Purpose: The objective of the work encapsulated and documented in this article is to identify the most promising practice oriented and also state-of-the-art approaches that robustly quantify hydrologic and hydraulic (H&H) model uncertainty by taking into account not only parameter, but also model input forcing and model structural uncertainty.

Preface: The US Army Corps of Engineers (USACE) is required to perform risk and uncertainty analyses of existing projects and project alternative measures. However, current capacity within the USACE to quantify the uncertainty of H&H model predictions, and consequently compute risk accurately, is low. Recommendations from three high profile studies underscored the need to better quantify the uncertainty inherent in computing flood-frequency curves from rainfall-runoff modeling, a critical component for conducting project studies and comparing project alternatives in a risk analysis framework (viz., (1) the 2000 National Research Council committee that evaluated the USACE’s use of risk-based analysis in flood damage reduction studies, (2) the 2006 Interagency Levee Policy Review Committee directed by the US Federal Emergency Management Agency to summarize and recommend changes to the current levee policy, and (3) a 2008 journal publication titled “USACE Experience in Implementing Risk Analysis for Flood Damage Reduction Projects” (Davis et al. 2008), co-authored by the former director of the USACE Hydrologic Engineering Center). This technical note summarizes efforts that have been directed at reviewing current available methods to estimate model parameter and predictive error/uncertainty. The intent is to identify specific methods, both practice-driven and state-of-the-art, to move forward with software development and the preparation of related guidance to support the U.S. Army Corps of Engineers’ use of risk-based analysis in flood damage reduction studies now and into the future.

Introduction: Hydrologic models, regardless of their type (e.g., either, empirical, physics-based), often contain parameters that cannot be measured directly because they have no physical basis, it would be impractical, or due to an incompatibility of scales, among other possible reasons. Hence, hydrologic model parameters are inferred by adjusting their values until an acceptable level of agreement is achieved between a set of historical observations of the real world system that the model represents and their simulated counterparts. While manual model calibration is certainly one approach to the problem, it is subjective, labor-intensive, and may also suffer from a lack of consistency and/or repeatability. Moreover, it is difficult to imagine how even an experienced modeler would manage, in a manual calibration context, the large number of estimable parameters associated with present-day, practice-driven, complex, hydrologic model deployments. Fortunately, the computer-based calibration of hydrologic models (which, in contrast with the manual approach to model calibration, is more objective, repeatable, and better capitalizes on the computational capacity of the modern computer) is an active area of research and development which has resulted in numerous automatic calibration methods that are readily available (see
Matott et al. 2009 and references cited therein) for the modern day hydrologic modeler to employ. The knowledge gained by their application and development has provided the hydrologic modeling community with a better understanding of some of the complications associated with calibrating hydrologic models. These complications include the existence of multiple local optima, non-smooth objective function surfaces, and long valleys in parameter space that are a result of excessive parameter correlation or insensitivity (Gupta et al. 2003; Duan et al. 1992).

As mentioned, hydrologic models are typically calibrated by adjusting parameters encapsulated in the simulator until there is an acceptable level of agreement between a set of historical data and their model simulated counterparts. The parameters obtained via calibration are often then used by the model to predict system behavior for one or more pre-defined scenarios of interest. Regardless of the calibration method employed and the type of model used (e.g., empirical or physics-based), some if not all of the parameter values obtained through the calibration process possess a degree of quantifiable uncertainty. This is because the observed data contain measurement errors and also because the model never perfectly represents the watershed system or exactly fits the observation data. Where model parameters are uncertain, so too will be model predictions.

The estimation of model parameter and predictive error/uncertainty is an active area of research and development in the hydrologic modeling community. As a result, there are a large number of methods available to estimate model error/uncertainty (Matott et al. 2009). However, error/uncertainty quantification is not commonplace in the H&H practice community (Montanari 2007). While the review encapsulated in this document was not complete in evaluating all available error/uncertainty analysis methods, it attempts to identify the most promising methods with which to move forward for current applications, and further research and development to support the U.S. Army Corps of Engineers’ use of risk-based analysis in flood damage reduction studies.

The objective of the work documented in this article is to identify the most promising practice oriented and state-of-the-art approaches that robustly quantify H&H model uncertainty by taking into account not only parameter, but also model input forcing and model structural uncertainty. Structural error refers to the inability of even the best model with optimal parameters to exactly reproduce the target model output. Planned future work activities will (1) modify the most promising error/uncertainty analysis method(s) identified in this document to be made more usable for a broader user base; such as the USACE H&H community of practice, (2) prepare guidance documents that demonstrate in a clear and practical manner how to use the (modified) method(s) within the context of the USACE risk analysis framework for a given local project, and (3) examine and develop means to improve upon the efficiency of the method(s), which can be computationally costly.

Model parameter and/or predictive error/uncertainty estimation provides the following benefits, among others (Schoups and Vrugt 2010; Schoups et al. 2008, 2010):

- Understanding of the predictive capabilities and limitations of the hydrologic/environmental models that are deployed for planning and design;
- A basis for model comparison and selection;
- Identification of the best water management strategies that reflect the likelihood of outcomes;
- Data collection aimed at improving hydrologic predictions and water management; and
- Capacity for regionalization and extrapolation of hydrologic parameters to ungauged basins.
While there is no single error/uncertainty analysis methodology that is applicable everywhere all of the time, the intent of this article is to review currently available practice oriented and state-of-the-art methods to identify those that best support the path forward for the previously mentioned future related work activities. Metrics for method selection include, among others:

- Capacity for the method to be complete with its analysis; i.e., the ability to account for parameter uncertainty, model input forcing uncertainty, and model structural uncertainty;
- Acceptance of the method within the H&H modeling community;
- Technical validity/soundness of the method; and
- Practicality of the method in terms of its implementation and/or use.

An important consideration in assessing the performance of model parameter estimation and uncertainty analysis software is that of run time. Minimizing the number of model runs required during the calibration/uncertainty analysis process is nearly always important, but particularly when the objective function landscape contains multiple local minima or model run times are high. Minimizing the number of required model runs will be one of the primary factors driving the noted planned future research and development activities such that the resulting optimization and uncertainty tool(s) are more compatible with the computationally expensive physics-based models that are becoming more commonly used within the practice community.

**REVIEW:** State-of-the-art methods that are considered/evaluated include Markov Chain Monte Carlo (MCMC), Bayesian model averaging (BMA), data assimilation (DA), and in consideration of the expense associated with estimating model parameter and predictive uncertainty, the use of an Artificial Neural Network (ANN) as a surrogate for the H&H model. Current practice oriented approaches that are evaluated include Generalized Likelihood Uncertainty Estimation (GLUE) (Beven and Binley 1992), post-calibration linear and nonlinear predictive error variance analysis, calibration-constrained Monte Carlo, and the use of a modified version of an evolutionary strategy to characterize the pareto front to examine tradeoffs associated model prediction.

**Markov Chain Monte Carlo:** Bayesian-based approaches to model calibration, wherein a prior distribution for the model is proposed, and the vector of adjustable model parameters is treated as a random variable with a target probability distribution that is conditioned with observed data, are a formal means to obtain a realistic and reliable estimate of model uncertainty. In particular, Markov Chain Monte Carlo (MCMC) simulation, which is more efficient than other Monte Carlo methods, is used for inference, search, and optimization with hydrologic models (Harmon and Challenor 1997; Kuczera and Parent 1998; Campbell et al. 1999; Campbell and Bates 2001; Makowski et al. 2002; Qian et al. 2003; Kanso et al. 2003; Vrugt et al. 2003; Vrugt et al. 2008a, b; Vrugt et al. 2009). Markov Chain Monte Carlo is a formal Bayesian approach for estimating the posterior probability distribution of model parameters. The idea behind MCMC is that while one wants to compute a probability density, $p(\theta | y)$, where $\theta$ and $y$ represent the vector of adjustable model parameters and the observed data, respectively, there is the understanding that such an endeavor may be impracticable. Additionally, simply being able to generate a large random sample from the probability density would be equally sufficient as knowing its exact form. Hence, the problem then becomes one of effectively and efficiently generating a large number of random draws from $p(\theta | y)$. It was discovered that an efficient means to this end is to construct a Markov chain, a stochastic process of values that unfold in time, with the following properties: (1) the state space (set of possible values) for the Markov chain is the same as that for $\theta$; (2) the Markov chain is easy to
simulate from; and (3) the Markov chain’s equilibrium distribution is the desired probability density \( p(\theta|y) \). By constructing such a Markov chain, one could then simply run it to equilibrium and subsequently sample from its stationary distribution.

A Markov chain with the above mentioned properties can be constructed, such that the probability distribution for where to go at time \( t+1 \) given the current location at time \( t \) is known, in the following manner: (1) choose a proposal distribution \( f(\theta_p, \theta_t) \) to consider where to go next given the current location \( \theta_t \), and generate a candidate point \( \theta_p \) from \( f \); (2) for a symmetric proposal distribution \( f \) wherein \( f(\theta_p, \theta_t) = f(\theta_t, \theta_p) \), at time \( t+1 \) accept the move to the candidate point with the Metropolis acceptance probability (Metropolis et al. 1953)

\[
\alpha_M(\theta_t, \theta_p) = \min\left[1, \frac{p(\theta_p|y)}{p(\theta_t|y)}\right]
\]

otherwise, set \( \theta_{t+1} = \theta_t \). In effect, toss a Bernoulli coin with probability \( \alpha_M \) of coming up heads, and if heads, then set \( \theta_{t+1} = \theta_p \), otherwise set \( \theta_{t+1} = \theta_t \). The Metropolis algorithm is implemented by sampling a uniform label \( r \) over the interval \([0,1]\) and comparing it with \( \alpha_M \). If \( r \leq \alpha_M \), then accept the proposal, and if \( r > \alpha_M \), then remain at the current point. The Metropolis algorithm will always move into a region of higher posterior probability, but it will also explore regions with lower posterior probability with probability \( r \). It has been shown that for symmetric proposal distributions the equilibrium distribution for the constructed Markov chain is \( p(\theta|y) \). Hastings (1970) extended the Metropolis algorithm to accommodate for non-symmetric proposal distributions.

The scale and orientation of the selected proposal distribution can impact the MCMC algorithm efficiency. The primary objective is to select a proposal distribution that facilitates optimal mixing (i.e., low serial correlation) of the chain(s). There are an infinite number of possible proposal distributions that one could select. With the Metropolis algorithm, the only requirement is that the proposal distribution be symmetric. A commonly used symmetric proposal distribution is a Gaussian distribution centered at the current location \( \theta_t \). While of course circular, general guidance is to select a proposal distribution that closely approximates the unknown target distribution. In this case, the constructed Markov chain will converge rapidly to the target distribution. On the other hand, if the proposal distribution is too wide (i.e., the proposal distribution standard deviation is too big), then acceptance rates for the proposed jumps will be low, which will result in poor mixing (high autocorrelation) and slow convergence of the chain(s) to their equilibrium target distribution. By contrast, if the proposal distribution is too narrow (i.e., the proposal distribution standard deviation is too small), then the acceptance rate will be high for the proposed jumps, but convergence to the target distribution will again be slow because each accepted jump will only be a small distance from the current location.

With DE-MC, multiple chains, initialized from overdispersed states, are run in parallel and learn from each other by way of jump proposals that are generated by taking the difference of two randomly selected chains from the current population. The probability of selecting the jump proposal is determined by using the Metropolis algorithm (Metropolis et al. 1953). Within DE-MC, the noted difference vector contained in the jump proposal simply and effectively encapsulates both the scale and orientation for the jumping distribution. It is easily seen that the DE-MC proposal vector is reversible; hence, defining a Metropolis step. The proposal vector for the simple but effective DE-MC algorithm is given by

\[ x_p = x_i + \gamma (x_{R1} - x_{R2}) + e \] (2)

where \( x_p, x_i, \gamma, x_{R1}, x_{R2}, \) and \( e \) represent the proposal vector of dimension \( d \), the \( i^{th} \) of the \( N \) chains which constitute the evolving population, a weighting factor, two unique vectors from the current population, excluding the \( i^{th} \) chain, randomly selected without replacement, and an error term randomly sampled from a symmetric distribution with small variance compared with that of the target. \( N \) and \( \gamma \) constitute the two parameters associated with the DE-MC method. ter Braak (2006) suggested \( N = 2d \) or \( 3d \) for simple unimodal targets and \( N = 10d \) to \( 20d \) for more complicated target distributions. Assuming the target distribution is multivariate normal, the optimal choice for \( \gamma \) is \( 2.38/2d \), and this value was observed to work well for the tests and examples considered by ter Braak (2006). Moreover, ter Braak (2006) demonstrated that adapting the DE-MC algorithm such that \( \gamma \) equals one every tenth generation mitigates against the potential of becoming trapped in a single mode within a multimodal distribution. With \( \gamma \) equal to one, if one of \( x_{R1} \) or \( x_{R2} \) is in a modal region and the other is close to \( x_i \), then the proposal vector (see Equation 2) can also jump into the modal region. In attempts to improve algorithm efficiency by allowing use of a smaller value for \( N \), a variation on the basic DE-MC algorithm previously mentioned that is useful for large \( d \), which often requires large \( N \), is to employ crossover updating wherein sampling occurs in lower dimensional subspaces. For example, one crossover updating strategy (ter Braak, 2006) is that each dimension (but not all) of the proposal, \( x_{ij} (j = 1, \ldots, d) \), is replaced by \( x_i \) with probability \( 1 - CR \), where \( CR \) is a predetermined crossover rate probability. The DE-MC algorithm without crossover updating corresponds to \( CR = 1 \).

ter Braak (2006) demonstrated notable efficiency gains for DE-MC, when compared with Random Walk Metropolis (RWM) MCMC for tests involving known target distributions, including small populations, large populations, and also a bimodal target. The simplicity of the adaptive DE-MC algorithm, its efficiency and effectiveness relative to RWM, and its capacity to appropriately tune the scale and orientation of the jumping distribution were all underscored.

Vrugt et al. (2008a, 2009) introduced differential evolution adaptive metropolis (DREAM) – an adaptive MCMC sampler that was designed to estimate the posterior probability density of hydrologic model parameters. DREAM is a modification to the DE-MC method described by ter Braak (2006) that includes adaptations directed at further improving overall algorithm efficiency and effectiveness. While it is described in detail by Vrugt et al. (2008a), Vrugt et al. (2009), and Minasny et al. (2011), the DREAM algorithm is summarized below:

1. Initialize the population, of size \( N \), using the predetermined prior distribution.
2. Compute the density \( \pi(x') \) for each point in the population.
3. Loop over the population (for $i = 1, \ldots, N$) and generate a proposal for each $i$

$$
\mathbf{x}_p = \mathbf{x}_i + (1_d + \mathbf{e}_d) \gamma(\mathbf{\delta}, d') \left[ \sum_{j=1}^{d} \mathbf{x}^{r_j}_i - \sum_{n=1}^{d} \mathbf{x}^{r_n}_i \right] + \mathbf{\epsilon}_d
$$

where $r_1(j), \quad r_2(n) \in 1, \ldots, N, \quad r_1(j) \neq r_2(n) \neq i$ for $j,n = 1, \ldots, \delta$, and $\mathbf{x}_p, \mathbf{x}_i, 1_d$, and $\gamma(\mathbf{\delta}, d')$ represent the proposal vector of dimension $d$, the $i^{th}$ of the $N$ chains constituting the population, a $d$ dimensional vector of unity, and a weighting factor that is a function of the number of vectors from the current population employed to create the proposal. The values of $\mathbf{e}_d$ are drawn from a uniform distribution $U_d(-b, b)$ wherein $|b| < 1$, and the values of $\mathbf{\epsilon}_d$ are drawn from a normal distribution centered at zero with variance small compared with that of the target.

4. For the proposal vector, update a randomized subspace of the $d$ dimensions using a binomial crossover updating scheme where the crossover probability is given by CR (ter Braak 2006); viz., for $j = 1, \ldots, d$, $\mathbf{x}^{\mathbf{\delta}}_p = \mathbf{x}^j_i$ if $U \leq 1-CR$, where $U \in [0,1]$ is drawn from a uniform distribution; otherwise, $\mathbf{x}^{\mathbf{\delta}}_p = \mathbf{x}^j_i$.

5. Compute the density for the proposal vector and accept the jump proposal with Metropolis acceptance probability (Metropolis et al. 1953).

6. Remove potential outlier chains using the inter-quartile range (IQR) statistic, but only during burn-in.

7. Compute the Gelman and Rubin convergence diagnostic (Gelman and Rubin 1992), $\hat{R}_j$, for each dimension using the last 50 percent of the samples in each chain.

8. If the Gelman and Rubin convergence diagnostic, $\hat{R}_j$, is less than or equal to 1.2 for all dimensions, then stop, otherwise, return to step 3.

Vrugt et al. (2008a, 2009) underscored that DREAM differs from its basis, DE-MC, in three important ways:

1. While ter Braak (2006) did mention crossover updating as a means to improve DE-MC algorithm efficiency, particularly for higher dimensional problems, DREAM consistently employs randomized subspace sampling, using the binomial crossover updating scheme presented in ter Braak (2006). Moreover, during burn-in, in attempts to accelerate convergence of the chains to their equilibrium distribution, DREAM determines crossover updating probability values that give preference to larger jumps over smaller ones in each of the $N$ chains.

2. With DE-MC, proposals are a function of only two members of the $N$ chains; however, with DREAM the proposals include higher order pairs in attempts to increase diversity. And in DREAM, the weighting parameter, $\gamma(\mathbf{\delta}, d')$, is more aggressively set than within DE-MC to promote jumping between different modes of the posterior.
3. In DREAM, outlier chains are removed during burn-in to facilitate convergence to the target distribution.

Vrugt et al. (2009) emphasized the capacity of DREAM to accommodate nonlinearity, high dimensionality, and multimodality in a manner generally superior to other adaptive MCMC samplers. One of their case studies involved using DREAM to infer the posterior parameter distribution of a Sacramento Soil Moisture Accounting (SAC-SMA) model deployment to the 1,950 square kilometer Leaf River watershed using two years of observed daily stream discharge data. Several studies have underscored the difficulties to be encountered calibrating the SAC-SMA model, including multiple local optima with both small and large regions of attraction, discontinuous first derivatives, and curving multidimensional ridges. DREAM outperformed other adaptive MCMC samplers also considered and also the popular Shuffled Complex Evolution (SCE-UA) global purpose solver (Duan et al. 1992; Duan et al. 1993).

Vrugt et al. (2008a) employed DREAM to infer the posterior distribution for the five parameter Hydrologic Model (HYMOD) (Boyle 2000) while also incorporating rainfall forcing data error into the estimation process. They observed that treatment of forcing data error during calibration significantly altered the posterior distribution of the hydrologic model parameters. Vrugt et al. (2008b) used DREAM with two HYMOD model deployments to infer parameter and predictive uncertainty while considering model structural, precipitation forcing, and parameter uncertainty. Another notable adaptive MCMC sampler recently developed and utilized for hydrologic modeling contexts is the shuffled complex evolution Metropolis algorithm SCEM-UA (Vrugt et al. 2003). Skahill and Baggett (2012) explored ways to improve upon the efficiency of the SCEM-UA MCMC sampler using local function approximation techniques. The adaptive MCMC algorithm SCEM-UA, a modification to the popular SCE-UA global optimization algorithm; however, has been documented to lack detailed balance (Vrugt et al. 2008a). DREAM is the current state-of-the-art adaptive MCMC sampler.

**Bayesian Model Averaging:** Understanding that no single model structure is necessarily superior to all other available structures for various predictions under all circumstances, Bayesian model averaging (BMA) can provide a basis to include the model selection process into the assessment of uncertainty. BMA is a scheme to infer a combined probabilistic prediction whose average possesses more reliability and skill than that which can be obtained from any one of the individual models that constitute the model combination (Madigan and Raftery 1994). The probability distribution function (pdf) for a specific prediction of interest computed using BMA is a weighted average of the pdfs from the individual models (Hoeting et al. 1999). It is the author’s perspective that the costs associated with the implementation and use of BMA would outweigh its benefits when considering its potential to support model predictive uncertainty analysis for planning and design studies within the USACE. Hoeting et al. (1999) list and discuss several other BMA-related implementation matters, one key nontrivial issue being inclusion/exclusion into the set of models that constitute the model combination.

**Data Assimilation:** Data assimilation (DA) is a method of combining an imperfect simulation model with imperfect or incomplete observation data. The ensemble Kalman filter (EnKF) is one approach to nonlinear state estimation, and it is widely used in weather forecasting, where the models are of extremely high order and nonlinear, the initial states are highly uncertain, and a large number of observations are available. EnKF is a sequential data assimilation method in that
model states are continuously updated as new observation data becomes available. Sequential (online) data assimilation methodologies, wherein model states are continuously updated as new observation data becomes available, provide a general framework for isolating and quantifying input, output, and model structural uncertainty, and for optimal merging of uncertain model predictions with observations. Prior to their application, it is typically assumed that the optimal parameter estimate is known (i.e., the forward model is calibrated). The EnKF is a sequential Monte Carlo method, an ensemble version of the classical Kalman Filter. With EnKF, the starting point is an ensemble of state estimates that captures the probability distribution of the state. The sample points are propagated through the nonlinear model and the probability density function of the actual state is approximated by the ensemble of the estimates. In hydrologic prediction, the EnKF has primarily been used to update states including soil moisture, snow water equivalent, and soil temperature (see Dechant and Moradkhani 2011, and references cited therein). As with BMA, it is the author’s perspective that the costs to implement and use DA to estimate model predictive uncertainty to support USACE planning and design studies outweigh its potential benefits. Data assimilation systems are primarily used in operational forecasting environments and their implementation is highly non-trivial.

Artificial Neural Networks (ANNs): Shrestha et al. (2009) estimate model parameter and predictive uncertainty via Monte Carlo Simulation (MCS) using an Artificial Neural Network (ANN) as a surrogate for a forward hydrologic model. Their study was motivated by the basic observation that while general in application, MCS is nonetheless expensive and often limited to models that are parsimonious with respect to their computational requirements. The prediction intervals obtained from application of the trained ANN agreed reasonably well with those obtained from the process model. While the focus of the study was solely on parameter uncertainty, the authors indicated that other sources of uncertainty; viz., input and structural, could also be included into the analysis.

Generalized Likelihood Uncertainty Estimation: Vrugt et al. (2008b) hypothesized that researchers and practitioners alike will continue to use, at least for the foreseeable future, relatively simple methods to quantify model uncertainty rather than more complete approaches, not only because of the mathematical rigor, but also possibly because of the implementation complexity associated with more sophisticated methods such as sequential filtering and Bayesian MCMC. The generalized likelihood uncertainty estimation (GLUE) method of Beven and Binley (1992) is a relatively simple approach for estimating model (parameter and predictive) uncertainty that since its inception has found widespread use, not only in hydrology, but also in other application settings (Vrugt et al. 2008b, and references cited therein). GLUE’s apparent attraction for estimating model uncertainty is likely not only due to its simplicity and ease of implementation, but also because of the few assumptions that it requires (Romanowicz and Beven 2005), its applicability to nonlinear systems, its apparent success in practical studies, and that it appears to provide the necessary estimation of uncertainty (Stedinger et al. 2008, and references cited therein).

Underpinning GLUE is the concept of model equifinality, which suggests there are multiple equally acceptable models given the possibly ambiguous data associated with and our incomplete understanding of the system(s) that we wish to simulate. GLUE was developed in an attempt to deal with those cases for which the theoretical assumptions associated with likelihood measures do not seem appropriate (Vrugt et al. 2008b), and also for overparameterized modeling contexts where model uniqueness/identifiability can of course be problematic (Romanowicz and Beven
2005). In the context of Bayes Theorem, a likelihood measure is unique for a given set of assumptions regarding the structure of the errors, and it corresponds to the probability density function of the errors. However, the GLUE method rejects the formal statistical approach underlying a likelihood measure. Instead, in the spirit of the concept of equifinality, it uses a likelihood measure to subjectively find a set of model representations (input, model structure, model parameters, model errors) that are acceptable/behavioral in terms of their capacity to reproduce historical observations, and to determine the extent with which each acceptable model is a member of the set.

For the GLUE method, the only requirements for a likelihood measure are that it monotonically increases with improved model-to-measurement fit and that a value of zero is assigned for non-behavioral model representations. Beven et al. (2000) summarized many of the likelihood measures that had been used with past applications of the GLUE method. Two likelihood measures that are popularly used with the GLUE method include the inverse error variance (see equation 4) and the model efficiency (see equation 5) functions:

\[
L(\theta|\hat{Y}, \hat{\zeta}, \hat{\phi}) = (\sigma^2)^T
\]

\[
L(\theta|\hat{Y}, \hat{\zeta}, \hat{\phi}) = \left(1 - \frac{\sigma^2}{\sigma^2_\phi}\right), \sigma^2_\phi \geq \sigma^2 \Rightarrow L = 0
\]

where \(\theta\), \(\hat{Y}\), \(\hat{\zeta}\), \(\hat{\phi}\), \(\sigma^2\), \(\sigma^2_\phi\), and \(T\) represent the model parameters, system output observations, measured boundary (forcing) conditions, measured initial conditions, the weighted variance of the residuals, the weighted variance of the observations, and a shaping factor that as it increases gives greater weight to models that yield a better fit, respectively. With GLUE, the likelihood weights can be updated as new data becomes available by way of Bayes Theorem. In light of the concept of equifinality, which is the basis for GLUE, assessment of model uncertainty is a natural process with application of the method. And for GLUE the selected informal likelihood measure impacts the uncertainty bounds on the predictions, as, for example, is evident with the choice of the shaping factor, \(T\), for the inverse error variance likelihood measure (see equation 4). Hence, with that understanding, for GLUE, model uncertainty is subjective.

GLUE is based on Monte Carlo simulation. Model parameter sets are sampled from a probability distribution, often uniform. The acceptability of each sampled parameter set is evaluated by way of a specified goodness of fit/likelihood measure which quantifies model to measurement misfit for a given calibration period. Each sampled model is classified as behavioral or non-behavioral, with the determination based on a predetermined threshold value associated with the goodness of fit measure. Non-behavioral models are discarded (i.e., assigned a likelihood value of zero), and the behavioral models are retained and subsequently rescaled in attempts to estimate model uncertainty. In particular, the GLUE method is as follows (Vrugt et al. 2008b):

- Use the specified prior distribution, \(p(\theta)\), to draw a sample of points \(\Theta\) of size \(N\).
- Compute the likelihood \(L(\theta|\hat{Y}, \hat{\zeta}, \hat{\phi})\) of each point of \(\Theta\), \(\theta^i, i = 1, \ldots, N\).
Define a threshold value for \( L \) to determine for each member of \( \Theta \) whether it is behavioral or non-behavioral. Retain the \( k \) members of \( \Theta \) subjectively classified as behavioral/acceptable in \( D \). Discard the subjectively classified non-behavioral models.

- Rescale the likelihood values of the models that were subjectively classified as behavioral and placed in \( D \) such that their \( k \) rescaled values sum to one.
- Weight each simulated value with its associated rescaled likelihood value.
- Sort the simulated output with their assigned weights to create, effectively a cumulative distribution function of the prediction, and subsequently use it to estimate uncertainty bounds for the prediction.

As previously mentioned, Vrugt et al. (2008b) used DREAM with two HYMOD model deployments to infer parameter and predictive uncertainty while considering model structural, precipitation forcing, and parameter uncertainty. The principal motivation for their study was to compare a statistically formal Bayesian MCMC approach with the informal GLUE method for assessing conceptual watershed model parameter and predictive uncertainty. Vrugt et al. (2008b) made several points during their comparison study, including, among others:

- The formal DREAM MCMC method is more efficient than the informal GLUE method in finding “acceptable”/“behavioral” models.
- The formal DREAM MCMC method is able to treat sources of model uncertainty separately; viz., input forcing, parameter, and model structural; whereas, the informal GLUE method maps all sources of uncertainty onto the model parameters. This point was emphasized by inspection of their histogram plots of the inferred parameter distributions obtained from the two methods for one of the two HYMOD model deployments. For DREAM, the five model parameters were well identified, despite the fifty-eight additional parameters that were included into the DREAM MCMC analysis to account for precipitation forcing and model structural uncertainty; whereas, for GLUE, the distributions of the five model parameters were much wider.
- The formal DREAM MCMC method was better able to cover stream discharge observations during both the calibration and evaluation periods in comparison with the GLUE method. And for DREAM, the coverage was in a manner closely consistent with the specified percent uncertainty bounds. In addition, it was observed that, for GLUE, by not separately accounting for input forcing uncertainty, when forcing error was large, total predictive uncertainty could be significantly underestimated.
- Estimates of total model predictive uncertainty obtained from the two methods can be quite similar.
- By separately accounting for sources of uncertainty, the formal Bayesian approach is better suited to improve our understanding of watershed systems and the science of hydrologic modeling.

Stedinger et al. (2008) used a simple linear rainfall-runoff model to clearly and simply demonstrate deficiencies associated with the GLUE method when an informal likelihood measure is employed. In this case, model calibration reduces to a linear regression problem wherein analytical expressions for model uncertainty are well established. The primary deficiency with the GLUE method is that when an informal likelihood measure is employed its predictive limits can be significantly different from those obtained using a statistically valid approach (Christensen 2004;
Montanari 2005; Mantovan and Todini 2006; Stedinger et al. 2008). Stedinger et al. (2008) maintain that if the GLUE method can’t reproduce the uncertainty for simple test cases, then the method cannot be expected to perform with difficult problems for which the correct answer is unknown. Stedinger et al. (2008) also demonstrate how to use GLUE to ensure that resulting uncertainty estimates are consistent with formal statistical methods. They mentioned that if an informal likelihood measure is employed that does not adequately represent the sampling distribution of the model errors, then GLUE’s results are arbitrary, lacking statistical validity, and should not be used in scientific work. Doherty and Welter (2010) discuss in detail model structural noise, that component of model-to-measurement misfit not attributable to measurement error but rather to the model’s incapacity to perfectly replicate the real world, and they also present alternative means to mitigate its potential negative impact on model calibration.

Post-calibration Linear and Nonlinear Predictive Error Variance Analysis: The hydrologic/environmental model calibration process is non-trivial and the final outcome; viz., the final set of specified adjustable model parameters, is identified with a degree of uncertainty due to observed data error, input forcing data (e.g., meteorological) error, model parameter error, and model structural error (i.e., the model as designed does not perfectly represent the system). Hence, predictions made with a calibrated model will also possess error/uncertainty. Predictive error variance attempts to determine how wrong predictions may be from a calibrated model. By way of example, Gallagher and Doherty (2007) evaluated the strengths and weaknesses of using linear individual parameter and predictive confidence intervals:

\[
\text{Prob} \left( p_i - t_{\frac{\alpha}{2}} \frac{n-m}{\sigma_i} \leq p_i \leq p_i + t_{\frac{\alpha}{2}} \frac{n-m}{\sigma_i} \right) = 1 - \alpha
\]

(6)

\[
\text{Prob} \left( s - t_{\frac{\alpha}{2}} \frac{n-m}{\sigma_s} \leq s \leq s + t_{\frac{\alpha}{2}} \frac{n-m}{\sigma_s} \right) = 1 - \alpha
\]

(7)

linear simultaneous parameter and predictive confidence intervals

\[
\text{Prob} \left( p_i - \sqrt{mF_a(m,n-m)} \frac{n-m}{\sigma_i} \leq p_i \leq p_i + \sqrt{mF_a(m,n-m)} \frac{n-m}{\sigma_i} \right) = 1 - \alpha
\]

(8)

\[
\text{Prob} \left( s - \sqrt{mF_a(m,n-m)} \frac{n-m}{\sigma_s} \leq s \leq s + \sqrt{mF_a(m,n-m)} \frac{n-m}{\sigma_s} \right) = 1 - \alpha
\]

(9)

nonlinear individual and simultaneous parameter and predictive confidence intervals and prediction intervals:

\[
\Phi(p) - \Phi(p) \leq \sigma^2 t_{\frac{\alpha}{2}} (n-m)
\]

(10)

\[
\Phi(p) - \Phi(p) \leq m \sigma^2 F_a(m,n-m)
\]

(11)

\[
\frac{\Phi(p) - \Phi(p)}{\sigma^2 + \sigma^2} \leq (m+1)/(n-m) F_a(m+1,n-m)
\]

(12)
and also Bayesian MCMC for estimating hydrologic model parameter and predictive uncertainty wherein $p_i$, $t_{\alpha/2}(n-m)$, $\sigma_i$, $p$, $s$, $s$, $m$, $n$, $F(p, m, n-m)$, $\Phi(p)$, $\sigma^2$, $\epsilon^2$, $\sigma^2$, $p$, $m$, $n$, $X$, $Q$, $Z$, $c$, $q$, $y_0$, Equations 10, 11, 12, and 13 represent the $i^{th}$ estimated parameter value, upper $\alpha/2$ confidence point of the Student’s $t(n-m)$ distribution, standard deviation of the $i^{th}$ individual estimated parameter, unknown true value of the $i^{th}$ parameter, level of confidence, prediction calculated by the calibrated model, unknown true value of the prediction, standard deviation of $s$, number of estimated model parameters, number of observations used in the calibration process, the F distribution, objective function for a parameter set $p$, objective function calculated for the local minimizer $p$, reference variance, predictive error term, variance of the error term corresponding to the prediction of interest, a parameter set $p$, a parameter set $p$, model sensitivity matrix, observation weight matrix, vector of sensitivities of model prediction to estimated model parameters, residual vector, observations, model output corresponding with the parameter set $p$, implicit definition of individual confidence intervals for both parameters and predictions for a nonlinear model, implicit definition of simultaneous confidence region for both parameters and predictions for a nonlinear model, prediction intervals that include the effects of errors that accompany measurements of system state, and the equation which can be solved using the method of Lagrange multipliers to define the boundary of the simultaneous confidence region for a prediction $s$ whose sensitivity vector is $Z$, respectively. For their evaluation study, Gallagher and Doherty (2007) selected a hydrologic model that, in terms of its run requirements, supported a complete analysis for each method.

Gallagher and Doherty (2007) mentioned that $q$ and $y$ should be replaced by their transformed equivalents if measures are taken to reduce heteroscedascity and serial correlation of measurement noise. However, based on their example, they hypothesized that a more pragmatic prediction specific weighting strategy should have been employed in attempts to improve the capacity of the calibrated model to predict high flows. They employed a Box-Cox transformation to achieve homoscedasticity of model to measurement residuals for the model calibration period, which was dominated by low flows. The hypothesis by Gallagher and Doherty (2007) was in fact introduced and confirmed in earlier work presented by Moore and Doherty (2005) who explored, among others, different means to tailor the calibration process to reduce model predictive error. Moore and Doherty (2005) acknowledged that a prediction specific weights strategy is in violation of the theory for traditional least squares practice, but stated that it is likely no less in violation than many other commonly implemented methods of weights assignment in light of the fact that where calibration data noise is dominated by structural error $C(e)$ is not known nor easily estimated.
Gallagher and Doherty (2007) mention that linear uncertainty analysis can be implemented with virtually no computational burden. It provides a simple basis for assessing parameter correlation and, in a relative sense, model parameter uncertainty. However, it is not very useful for quantifying model prediction uncertainty. Gallagher and Doherty (2007) demonstrated that nonlinear calibration constrained optimization can be employed to obtain good estimates of model predictive uncertainty; albeit, in a manner not as complete as the parameter and predictive uncertainty estimates that one can obtain using far more computationally intensive Bayesian MCMC methods. And due to its low computational cost, Gallagher and Doherty (2007) recommended its usage to be a routine part of the many model deployment contexts for which it is applicable.

Modern hydrologic model deployments more often than not can easily become highly parameterized by virtue of the spatial data products (e.g., elevation, soils, land use, vegetative cover) and related spatial processing tools such as the Watershed Modeling System (chl.erdc.usace.army.mil/wms) that are now readily available to support the model development and parameterization process. And, the observed system response data that we have at our disposal and/or impart to the model calibration process is often not compatible with the complexity of the model that we (and/or our clients) wish to deploy, in that it does not support the unique estimation of all of the model parameters that must be specified as adjustable due to the desired complexity that was included into the model. But, the complexity is retained in the model because it is presumed that the simulated processes and/or predictions depend on the system detail that is reflected in the model.

“Regularization” is a mathematical term that, in its broadest sense, refers to any measure that is taken to ensure that a stable solution is obtained to an otherwise ill-posed inverse problem. In traditional calibration practice, this is achieved through adherence to the so-called “principle of parsimony” in which parameters requiring adjustment through the calibration process are reduced to a number for which a unique estimate can be obtained for each. If calibration is computer-assisted, then prior to initiating execution of the parameter estimation software package, the modeler selects those parameters that he/she wishes to estimate (normally on the basis of anticipated higher sensitivity of model outputs to these parameters) and holds other parameters fixed at “sensible values”. The principle of parsimony underpins the linear and nonlinear calibration constrained predictive uncertainty analysis methods that were presented and evaluated by Gallagher and Doherty (2007).

Through the use of parameter estimation algorithms that allow mathematical regularization to be implemented as part of the parameter estimation process itself, the result is a stable solution to the inverse problem (regardless of how ill-posed it is), and avoidance of the deleterious effects of numerical instability on both the parameter estimation process itself, and on the outcomes of that process, namely the set of estimated parameter values. A well-designed regularization algorithm, like its manual counterpart, achieves numerical stability by re-formulating the inverse problem in a way that recognizes the level of parsimony that is necessary to attain a stable solution to that problem. However, this “parsimonizing” is undertaken in the context of a specific calibration dataset, allowing numerical stability to be achieved without compromising model-to-measurement fit any more than is deemed necessary by the modeler.

Tonkin et al. (2007) reformulated the original predictive error variance analysis method presented by Vecchia and Cooley (1987), summarized by Gallagher and Doherty (2007), and in so doing
demonstrated an efficient means to perform predictive error variance analysis for highly parameterized nonlinear model deployments wherein subspace regularization was utilized to support model calibration. The approach introduced by Tonkin et al. (2007) also incorporates the error associated with the parsimonizing that was required to calibrate the model; i.e., the error associated with making predictions with the model that had to be simplified via a regularization calibration methodology. More specifically, the method of Tonkin et al. (2007) accounts for that component of error which accounts for fine system detail upon which the model prediction depends but that cannot be represented in the calibrated model.

**Subspace Monte Carlo:** Tonkin and Doherty (2009) present a subspace Monte Carlo (SSMC) method that efficiently estimates the uncertainty of a highly parameterized model in a calibration constrained Monte Carlo context. With the SSMC method, the model is first calibrated using a subspace regularization methodology; for example, such as Truncated Singular Value Decomposition (TSVD) (Aster et al. 2005) or the hybrid method SVD Assist (Tonkin and Doherty 2005), which attempts to combine the strengths of two alternative strategies; viz., TSVD and Tikhonov-based regularization (Doherty and Skahill 2006 and references cited therein). Following the use of a subspace regularized inversion methodology to calibrate the model and estimation of the model calibration null space, the SSMC method randomly generates and processes parameter realizations that efficiently also calibrate the model or require little additional computational effort to recalibrate the model. The SSMC method is a blend of error variance theory with Monte Carlo simulation. As with the post subspace regularized inversion nonlinear predictive error variance analysis methodology of Tonkin et al. (2007), the SSMC method of Tonkin and Doherty (2009) permits fine system detail to be incorporated into the error analysis. In contrast with error variance analysis wherein the process is for a specific prediction of interest, with the SSMC method, once an ensemble of models is determined, the error for a variety of model outputs may be evaluated.

**Efficient Regularized Inversion and Uncertainty Analysis using an Evolutionary Strategy:** Moore et al. (2010) utilized a modified version of the state-of-the-art Covariance Matrix Adaption Evolutionary Strategy (CMAES) (Hansen and Ostermeier 2001); Hansen et al. 2003) to characterize the pareto front not only to examine tradeoffs associated with regularized inversion (i.e., fitting the hard and soft data imparted to the model calibration process, respectively) but also model prediction. While the method is applicable for parsimonious and highly parameterized model deployments its practical application is presently limited to models that are computationally efficient. Doherty (2010) developed an additional more efficient means to traverse the pareto front.

**DISCUSSION AND CONCLUSIONS:** To support the U.S. Army Corps of Engineers’ use of risk-based analysis in flood damage reduction studies, the objective of the work effort encapsulated in this technical note is to review and select the most promising practice oriented and state-of-the-art approaches that robustly quantify H&H model uncertainty. Upon selection, the intent is to move forward with the previously mentioned planned future related work activities, to include software development, preparation of practice oriented guidance documentation, and research and development directed at improving the efficiency of the algorithm(s). While not inclusive of all available methods, the error/uncertainty analysis methodologies summarized in the review section of this document include Bayesian MCMC, Bayesian model averaging (BMA), Data Assimilation (DA), ANNs as a surrogate for the forward hydrologic/environmental model, the popular GLUE method, linear and nonlinear post calibration predictive error variance analysis, post subspace
regularized inversion calibration constrained Monte Carlo, and the use of a modified state-of-the-art evolutionary strategy designed to track the Pareto front to evaluate the tradeoffs associated with regularized inversion and also model prediction.

Bayesian MCMC, and in particular DREAM (Vrugt et al. 2008a, 2009), and/or its basis, DE-MC (ter Braak 2006), was selected as the state-of-the-art method for estimating model parameter and predictive uncertainty. The demonstrated capacity of DREAM/DE-MC to be complete with respect to its treatment of uncertainty, its acceptance within the hydrologic modeling (research) community, its technical soundness, and also the noted simplicity and ease of implementation of the method were all factors influencing the decision. Future related research and development activities will consider ANNs, among possible other surrogates, as a means to reduce the computational expense associated with the application of DREAM/DE-MC. Both BMA and DA were not selected as the state-of-the-art pathway forward due to the anticipated likely costs associated with their implementation and/or use. Water resources planning and management and hydrologic design are the primary application settings which focus the content of this document. DA systems; i.e., the algorithms for updating state and also the necessary infrastructure to receive and process new state information as it becomes available, are typically employed to support operational forecasting environments.

While very popular, in part likely due to its simplicity, the expressed concerns regarding the GLUE method (Beven and Binley 1992), including its subjectivity, lack of statistical validity, the fact that all sources of uncertainty get mapped onto parameter space, and its sampling inefficiencies, among others, lead to it not being selected as the practice driven approach path forward for assessing model uncertainty for USACE planning and design studies.

Recent work (Tonkin et al. 2007; Tonkin and Doherty 2009; Moore et al. 2010) has introduced means to compute model predictive error for highly parameterized models. These methods do not presuppose a uniquely identifiable parsimonious model. Rather, model complexity is included and retained because it is assumed or known to be of importance and a mathematical regularization methodology is employed to stabilize and guide the model calibration. With these methods, rather than use the Bayes equation to estimate model uncertainty, following regularized model inversion (calibration), the degree to which a specific model prediction is wrong is quantified. From a computational perspective, these approaches are in general more amenable for practical application than Bayesian-based methods and therefore serve as a current practice oriented path forward for future software development, preparation of related guidance, and evaluation of opportunities for algorithm enhancement.

In summary, several current state-of-the-art and practice-driven approaches for estimating model error/uncertainty were evaluated and the DREAM/DE-MC method and methods based on post calibration predictive error variance analysis were selected as the pathways forward to serve as the basis for future software development, preparation of related guidance documentation, and research and development activities directed to improving the computational efficiency of the algorithms.

The previously mentioned planned future research and development activity will initially focus on efficiency enhancements of the native algorithms (i.e., DREAM and/or DE-MC; and also one or more of the methods outlined by Tonkin et al. (2007), Tonkin and Doherty (2009), or Moore
et al. (2010)), in the spirit of the works of Skahill et al. (2009), Baggett and Skahill (2010), and Skahill and Baggett (2012). Subsequently, there will be examination of means to improve their computational load via utilization of multiple processors.

ACKNOWLEDGMENTS: This technical note was externally reviewed by David Welter, South Florida Watershed Management District, and internally reviewed by Drs. Aaron Byrd, Matthew Farthing, and Nawa Pradhan from the Coastal and Hydraulics Laboratory at the ERDC for the accuracy of its content and also for suggestions to improve upon its breadth and clarity. The author thankfully acknowledges their review comments and suggestions which improved the content and clarity of this final draft of the document.

ADDITIONAL INFORMATION: For additional information, contact Dr. Brian E. Skahill, at Brian.E.Skahill@usace.army.mil. This CHETN should be cited as follows:


REFERENCES


NOTE: The contents of this technical note are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such products.