennedy, 2007; Kennedy and Eberhart, 1995). Overall, the literature regarding optimization algorithms, including evolutionary algorithms and genetic algorithms for many-objective problems and conflicting objectives, is extremely rich and diverse. For recent comprehensive reviews of multi-objective evolutionary algorithms related to hydrologic modeling, see Reed et al. (2013) and Maier et al. (2014).

In addition to automatically performing model calibration, many methods exist to characterize the sensitivity of model performance to model parameters and to determine the overall uncertainty of hydrologic models. It is important to define and distinguish between uncertainty and sensitivity analyses as they apply to hydrologic models. This study is restricted to uncertainty and sensitivity analyses related specifically to model parameters and thus does not include other uncertainties, such as model structure or measurement. Within this scope, sensitivity analysis determines the relative degrees to which model performance changes given a variation in model parameters. Uncertainty analyses extend sensitivity by quantifying the relative changes in a model's performance given variations or uncertainties in the model's parameters. Both uncertainty and sensitivity analyses are concerned with how changes in model parameters affect changes in model output (e.g., discharge, nutrient loadings, stream temperature). However, sensitivity analysis is concerned with relative changes due to changes in model parameters in order to rank impactful and non-impactful parameters, while uncertainty analysis attempts to quantify the influence of parameter uncertainties on the resulting model output.

Broadly speaking, sensitivity analyses can be categorized as local sensitivity analysis (LSA) or global sensitivity analysis (GSA). LSA typically consists of simple methods that use one-at-a-time (OAT) perturbation (Hamby, 1994), which changes a single parameter at a time over a sequence of steps and surveys the hydrologic output. However, OAT has been found to be inefficient and inadequate to capture higher-order interaction effects ubiquitous within hydrologic models (Saltelli and Annoni, 2010). Hydrologic models are complex, dynamic representations of the real world and contain large numbers of parameters and interactions that are not captured with a simple method. Therefore, refinements to the local OAT method take the form of GSA methods. One type of GSA divides the entire feasible model parameter space into a subset of feasible regions, and parameter set samples are determined and tested throughout the entire parameter space. For example, stratified sampling (Castaings et al., 2012) seeks to divide the entire parameter space into a number of samples that are randomly distributed. Another method is Latin hypercube sampling (Manache and Melching, 2008), in which each parameter's feasible range is divided into a number of equal-probability subsets from which samples are drawn to create testable parameter sets. Sobol's variance-based method is one such Latin hypercube sampling method that sequentially samples the parameter space in progressively finer detail according to the number of samples (Tang et al., 2007). Other derivative-based methods exist and stand as

alternatives to the Sobol variance-based method (Morris, 1991). These methods are able to demonstrate elementary effects from sample means and standard deviations from a random sample. Aside from these more traditional GSA approaches, some additional approaches include VARS (Razavi and Gupta, 2016a, 2016b), which tries to address the limitations of the Morris and Sobol methods by deriving variograms and covariograms of the parameter space. As with evolutionary algorithms, the literature regarding local, regional, and global sensitivity analyses is diverse and extensive, and we have presented only a small portion of the available methods. For further reviews of sensitivity analyses, see Song et al. (2015) and Hamby (1994).

Uncertainty analysis considers the impact of input parameters further. It determines the relative changes in a model output given uncertainty in the input parameters. For this study, we limit ourselves to discussing the uncertainty that results from changes in model parameters. We leave a discussion of other kinds of uncertainty (e.g., input data uncertainty or mechanistic or structural uncertainty) for other investigations. We also note that uncertainty analyses within this scope strive to demonstrate the degree of uncertainty a model exhibits in its output given uncertainty in the input parameters. There are vast amounts of literature related to measuring the uncertainty in modeling hydrologic components within hydrologic models given uncertain input parameters (e.g., Guzman, 2015). Generally, these methods use Monte Carlo statistical simulations (Zhang, 2016; Blasone, 2008) as well as Bayesian analysis to determine the relative uncertainties in model performance given uncertainties in model parameters (Dong, 2013).

The aim of this article is to present an algorithm that combines the advantages of three separate tools (i.e., sensitivity analysis, uncertainty analysis, and calibration) using a genetic algorithm. This algorithm, the multi-objective evolutionary sensitivity handling algorithm (MOESHA), is essentially an all-inclusive tool that incorporates parameter sensitivity and uncertainty within an optimization calibration routine. Table 1 shows this incorporation concept, in which the new algorithm combines benefits from all three types of analyses.

Previous studies have typically conducted calibration and sensitivity analyses as separate steps when constructing a calibrated model, and the sensitivity analyses inform which parameters to tune for calibration. However, MOESHA combines aspects from three different tools (i.e., sensitivity analysis, uncertainty analysis, and calibration) into a single framework. MOESHA requires very little user guidance. The user specifies which parameters to include in the process and the ranges in which they are allowed to vary. MOESHA then provides a calibrated model and specifies which of the parameters are impactful (i.e., model output is sensitive to their fluctuations). MOESHA also provides a distribution of model performances that result from a distribution of model parameters, providing an estimate of uncertainty due to uncertain model parameters.

MOESHA optimizes objectives of model uncertainty and robustness given distributions of input parameters. Therefore, in a single framework, MOESHA provides the utility of three separate tools: sensitivity analysis, uncertainty analysis, and calibration. Users obtain a calibrated model that includes optimal distributions of input parameters that maximize model robustness and minimize model error. Therefore, MOESHA can be used to provide

users with a simpler method for calibration and sensitivity analyses using hydrologic models.

Table 2 compares MOESHA with a few other all-inclusive algorithms that are available. While these algorithms provide outcomes similar to MOESHA (i.e., calibration, sensitivity, and uncertainty analyses), MOESHA distinguishes itself by using a genetic algorithm that dynamically searches the parameter space, as opposed to the static sampling that other methods rely on. Rather than testing a portion of the parameter space in a static manner (e.g., in Latin hypercube sampling or uniform parameter sampling) to determine model sensitivity to different model inputs, MOESHA uses a genetic algorithm to dynamically search for the representative distribution of input parameters within the parameter space.

To clarify this point, note that static sampling algorithms sample the entire input parameter space and produce many input parameter sets to be tested (which are representative of the entire parameter space). Each parameter set is then tested, one at a time, to determine the parameters sets that produce the most optimal solutions. The algorithms and frameworks listed in table 2, apart from MOESHA, use these static methods. For example, SWAT-CUP and PEST use Latin Hypercube sampling, whereas MOUSE allows several static sampling methods, including random, Latin hypercube, multi-point random, and Halton and Sobol sequence sampling methods (Ascough et al., 2014

In contrast, MOESHA uses a genetic algorithm that dynamically searches the input parameter space. We specify the term “dynamic” because the algorithm produces each new parameter set to be tested at each generation. Rather than sampling the entire space, the genetic algorithm searches different regions of the parameter space depending on the outcomes of other solutions that it has tested.

In the next section, we describe the general methodology for the calibration and uncertainty tool. We also describe the hydrologic model used to test the algorithm and the genetic algorithm framework used for the optimization. We then demonstrate the utility of the algorithm to calibrate a hydrologic model for river discharge in the Dee River catchment in Wales. Finally, we discuss how this algorithm can be used with other hydrologic models, and we discuss avenues for future research.

Methods

Overview of MOESHA

MOESHA is an optimization framework that combines the ability of a genetic algorithm with uncertainty and sensitivity analyses to achieve a calibrated model while quantifying uncertainties due to variations in model parameters. Through this process, it is also able to differentiate non-impactful parameters from impactful parameters. First, consider a black-box model, as shown in Figure 1, that converts a single input parameter x into a time series of outputs Y_{sim} , which are then converted to a closeness metric (in this case, the Kling-Gupta efficiency (KGE) criterion, which will be defined below) that depends on simulated and observed data.

In the case of hydrologic models, the simulated model output Y_{sim} is typically the discharge at the watershed outlet, which is subsequently compared to the streamflow observed at a hydrologic gauge. Multiple closeness metrics can be used to determine the goodness-of-fit for simulated hydrologic data. Two of the most common are the Nash-Sutcliffe efficiency (NSE) (Nash and Sutcliffe, 1970) criterion and the Kling-Gupta efficiency (KGE) criterion (Gupta et al., 2009), which both range from $-\infty$ to 1. Both metrics give values closer to 1 to designate a close match between simulated and observed data. We used the KGE criterion as the metric for our example (Gupta et al., 2009); however, our methodology is easily extendable to include different efficiency criteria or closeness metrics as well as additional objective functions. For example, Willmott et al. (2015) assessed three dimensionless measures that could be used to quantify model performance as alternatives to NSE and KGE. These could be included in our algorithm if a user so desired.

MOESHA currently assumes an initial normal distribution of a single input parameter x with a mean value μ_x and a standard deviation σ_x . Normal distributions were assumed in this application for simplicity, but the algorithm can be generalized to include other input parameter distributions, such as uniform distributions. Next, we take M random draws from the distribution of x , as shown in figure 2, and we obtain M different input parameters that can be inserted into the model.

The number M should be sufficiently large to represent the underlying parameter distribution, but it should not be overly large because further draws require model runs and therefore present a computational burden. The outputs result in M different realizations of the outputs and subsequent KGE values. To obtain single objective values from the M different realizations, we calculate the mean value μ_{KGE} and standard deviation σ_{KGE} .

The uncertainty of the model can therefore be quantified by the distribution (or, more quantitatively, the variance) of the resulting KGE values. The mean represents the model closeness with respect to the observations, and the variance (or standard deviation) represents the model robustness, or the ability to produce similar results given different input parameters. For this case, we assume that all other inputs (e.g., inputs of weather, land use, soil types, slopes) remain constant.

The next step is for the algorithm to find the optimal input parameter distributions that maximize the model closeness to the observations and minimize the variance of the outputs. Briefly, this is accomplished by testing many possible μ_x and σ_x parameters and by examining many outputs of μ_{KGE} and σ_{KGE} . As described above, the optimization literature is highly sophisticated and diverse and includes numerous methods to achieve objectives while testing enormous numbers of inputs. For this article, we use a genetic algorithm (described in more detail below) to search for the optimal distribution of model input parameters μ_x and σ_x .

Note that genetic algorithms have been demonstrated to be extremely efficient at finding global optima to multi-objective problems. The algorithm dynamically searches the parameter space to find optimal solutions, it uses crossover operations to create new solutions based on the best solutions found in the previous step, and it uses mutation

operations to avoid finding only local optima and to ensure that global solutions are found (Deb et al., 2002; Fortin et al., 2012).

An overview of the genetic algorithm is shown in figure 3. The user initially sets the μ_{min} and μ_{max} values as well as the σ_{min} and σ_{max} values for each input parameter that will be calibrated in the model. For example, if the hydrologic model has four input parameters that need to be calibrated, the genetic algorithm will attempt to optimize eight parameters.

The distribution parameters of model inputs are optimized rather than the parameters themselves. This example is demonstrated for a single input parameter x ; however, in practice, multiple input parameters are used. We note that the algorithm essentially doubles the number of parameters that need to be analyzed, thus increasing the parameter search space. However, previous studies have used genetic algorithms to calibrate hundreds (Barnhart et al., 2014) to thousands (Whittaker et al., 2010) of parameters and have shown that they achieve convergence; therefore, we do not expect convergence to be an issue for small to medium applications.

The genetic algorithm chooses solutions that minimize the model output variance to seek robust solutions while maximizing model closeness metrics to the observed data. The user must also specify the objectives for the genetic algorithm. To formulate our objectives, we note that an ideal model is one that produces low errors and robust solutions. Therefore, it is our goal to maximize μ_{KGE} and minimize σ_{KGE} . In addition to these two objectives, we included a third objective to maximize the standard deviation of the input parameter distribution (or the sum if multiple parameters are to be calibrated), that is, $\max(\sigma_x)$. This may seem counterintuitive because it seems that we are putting selective pressure to look for large variations in input parameters. However, this ensures that our optimized solutions will include the largest σ_x possible that also simultaneously optimizes the first two objectives, i.e., maximizing model fit and robustness. This is important to differentiate between impactful and non-impactful parameters. An impactful parameter, which is a parameter to which the model is sensitive, will be characterized by a narrow distribution, while a non-impactful parameter will have a wider distribution or variance. A dummy variable, which does not impact the model's output, is included in the optimization to demonstrate the utility of this concept. The MOESHA algorithm output provides distributions of input parameters and distributions of model output. The algorithm was run for 1000 generations, and no additional stopping criteria were used. For 64 individuals tested at each generation 1000 times, this amounts to 64 million model evaluations. We note that no additional solutions were found after, say, several hundreds of generations; however, this criterion can and should be extended for larger-scale studies where a larger number of input parameters are analyzed.

Hydrologic Model

The EXP-HYDRO model (Patil and Stieglitz, 2014) was used to test our algorithm. EXP-HYDRO is a lumped rainfall-runoff model that can simulate daily streamflow from daily precipitation (P) and mean daily temperature (T_d) by simultaneously solving the following coupled differential equations at each time step:

$$\frac{dS_{Snow}}{dt} = P_{snow} - Q_{melt} \quad (1)$$

$$\frac{dS}{dt} = P_{rain} + Q_{melt} - ET - Q_{bucket} - Q_{spill} \quad (2)$$

where S is the amount of water stored in the catchment, and S_{snow} is the amount of water stored in the so-called snow accumulation bucket. P_{rain} and P_{snow} are the rainfall and snowfall amounts (mm d^{-1}), and ET is the actual evapotranspiration (mm d^{-1}) from the so-called catchment bucket. These amounts are measured, observed data and are included as inputs into the model. The flows are then characterized in mm d^{-1} as snowmelt (Q_{melt}), subsurface runoff (Q_{sub}), and capacity-excess surface runoff (Q_{runoff}). A conceptual diagram of the model is shown in figure 4.

As shown in figure 4, EXP-HYDRO is a simple single-cell, single-layer hydrologic model. The model uses inputs of precipitation, relative humidity, evapotranspiration, and air temperature and determines subsequent snowmelt and surface and subsurface runoff for a location. The model relies on six parameters for calibration to a single observed hydrologic gauge. These six parameters include a rate of decline in runoff (f), maximum storage of watershed bucket (S_{max}), maximum subsurface runoff at full bucket (Q_{max}), degree day snow factor (D_f), melt temperature (T_{max}), and snow threshold temperature (T_{thresh}). EXP-HYDRO is a simple conceptual model with low computation time, which is ideal for testing the applicability and usefulness of novel approaches because many simulations can be performed.

The hydrologic model was run for the Dee River catchment in Wales. This catchment was chosen because it was used as the example watershed in the software documentation of the EXP-HYDRO model (Patil and Stieglitz, 2014; Patil et al., 2014). Precipitation, temperature, relative humidity, and evapotranspiration data were provided, as well as discharge data used to compare with the simulated discharge. EXP-HYDRO has also been used in watersheds throughout the U.S. of varying sizes and with a diverse range of climatic and physical attributes (Patil and Stieglitz, 2015). For more information on the EXP-HYDRO model and the example watershed formulation, see Patil and Stieglitz (2014) and Patil et al. (2014).

Genetic Algorithm

We used the Distributed Evolutionary Algorithms in Python (DEAP) (Fortin et al., 2012) for the application of our genetic algorithm. DEAP is a flexible framework that incorporates several tools that can be used to construct different evolutionary algorithms. It is able to accommodate a large number of crossover, mutation, and variation parameters. For this application, we used the traditional ($\mu + \lambda$) evolutionary algorithm with NSGA-II sorting. This creates a parent population of μ individuals and subsequently generates λ offspring within the population. Each individual represents a set of input parameters to be tested by

running the hydrologic model. After the μ and λ individuals are used within the model, the results (in this case, KGE values) are sorted, and the μ best individuals from the $(\mu + \lambda)$ population are selected for the next generation. NSGA-II was used to sort and select the best individuals.

The DEAP framework offers many different variational parameters, including different crossover and mutation operators. These operators are the computational methods for creating a new generation of individuals to be tested with the hydrologic model. For this application, we used the ‘cxBlend’ operator for crossover (with the α parameter set to 0.1), and the Gaussian mutation operator (with $\mu = 0$ and $\sigma = 1$). The mutation probability was set to the inverse of the number of variables included in the genetic algorithm (1/14 or 0.07). In addition, the probability for a crossover to occur was set to 0.95. DEAP is highly integrated with the SCOOP library in Python (Nilsen and Prcovic, 1998), which enables parallel implementation of the algorithm to reduce computational burden. The SCOOP parallelization allows hydrologic model simulations to be performed concurrently, and it works seamlessly with multi-processor personal computers or with large-scale superclusters. For more details regarding DEAP and SCOOP, refer to their online documentation. For this application, we used 64 parallel nodes on a Dell Server R820 running 64-bit Windows and equipped with four Intel Xeon processors (E5–4650 at 2.70 GHz).

Sobol Sensitivity Analysis

For validating MOESHA, we used Sobol GSA (Sobol and Levitan, 1999; Sobol, 2001) as a benchmark method. The GSA algorithm generated 20,000 samples to ensure a robust parameter sensitivity result. Therefore, MOESHA is said to be validated if its parameter sensitivity results correspond with the variance-based approach inherent in Sobol GSA. The GSA was carried out using ‘SALib’, a sensitivity analysis library built specifically for the Python programming language.

Results

Running MOESHA on the Dee River catchment in Wales took approximately 6.4 min per generation. Each generation included 64,000 model runs (i.e., 64 parallel nodes running 1,000 runs for each generation). We ran MOESHA for 1,000 generations, and the algorithm took approximately 4.4 days to complete. This culminated in 64 million model runs. After testing MOESHA on the Dee River catchment in Wales, we plotted the results for our first two objectives (i.e., to maximize the mean KGE and minimize the standard deviation of KGE), as shown in figure 5. Note that while this is a multi-objective optimization problem, the objectives μ_{KGE} and σ_{KGE} are not conflicting objectives. This is shown in figure 5, where higher μ_{KGE} values generally correspond to lower σ_{KGE} values. Because these objectives are not conflicting, we do not expect to see a traditional Pareto frontier that exhibits trade-offs between the two objectives. Instead, the “best” or most optimal solutions are those that have the highest μ_{KGE} values and the lowest σ_{KGE} values.

The most optimal solutions give μ_{KGE} values that are greater than 0.9 and σ_{KGE} values that are less than 0.001. To further evaluate these solutions, we chose the 50 best individuals that maximized μ_{KGE} values, and we constructed parameter sets from these μ_{KGE} and σ_{KGE}

values to test within the EXP-HYDRO model. The decision to select 50 individuals is arbitrary, and the results (figure 6) are consistent regardless of the exact number of individuals chosen. The distributions of the input parameter sets and the KGE outputs are shown in figure 6. Recall that the EXP-HYDRO parameters represent the following physical mechanisms of the model: p1 = rate of decline in runoff (f), p2 = maximum storage of watershed bucket (S_{max}), p3 = maximum subsurface runoff at full bucket (Q_{max}), p4 = degree day snow factor (D_f), p5 = melt temperature (T_{max}), and p6 = snow threshold temperature (T_{min}).

The variance or “spread” of the distributions indicates whether a parameter is impactful to the model or, in other words, whether the model is sensitive that parameter. The x ranges for each of the variables were set within the limits of the calibration recommendations for the hydrologic model. Note that the dummy variable shows that all 50 of the input parameter distributions give a very wide distribution. This indicates that the parameter can be any value within that range and still produce the same accurate and robust model output. Conversely, the rate of decline in runoff (p1) shows a very narrow distribution and therefore highly impacts the resultant runoff of the model. If other values are used that are outside of this narrow distribution, they will greatly change the resulting output. Therefore, the model is extremely sensitive to the rate of decline in runoff, and p1 can be considered an impactful parameter. Parameters p2 and p3, which represent the maximum storage and maximum allowed subsurface runoff, are less impactful, whereas the snow factor (p4), melt temperature (p5), and snow threshold temperature (p6) as well as the dummy parameter do not impact the model output, i.e., the model is not sensitive to their variations. Overall, these results suggest that variations in snow dynamics will not impact the hydrology of the Dee River catchment.

In addition to graphically comparing the distributions of the parameters with the dummy parameter, we compared the standard deviations for each parameter distribution with the standard deviation for the dummy parameter distribution using the 50 parameter sets shown in figure 6. To normalize these calculations for accurate comparison, we divided the standard deviations by the allowed range of each parameter (i.e., the difference between the minimum and maximum allowed parameter values). These ranges, as shown in the plots in figure 6, are 0.1, 1400, 40, 5, 3, 3, and 1 for the six parameters and the dummy variable, respectively. The calculated comparison metrics were then 0.02, 0.05, 0.04, 0.22, 0.43, and 0.41 for the six parameter distributions and 0.34 for the dummy parameter, respectively. These results quantify what is already shown in figure 6. That is, parameters 1, 2, and 3 have much smaller standard deviations and are therefore impactful, whereas parameters 4, 5, and 6 have distributions that more closely resemble that of the dummy parameter and are therefore not impactful for the model output.

To further validate the algorithm, we compared our results to Sobol GSA. The results of the Sobol analysis are shown in table 3. This method produces the first-order sensitivity (influence of the parameter alone), second-order sensitivity (interactive influence between parameters), and total sensitivity. As expected from the theory behind the algorithm, the results of MOESHA are most comparable to the total parameter sensitivity. In agreement with the results of MOESHA, the Sobol analysis also found that the most impactful model

parameter was p1, with 84% of the total model sensitivity, and p2 was found to be moderately sensitive, with 13% of the total model sensitivity. The majority of parameter sensitivity displayed by p2 was found in the interactive effect with other parameters, instead of its first-order sensitivity. The Sobol analysis also found that parameters p3, p4, p5, and p6 did not impact the model results, and therefore the model was considered insensitive to their variations. In fact, MOESHA found that p3 was a semi-impactful parameter, and the Sobol GSA showed that while p3 did not have first-order sensitivity, it showed a small amount of interactive (i.e., higher order) sensitivity. Therefore, we suggest that p3 should be considered a semi-impactful parameter and should not be removed from further calibration work. Conversely, the dummy parameter was shown to have exactly 0% contribution to the overall model sensitivity using Sobol GSA, and MOESHA also showed that it did not impact the model's performance, as expected. The addition of all total sensitivities, as opposed to the relative sensitivities of each parameter, suggests that the model performance is moderate, in large part due to p1 and p2.

Overall, MOESHA generally agrees with the results of Sobol GSA for the impactful parameters, non-impactful parameters, and the dummy variable. The advantage of MOESHA over a simple Sobol GSA method is that MOESHA provides a calibrated model as well as an estimate of the distribution of model performance that varies given alternative distributions of model parameters, as shown in figure 6. MOESHA acts as a sensitivity analysis tool and also provides the advantages of uncertainty analysis (with respect to model parameters) and calibration, all within a single framework.

Discussion and Conclusions

We presented an algorithm called MOESHA to simultaneously calibrate a hydrologic model and determine the impacts of input parameter variation on model uncertainty. While numerous calibration and sensitivity handling methods exist, few combine both calibration and model uncertainty within a single framework that uses a genetic algorithm optimization routine. Our methodology uses NSGA-II within the DEAP framework and leverages parallelization using SCOOP within Python. The parallelization aspect of this method is especially important because the key drawback of this method is the computational burden required. In addition to the tens of thousands of individuals simulated within a traditional evolutionary algorithm, we require an additional M -fold (in this case, 1000) increase in computational burden. One avenue for future research would be to reduce the number of draws from 1000 to, for example, 500 or 100 and to analyze the results. At some point, the limited number of draws will prove too few to describe the underlying distribution of the model parameters. However, increasing the number of draws increases the computational burden. SCOOP parallelization ensures that the number of individuals that can be simultaneously tested scales with the number of processors available; however, for large watersheds, this remains an obstacle. With increased accessibility to high-performance computing clusters, we expect that this obstacle will dissolve in the near future.

Another drawback of our methodology was that we assumed normal distributions for our input parameters. One simple extension could be to include uniform distributions of input parameters with variable distribution widths. However, non-parametric representations of

input distributions would be ideal, such as the distributions used in sophisticated Bayesian methods that use Markov chain Monte Carlo (MCMC) methods. Apart from the shape of the input parameter distributions, another problem that may impact the utility of this method is if the so-called equifinality condition is met, in which multiple input parameter sets achieve similar or equal simulated outputs (Beven, 2006; Ficklin and Barnhart, 2014). Further research should be pursued to determine whether the inclusion of multiple distributions for a single input parameter would provide further flexibility when determining the optimal sets of parameters to achieve closeness to the observed data.

Apart from these caveats, MOESHA has been shown to be an effective tool for calibration and calculation of model uncertainty related to changes in input parameters. This extends the utility of genetic algorithms for calibration and calculation of model uncertainty to provide more reliable model outputs from hydrologic models. It also provides a dynamic method for sensitivity analysis in general, whose main alternatives typically rely on deconstructing the parameter space into a finite number of testable subregions. The use of an evolutionary algorithm enables the global parameter space to be adequately searched dynamically while ensuring that local parameter space estimates are tested as well.

While the algorithm has proven useful for the single-cell, single-layer version of EXP-HYDRO, more sophisticated hydrologic models that incorporate watershed-scale dynamics should be used with the algorithm to test its scalability. In particular, VELMA (Visualizing Ecosystem Land Management Assessments) is a hydrologic and biogeochemical environmental watershed model that uses a grid-based approach and may be especially well-suited for this algorithm (Abdelnour et al., 2011, 2013). Testing should also be performed using semi-distributed environmental models such as SWAT (Soil and Water Assessment Tool) and HSPF (Hydrologic Simulation Program - Fortran). Because MOESHA is a computationally expensive algorithm, the use of high-performance computing clusters that have the capability of running many concurrent nodes is encouraged to reduce the computational burden of the algorithm. Finally, additional research should be pursued to combine genetic and evolutionary calibration algorithms to other forms of sensitivity and uncertainty analyses to provide comprehensive tools for sensitivity, uncertainty, and calibration analyses of hydrologic models.

Acknowledgements

The views expressed in this article are those of the authors and do not necessarily reflect the views or policies of the U.S. Environmental Protection Agency. The manuscript has been subjected to agency review and has been approved for publication. Mention of trade names or commercial products does not constitute endorsement or recommendation for use.

References

- Abbaspour KC (2011). SWAT-CUP4: SWAT calibration and uncertainty programs: A user manual Dübendorf: Switzerland: Swiss Federal Institute of Aquatic Science and Technology (Eawag).
- Abbaspour KC, Rouholahnejad E, Vaghefi S, Srinivasan R, Yang H, & Klove B (2015). A continental-scale hydrology and water quality model for Europe: Calibration and uncertainty of a high-resolution large-scale SWAT model. *J. Hydrol*, 524, 733–752. 10.1016/j.jhydrol.2015.03.027

- Abdelnour A, McKane RB, Stieglitz M, Pan F, & Cheng Y (2013). Effects of harvest on carbon and nitrogen dynamics in a Pacific Northwest forest catchment. *Water Resour. Res.*, 49(3), 1292–1313. 10.1029/2012WR012994
- Abdelnour A, Stieglitz M, Pan F, & McKane R (2011). Catchment hydrological responses to forest harvest amount and spatial pattern. *Water Resour. Res.*, 47(9), W09521 10.1029/2010WR010165
- Arsenault R, Poulin A, Cote P, & Brissette F (2014). Comparison of stochastic optimization algorithms in hydrological model calibration. *J. Hydrol. Eng.*, 19(7), 1374–1384. 10.1061/(ASCE)HE.1943-5584.0000938
- Ascoug II JC, Fischer C, Lighthart NP, David O, Green TR, & Kralisch S (2014). Overview and application of the Model Optimization, Uncertainty, and SEnsitivity Analysis (MOUSE) toolbox. Environmental Modeling International Conference Proceedings Available: http://www.iemss.org/sites/iemss2014/papers/iemss2014_submission_358.pdf.
- Barnhart BL, Whittaker GW, & Ficklin DL (2014). Improved stream temperature simulations in SWAT using NSGA-II for automatic multi-site calibration. *Trans. ASABE*, 57(2), 517–530. <https://doi.org/10.13031/trans.57.10472>
- Beven K (2006). A manifesto for the equifinality thesis. *J. Hydrol.*, 320(1), 18–36. 10.1016/j.jhydrol.2005.07.007
- Blasone RS, Madsen H & Rosbjerg D (2008). Uncertainty assessment of integrated distributed hydrological models using GLUE with Markov chain Monte Carlo sampling. *J. Hydrol.*, 353(1), 18–32. 10.1016/j.jhydrol.2007.12.026.
- Bratton D, & Kennedy J (2007). Defining a standard for particle swarm optimization Proc. IEEE Swarm Intelligence Symp. (pp. 120–127). Piscataway, NJ: IEEE 10.1109/SIS.2007.368035
- Castaigns W, Borgonovo E, Morris MD, & Tarantola S (2012). Sampling strategies in density-based sensitivity analysis. *Environ. Model. Software*, 38, 13–26. 10.1016/j.envsoft.2012.04.017
- Deb K, & Jain H (2014). An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach: Part I. Solving problems with box constraints. *IEEE Trans. Evolutionary Computation*, 18(4), 577–601. 10.1109/TEVC.2013.2281535
- Deb K, Pratap A, Agarwal S, & Meyarivan T (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evolutionary Computation*, 6(2), 182–197. 10.1109/4235.996017
- Doherty J, & Johnston JM (2003). Methodologies for calibration and predictive analysis of a watershed model. *JAWRA*, 39(2), 251–265. 10.1111/j.1752-1688.2003.tb04381.x
- Dong L, Xiong L & Yu KX (2013). Uncertainty analysis of multiple hydrologic models using the Bayesian model averaging method. *J. Appl. Math* 2013 10.1155/2013/346045
- Ficklin DL, & Barnhart BL (2014). SWAT hydrologic model parameter uncertainty and its implications for hydroclimatic projections in snowmelt-dependent watersheds. *J. Hydrol.*, 519, Part B, 2081–2090. 10.1016/j.jhydrol.2014.09.082
- Fortin F-A, Rainville F-MD, Gardner M-A, Parizeau M, & Gagne C (2012). DEAP: Evolutionary algorithms made easy. *J. Machine Learning Res.*, 13, 2171–2175.
- Gupta HV, Kling H, Yilmaz KK, & Martinez GF (2009). Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling. *J. Hydrol.*, 377(1–2), 80–91. 10.1016/j.jhydrol.2009.08.003
- Gupta HV, Sorooshian S, & Yapo PO (1999). Status of automatic calibration for hydrologic models: Comparison with multilevel expert calibration. *J. Hydrol. Eng.*, 4(2), 135–143. 10.1061/(ASCE)1084-0699(1999)4:2(135)
- Guzman JA, Shirmohammadi A, Sadeghi AM, Wang X, Chu ML, Jha MK, Parajuli PB, Harmel RD, Khare YP & Hernandez JE (2015). Uncertainty considerations in calibration and validation of hydrologic and water quality models. *Trans. ASABE*, 58(6), 1745–1762. <https://doi.org/10.13031/trans.58.10710>.
- Hamby DM (1994). A review of techniques for parameter sensitivity analysis of environmental models. *Environ. Monit. Assess.*, 32(2), 135–154. 10.1007/bf00547132 [PubMed: 24214086]
- Kennedy J, & Eberhart R (1995). Particle swarm optimization Proc. IEEE Intl. Conf. on Neural Networks (pp. 1942–1948). Piscataway, NJ: IEEE 10.1109/icnn.1995.488968

- Larsen MA, Refsgaard JC, Jensen KH, Butts MB, Stisen S, & Mollerup M (2016). Calibration of a distributed hydrology and land surface model using energy flux measurements. *Agric. Forest Meteorol*, 217, 74–88. 10.1016/j.agrformet.2015.11.012
- Laumanns M, Thiele L, Deb K, & Zitzler E (2002). Combining convergence and diversity in evolutionary multiobjective optimization. *Evol. Comput*, 10(3), 263–282. 10.1162/106365602760234108 [PubMed: 12227996]
- Maier HR, Kapelan Z, Kasprzyk J, Kollat J, Matott LS, Cunha MC, ... Reed PM (2014). Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges, and future directions. *Environ. Model. Software*, 62, 271–299. 10.1016/j.envsoft.2014.09.013
- Manache G, & Melching CS (2008). Identification of reliable regression- and correlation-based sensitivity measures for importance ranking of water-quality model parameters. *Environ. Model. Software*, 23(5), 549–562. 10.1016/j.envsoft.2007.08.001
- Morris MD (1991). Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33(2), 161–174. 10.1080/00401706.1991.10484804
- Nash JE, & Sutcliffe JV (1970). River flow forecasting through conceptual models: Part I. A discussion of principles. *J. Hydrol*, 10(3), 282–290. 10.1016/0022-1694(70)90255-6
- Nilsen PK, & Prcovic N (1998). Parallel optimisation in the SCOOP library. *Parallel and Distributed Proc*, 1388, 452–463. 10.1007/3-540-64359-1_719
- Patil SD, & Stieglitz M (2015). Comparing spatial and temporal transferability of hydrological model parameters. *J. Hydrol*, 525, 409–417. 10.1016/j.jhydrol.2015.04.003
- Patil SD, Wigington PJ, Leibowitz SG, Sproles EA, & Comeleo RL (2014). How does spatial variability of climate affect catchment streamflow predictions? *J. Hydrol*, 517, 135–145. 10.1016/j.jhydrol.2014.05.017
- Patil S, & Stieglitz M (2014). Modelling daily streamflow at ungauged catchments: What information is necessary? *Hydrol. Proc*, 28(3), 1159–1169. 10.1002/hyp.9660
- Petrucci G, & Bonhomme C (2014). The dilemma of spatial representation for urban hydrology semi-distributed modelling: Trade-offs among complexity, calibration, and geographical data. *J. Hydrol*, 517, 997–1007. 10.1016/j.jhydrol.2014.06.019
- Razavi S, & Gupta HV (2016a). A new framework for comprehensive, robust, and efficient global sensitivity analysis: 1. Theory. *Water Resour. Res*, 52(1), 423–439. 10.1002/2015WR017558
- Razavi S, & Gupta HV (2016b). A new framework for comprehensive, robust, and efficient global sensitivity analysis: 2. Application. *Water Resour. Res*, 52(1), 440–455. 10.1002/2015WR017559
- Reed PM, Hadka D, Herman JD, Kasprzyk JR, & Kollat JB (2013). Evolutionary multiobjective optimization in water resources: The past, present, and future. *Adv. Water Resour*, 51, 438–456. 10.1016/j.advwatres.2012.01.005
- Saltelli A, & Annoni P (2010). How to avoid a perfunctory sensitivity analysis. *Environ. Model. Software*, 25(12), 1508–1517. 10.1016/j.envsoft.2010.04.012
- Sobol IM (2001). Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Math. Comput. Sim*, 55(1), 271–280. 10.1016/S0378-4754(00)00270-6
- Sobol IM, & Levitan YL (1999). On the use of variance-reducing multipliers in Monte Carlo computations of a global sensitivity index. *Comput. Phys. Commun*, 117(1), 52–61. 10.1016/S0010-4655(98)00156-8
- Song X, Zhang J, Zhan C, Xuan Y, Ye M, & Xu C (2015). Global sensitivity analysis in hydrological modeling: Review of concepts, methods, theoretical framework, and applications. *J. Hydrol*, 523, 739–757. 10.1016/j.jhydrol.2015.02.013
- Tang Y, Reed P, Wagener T, & van Werkhoven K (2007). Comparing sensitivity analysis methods to advance lumped watershed model identification and evaluation. *Hydrol. Earth Syst. Sci*, 11(2), 793–817. 10.5194/hess-11-793-2007
- Tigkas D, Christelis V, & Tsakiris G (2016). Comparative study of evolutionary algorithms for the automatic calibration of the Medbasin-D conceptual hydrological model. *Environ. Proc*, 3(3), 629–644. 10.1007/s40710-016-0147-1
- Vrugt JA, Gupta HV, Bastidas LA, Bouten W, & Sorooshian S (2003). Effective and efficient algorithm for multiobjective optimization of hydrologic models. *Water Resour. Res*, 39(8), 1214. 10.1029/2002WR001746

- Wang QJ (1991). The genetic algorithm and its application to calibrating conceptual rainfall-runoff models. *Water Resour. Res.*, 27(9), 2467–2471. 10.1029/91WR01305
- Whittaker G, Confesor R, Jr., Di Luzio M, & Arnold JG (2010). Detection of overparameterization and overfitting in an automatic calibration of SWAT. *Trans. ASABE*, 53(5), 1487–1499. <https://doi.org/10.13031/2013.34909>
- Willmott CJ, Robeson SM, Matsuura K, & Ficklin DL (2015). Assessment of three dimensionless measures of model performance. *Environ. Model. Software*, 73, 167–174. 10.1016/j.envsoft.2015.08.012
- Yapo PO, Gupta HV, & Sorooshian S (1998). Multi-objective global optimization for hydrologic models. *J. Hydrol.*, 204(1), 83–97. 10.1016/S0022-1694(97)00107-8
- Zhang J, Li Y, Huang G, Chen X & Bao A (2016). Assessment of parameter uncertainty in hydrological model using a Markov-Chain-Monte-Carlo-based multilevel-factorial-analysis method. *J. Hydrol.*, 538, 471–486. 10.1016/j.jhydrol.2016.04.044
- Zitzler E, Laumanns M, & Thiele L (2002). SPEA2: Improving the strength Pareto evolutionary algorithm Proc. Eurogen 2001: Evolutionary methods for design, optimization, and control (pp. 95–100). Barcelona, Spain: International Center for Numerical Methods in Engineering (CIMNE).

Figure 1.
Simple single parameter input x to a model that produces a time series of outputs Y_{sim} .

Figure 2.

Testing multiple instances of an input parameter distribution using M models. The distribution of a single input parameter is characterized by the user as the mean and standard deviation. After performing M model runs, the outputs can be combined into a mean and standard deviation associated with those outputs.

Figure 3. Genetic algorithm used to simultaneously calibrate and determine model sensitivity to input parameters.

Figure 4. Schematic of the EXP-HYDRO model after Patil and Stieglitz (2014). T_a is the mean daily temperature, and T_{min} is the temperature threshold parameter between rain and snow.

Figure 5.

Comparison plot of two objectives used in the optimization algorithm. The objectives of the optimization algorithm that are shown were to maximize the mean model output (x -axis) and to minimize the variance of the model output (y -axis). The 50 best solutions are boxed and were used for further examination.

Figure 6.

Parameter distributions for six input parameters and one dummy parameter selected from 50 individuals that exhibited the highest μ_{KGE} . p1 = rate of decline in runoff (f), p2 = maximum storage of watershed bucket (S_{max}), p3 = maximum subsurface runoff at full bucket (Q_{max}), p4 = degree day snow factor (D_f), p5 = melt temperature (T_{max}), and p6 = snow threshold temperature (T_{min}). The KGE distributions are also shown. The variance or “spread” of the distributions demonstrates the sensitivity of the model to the input parameter. A narrow input parameter distribution means that the parameter is highly impactful for the model, that is, only a small range of values can produce an accurate and robust set of model solutions. Results show that the model is very sensitive to p1 and less sensitive to p2 and p3. The model is not sensitive to p4, p5, p6, or the dummy variable.

Table 1.

Calibration, sensitivity, and uncertainty components of MOESHA

Sensitivity Analysis	+	Uncertainty Analysis	+	Calibration Method	=	MOESHA
<ul style="list-style-type: none"> • Differentiate between impactful (sensitive) and non-impactful (insensitive) parameters. • Determine “importance” of parameters relative to one another. 		<ul style="list-style-type: none"> • Calculate model output variations given uncertainties in input parameters. • Determine “accuracy” and “robustness” of model input parameter uncertainties. 		<ul style="list-style-type: none"> • Determine optimal parameter values to minimize model error and maximize model robustness. 		Combines three methods to dynamically calibrate a hydrologic model while differentiating sensitive parameters and determining model uncertainty.

Table 2.

Comparison of MOESHA to other all-inclusive modeling algorithms that use sensitivity analysis, uncertainty analysis, and calibration.

Program	Sensitivity Analysis	Uncertainty Analysis	Calibration Algorithm	Reference
MOESHA	✓	✓	Dynamic (genetic algorithm)	This article
MOUSE	✓	✓	Static (multiple methods)	Ascough et al. (2014)
PEST	✓	✓	Static (Latin hypercube)	Doherty and Johnston (2003)
SWAT-CUP	✓	✓	Static (Latin hypercube)	Abbaspour (2011)

Table 3.

Results of Sobol GSA suggesting that the parameter sensitivity of p1 is high, the sensitivity of p2 is moderate, and the sensitivities of the remaining parameters are low. The majority of total sensitivity contributed by p2 is through interactive effects between p2 and other parameters, in contrast to its first-order sensitivity.

Sensitivity	p1	p2	p3	p4	p5	p6	Dummy
First-order	0.84	0.02	0.00	0.00	0.00	0.00	0.00
Total	0.98	0.15	0.04	0.00	0.00	0.00	0.00
Percentage	84%	13%	3%	0%	0%	0%	0%