HYDROLOGICAL PROCESSES
Hydrol. Process. **30**, 2703–2716 (2016)
Published online 17 March 2016 in Wiley Online Library
(wileyonlinelibrary.com) DOI: 10.1002/hyp.10799

Efficient estimation of flood forecast prediction intervals via single- and multi-objective versions of the LUBE method

Lei Ye,[1,2] Jianzhong Zhou,[1,2]* Hoshin V. Gupta,[3] Hairong Zhang,[1,2] Xiaofan Zeng[1,2] and Lu Chen[1,2]

[1] *School of Hydropower and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China*
[2] *Hubei Key Laboratory of Digital Valley Science and Technology, Wuhan 430074, China*
[3] *Department of Hydrology and Water Resources, The University of Arizona, 1133 E North Campus Dr., Tucson, AZ, 85721, USA*

Abstract:

Prediction intervals (PIs) are commonly used to quantify the accuracy and precision of a forecast. However, traditional ways to construct PIs typically require strong assumptions about data distribution and involve a large computational burden. Here, we improve upon the recent proposed Lower Upper Bound Estimation method and extend it to a multi-objective framework. The proposed methods are demonstrated using a real-world flood forecasting case study for the upper Yangtze River Watershed. Results indicate that the proposed methods are able to efficiently construct appropriate PIs, while outperforming other methods including the widely used Generalized Likelihood Uncertainty Estimation approach. Copyright © 2016 John Wiley & Sons, Ltd.

KEY WORDS prediction interval; uncertainty; LUBE; flood forecasting; multi-objective; artificial neural networks

Received 21 May 2015; Accepted 22 January 2016

INTRODUCTION

The concept of artificial neurons, first introduced by McCulloch and Pitts (1943), is inspired by a desire to understand the human brain and mimic its functioning (Chen *et al.*, 2014a). The use of *artificial neural network* (ANN) methods to develop *deterministic* forecasts has been shown to provide promising results with good computational efficiency, and without the need to have detailed knowledge regarding the nature of the underlying relevant physical processes. However, deterministic forecasts can be of limited general value if they are not accompanied by information about the intrinsic level of forecast uncertainty (Goodwin *et al.*, 2010).

One approach to quantifying such uncertainty is to construct estimates of upper and lower bound prediction intervals (PIs) that indicate the range within which the observed data are likely to occur with some probabilistic level of confidence (such as 90%). These are typically reported as $(1-\alpha)\%$ confidence levels, where α represents the probability of the target variable lying outside the PIs. PIs are similar to confidence intervals (CIs), but with the distinction that CIs are associated with the uncertainty in the prediction of an unknown but fixed value, whereas PIs are assigned to a random variable yet to be observed (De Gooijer and Hyndman, 2006). Because they also account for model misspecification and noise variance, by definition PIs enclose CIs of corresponding confidence levels (Taormina and Chau, 2015). Not only does the PI provide a range that the observed data are likely to occur within, it also provides an indication of the accuracy and precision of the forecast (Quan *et al.*, 2014).

Several ANN-based methods for the construction of PIs have been proposed in the literature. While they differ in manner of implementation, the approaches are generally rather similar – first train an ANN model through minimization of an error-based function such as sum of squared errors, and subsequently construct a PI for the ANNs model outputs (Kasiviswanathan *et al.*, 2013). For instance, the delta technique introduced by Chryssolouris *et al.* (1996) linearizes the ANN model around a set of parameters obtained through minimization of the sum of squared error cost function, and standard asymptotic theory is then applied to the linearized model to construct the PI (Seber and Wild, 1989). However, the method is based on an assumption that the noise is homogenous and normally distributed, which may not generally be true in real-world problems (Ding and He, 2003).

The Bayesian technique is another approach for constructing PIs, in which each parameter is represented

*Correspondence to: Name: Jianzhong Zhou, School of Hydropower and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China.
E-mail: jz.zhou@hust.edu.cn

using a probability distribution rather than a single value, and therefore, the model output will be a distribution conditioned on the observed data (Ye *et al.*, 2014). However, despite the strength of the supporting theory, the Bayesian approach requires the computation of the Hessian matrix at each iteration, which in turn causes singularity problems that may harm the quality of the PIs while incurring heavy computational costs (Taormina and Chau, 2015).

The *Generalized Likelihood Uncertainty Estimation* (GLUE) method proposed by Beven and Binley (1992) is based on Monte Carlo simulation, where a large number of model runs are conducted, each with an assigned prior probability and random parameter values selected from a probability distribution for each parameter (Aronica *et al.*, 2002). The acceptability of each run is evaluated against observed values, and if the acceptability is below a certain subjective threshold, the run is considered to be non-behavioural and the associated parameter combination is removed from further analysis (Li *et al.*, 2010). Although GLUE is relatively easy to implement, it has evident shortcomings, including the subjective choice of Likelihood function and truncation threshold used to separate behavioural and non-behavioural models.

The bootstrap method is a computational procedure that uses intensive resampling with replacement to estimate the uncertainty associated with a model forecast (Efron and Tibshirani, 1994). The method generates different realizations of a dataset to create bootstrap samples, from which the mean and variability of the estimates can be computed (Tiwari and Chatterjee, 2011; Chen *et al.*, 2015a). It has the advantage of simplicity and ease of implementation, but makes the assumption that an ensemble of models will produce a less biased estimate of the true regression of the targets.

To overcome problems associated with traditional methods for constructing PIs, Khosravi *et al.* (2011) proposed the '*Lower Upper Bound Estimation*' (LUBE) method for constructing PIs. Unlike traditional ANN-based methods that compute only one output for simulating the variable of interest, the LUBE approach employs an ANN model having two outputs to directly construct estimates of the upper and lower bounds of the PI. A univariate cost function considering both '*coverage probability*' and '*width*' is used to optimize the properties of the PI. The parameters of the ANN model are then adjusted via single-objective optimization. Advantages of this approach include the following: (i) it does not require that assumptions be made regarding data or error distributions, (ii) it is computationally inexpensive to implement, and (iii) it has been demonstrated to be simpler, faster, and more reliable than traditional techniques (Khosravi *et al.*, 2011).

Notwithstanding its advantages, the LUBE method suffers from several problems. Because it makes use of a compound coverage width-based criterion (CWC) cost function that simultaneously considers the coverage probability and width of the PI, a situation can arise wherein the minimum CWC (zero) is achieved by finding a zero width index (in which case the coverage probability of PI is invalid and non-informative). A more minor issue is that the constraint linking the first and second outputs strictly to the upper and lower bounds of the PI respectively is unnecessary and limits the feasible solution space. Perhaps, the most serious issue is that the conventional CWC cost function is highly nonlinear, complex, discontinuous, and non-differentiable, making it difficult to optimize via classical local optimization algorithms.

It is important to note that the goals of maximizing coverage probability while minimizing PI width are conflicting (higher coverage probabilities typically result in wider PIs) (Ye *et al.*, 2014). The conventional CWC cost function converts what is really a multi-objective problem to a scalarized single-objective one. To achieve this, the scalarization coefficients must be properly selected, a difficult problem requiring an iterative trial-and-error approach. Further, as indicated in the preceding texts, combining the two conflicting objectives into the single CWC cost function results in a highly nonlinear, complex, discontinuous, and non-differentiable problem that is difficult to solve. Implementing a multi-objective formulation of LUBE would help to avoid these problems.

In this paper we propose and compare two improved versions of the LUBE method for constructing flood forecast PIs; these can be useful for a variety of applications including flood prevention, disaster relief, and public safety. The first version provides an improved single-objective formulation of LUBE that can be solved in the standard manner. In this version, the CWC cost function is reformulated by introducing an additional hyper-parameter to achieve an appropriate trade-off between coverage probability and PI width, while a constant 1 is added to the width-based index to prevent the generation of very narrow PIs that do not bracket any of the observed data. We also remove the constraint linking the first and second outputs strictly to the upper and lower bounds of the PI respectively, and instead compute two outputs at each time step such that the larger value is used as the upper bound and the smaller value is used as the lower bound. Calibration of the ANN model is achieved using the Shuffled Complex Evolution (SCE) algorithm developed by Duan *et al.* (1992).

The second version is a multi-objective implementation of LUBE that provides, in a single optimization run, a Pareto optimal set of so-called '*differently good solutions*' without the need to formulate a CWC cost function and preselect scalarization coefficients. By maintaining inde-

pendence of the objective functions, multi-objective analysis clearly reveals the trade-offs involved in the decision process (Gupta *et al.*, 1998). By examining these trade-offs, the user can select a suitable weighting of objectives that provides a robust solution to the decision problem. As we will see, the multi-objective implementation tends to provide solutions that are superior to the single-objective formulation.

The remainder of this paper is organized as follows: the *Indices for Evaluating Prediction Intervals* section introduces the indices used to evaluate the quality of a PI, the *Improvements to the Single-Objective LUBE Method* section discusses our proposed improvements to the single-objective LUBE method, and the *Multi-objective LUBE Method* section describes the new multi-objective implementation. In the *Case Study* section, a real-world case study with results and discussions is presented. The last section summarizes the conclusions of this study.

INDICES FOR EVALUATING PREDICTION INTERVALS

The fundamental property of a PI is the Prediction Interval Coverage Probability (PICP) defined as

$$\mathrm{PICP} = \left(\frac{1}{n}\sum_{i=1}^{n} c_i\right)*100\% \tag{1}$$

where n is the total number of samples, U_i and L_i are the upper and lower bound estimates of the ith sample, and y_i is the observed value of ith sample. The quantity $c_i = 1$ if the observed values of the target $y_i \in [L_i, U_i]$, and otherwise $c_i = 0$. The ideal case is PICP = 100%, in which case all of the observed values of target fall within the prediction band.

It is easy to see that a 100% PICP can be easily achieved if the width of the PI is made large enough. However, an overly wide PI conveys poor information about the value of the target. Ideally, we would like the PI to bracket the target while having as narrow a *width* as possible. Previously proposed measures of the average PI width include the Prediction Interval Normalized Average Width (PINAW) and Prediction Interval Normalized Root-mean-square Width (PINRW) defined as

$$\mathrm{PINAW} = \frac{1}{nR}\sum_{i=1}^{n}(U_i - L_i)*100\% \tag{2}$$

$$\mathrm{PINRW} = \frac{1}{R}\sqrt{\frac{1}{n}\sum_{i=1}^{n}(U_i - L_i)^2}*100\% \tag{3}$$

where U_i, L_i, and n are the same as in PICP and R is the range of the target variable. Note, however, that both of these formulations include only information regarding the overall range of the target variable. Here, we propose instead the dimensionless Prediction Interval Average Relative Width (PIARW) to provide improved information about the target variable at each of the time steps, defined as

$$\mathrm{PIARW} = \frac{1}{n}\sum_{i=1}^{n}\frac{U_i - L_i}{y_i}*100\% \tag{4}$$

where U_i, L_i, n, and y_i are the same as in PICP.

IMPROVEMENTS TO THE SINGLE-OBJECTIVE LUBE METHOD

Coverage width-based criterion cost function

In constructing an optimal estimate of the PI, the preferred result is a higher coverage probability (PICP) and narrower width (PIARW), which tend to conflict. Previous implementations of LUBE have implemented two types of CWC cost functions defined as follows:

$$\mathrm{CWC}_{original} = \mathrm{PINAW}\left(1 + \gamma(\mathrm{PICP})e^{-\eta(\mathrm{PICP}-\mu)}\right) \tag{5}$$

$$\mathrm{CWC}_{Quan} = \mathrm{PINRW}\left(1 + \gamma(\mathrm{PICP})e^{-\eta(\mathrm{PICP}-\mu)}\right) \tag{6}$$

where $\mathrm{CWC}_{Original}$ was used in Khosravi *et al.* (2011) and CWC_{Quan} was used in Quan *et al.* (2014). In these formulations, the quantities μ and η are hyper-parameters that determine the manner by which the penalty term ($e^{-\eta(\mathrm{PICP}-\mu)}$) acts to penalize solutions having low coverage probability. Here μ is the nominal confidence level associated with the PI and can be set to $[(1-\alpha)\%]$. Meanwhile, η exponentially magnifies the difference between the PICP and μ. During the calibration period, γ(PICP) is set to the constant value 1, while in the evaluation period, γ(PICP) is a step function that depends entirely on the acquired PICP.

$$\gamma(\mathrm{PICP}) = \begin{cases} 0 & \mathrm{PICP} \geq \mu \\ 1 & \mathrm{PICP} < \mu \end{cases} \tag{7}$$

As long as the PICP reaches the nominal confidence level μ, the PI can be regarded as valid. Therefore, the exponential term is eliminated during the evaluation period whenever PICP $\geq \mu$.

A major problem with these two forms of CWC is that only a single hyper-parameter η is available to control the strength and nature of the penalty, thereby making it difficult to achieve an appropriate trade-off between coverage probability and PI width. Accordingly, we propose the improved formulation:

$$\mathrm{CWC}_{Proposed} = (1 + \eta_1 \mathrm{PIARW})\left(1 + \gamma(\mathrm{PICP})e^{-\eta_2(\mathrm{PICP}-\mu)}\right) \quad (8)$$

in which PIARW is used in place of PINAW and PINRW, while a third hyper-parameter has been added to provide more flexibility in the formulation. Here, hyper-parameter η_1 linearly magnifies the PIARW, and the parameter η_2 exponentially magnifies the difference between the PICP and μ. This allows more weight to be given to the variation of PICP. Further, the reformulation prevents the width-based multiplication factor $(1 + \eta_1 \mathrm{PIARW})$ from falling below a minimum value of 1, thereby effectively avoiding the generation of very narrow PIs that do not racket any of the observed data. In the *Comparison of the three single-objective LUBE methods* section, we illustrate situations where invalid PIs are generated by the previous versions of LUBE because of poor formulation of $\mathrm{CWC}_{Original}$ and CWC_{Quan}.

Artificial neural network architecture

LUBE employs a simple ANN model architecture having two output neurons (Figure 1A) to directly estimate the lower and upper bounds of the PI at each time step (the ANN architecture shown is symbolic and can be adapted to include more hidden layers as necessary for a given application). In this paper, we remove the constraint linking the first and second outputs strictly to the upper and lower bounds of the PI respectively, and instead compute two outputs at each time step such that the larger value is used as the upper bound and the smaller value are used as the lower bound. A typical neuron is shown schematically in Figure 1B. The input vector $\mathbf{X} = (x_1, x_2, \ldots, x_n)$ to each neuron may come either from system inputs or from the outputs of other neurons. The parameters to be determined are $\mathbf{W}$ and b, where $\mathbf{W} = (w_1, w_2, \ldots, w_n)$ represents the connection weights between a neuron and the neurons in the preceding layer, and b is a threshold value or bias. The output of the neuron y is defined as

$$y = f(\mathbf{X} \cdot \mathbf{W} - b) \quad (9)$$

where f is the neural activation function. Here we use the popular sigmoid activation function:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (10)$$

Implementation

To deal with the nonlinear, complex, and non-differentiable nature of $\mathrm{CWC}_{Proposed}$, we use the SCE optimization algorithm developed by Duan *et al.* (1992)

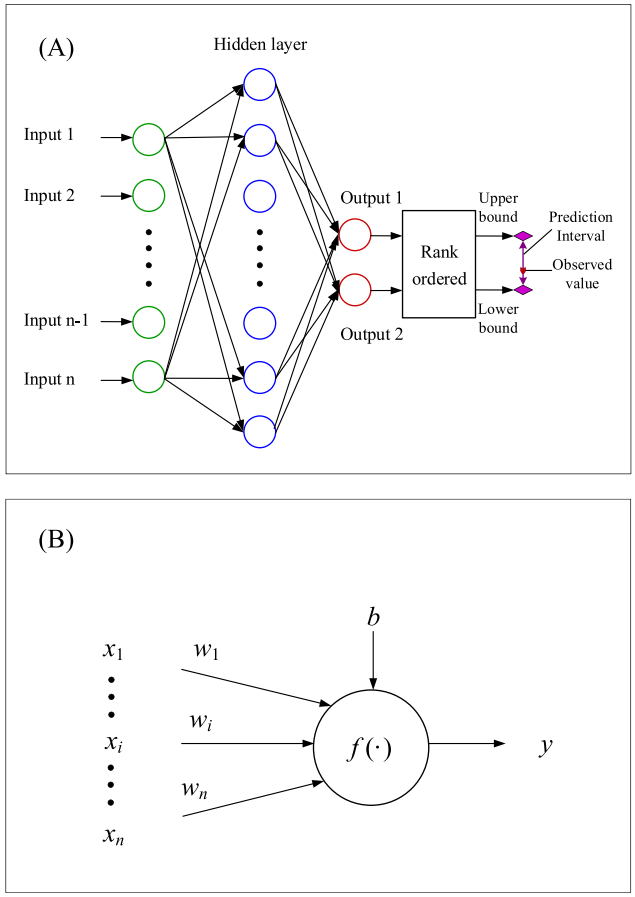

Figure 1. ANN model architecture for constructing lower and upper bounds of PI

to search for optimal values of the ANN model parameters. SCE is an evolutionary-based procedure that simultaneously evolves a population of solutions (parameter sets) towards better solutions in the search space, progressively converging towards the global optimum of the objective function (Blasone *et al.*, 2007). SCE has been widely shown to be effective and efficient for calibration of hydrological models (Madsen, 2003; Ajami *et al.*, 2004; Muttil and Jayawardena, 2008; Zhang *et al.*, 2013), while being superior to other search techniques, such as the Multiple Start Simplex, Genetic Algorithm, and Simulated Annealing (Gan and Biftu, 1996; Cooper *et al.*, 1997; Kuczera, 1997; Franchini *et al.*, 1998). A brief step-by-step description of our implementation of LUBE appears in the Appendix A.

Note that at the start of the calibration process, when PICP is always lower than μ, the cost function $\mathrm{CWC}_{Proposed}$ imposes an extremely heavy penalty because of the dominance of the PICP term, causing the PICP to increase rapidly. As the calibration proceeds, the penalty term exponentially decreases with the increase of PICP. When PICP is greater than but close to μ, the penalty term for PICP is very small, and therefore,

PIARW starts to decrease because it is now the dominant term in $CWC_{Proposed}$. Finally, the values of PICP and PIARW change gradually to achieve the smallest value for $CWC_{Proposed}$. Consequently, $CWC_{Proposed}$ provides a solution that trade-offs between the informativeness and validity of PI, in a manner determined by η_1 and η_2.

MULTI-OBJECTIVE LUBE METHOD

The LUBE approach can also be implemented in a multi-objective fashion so as to obtain a Pareto optimal set of differently good solutions via a single run of a multi-criteria optimization algorithm. Each point in the Pareto optimal set represents a differently 'optimal' solution in terms of the trade-off between PICP and PIARW, from which the user can choose a solution that reflects his/her priorities (without the need to iteratively test different combinations of the CWC hyper-parameters). The ANN model architecture remains the same. An outline of the multi-objective implementation of the LUBE method appears in Appendix B.

A number of different multiple-criteria optimization algorithms are available in the literature. Here we use the Multi-objective Shuffled Complex Differential Evolution (MOSCDE) algorithm, developed by Guo *et al.* (2013) and applied for hydrologic model calibration by Ye *et al.* (2014), to develop an estimate of the Pareto-optimal front. MOSCDE is a multi-objective extension of the SCE (Duan *et al.*, 1992) global optimization algorithm, where MOSCDE employs differential evolution (Storn and Price, 1997) instead of simplex search, and introduces Cauchy mutation (Yao and Liu, 1996) to prevent premature convergence problem. A brief description of MOSCDE appears in the Appendix C.

CASE STUDY

Study area and data

To demonstrate the utility of the single- and multi-objective LUBE methods, we consider the case of flood forecasting for the Yangtze River (Chang Jiang). The Yangtze is the longest river in China and the third longest river in the world and plays a vital role in the economic development and ecological environmental conservation of China. It originates in the Qinghai–Tibet Plateau at an elevation of 6600 m above sea level and flows 6300 km eastward into the East China Sea at Shanghai. The drainage basin lies between 91°E–122°E and 25°N–35°N and covers a total area of 1 808 500 km^2.

This study focuses on the upper Yangtze, which makes up 3/4 of the length of Yangtze River and drains 55.6% of the watershed area (Chen *et al.*, 2015b). The upper Yangtze is composed of a complex of tributaries, principally Jinsha, Min, Tuo, Jialing, and Wu Rivers. The Yalong River joins the Jinsha River, and thus can be viewed as a tributary of the Jinsha. Figure 2 provides a schematic of the regional major tributary rivers and gauging stations. Six hydrological control stations for the five tributaries and mainstream are taken into consideration. From upstream to downstream, these are the Pingshan, Gaochang, Lijiawan, Beibei, Wulong, and Yichang stations. We take the past values of flow at

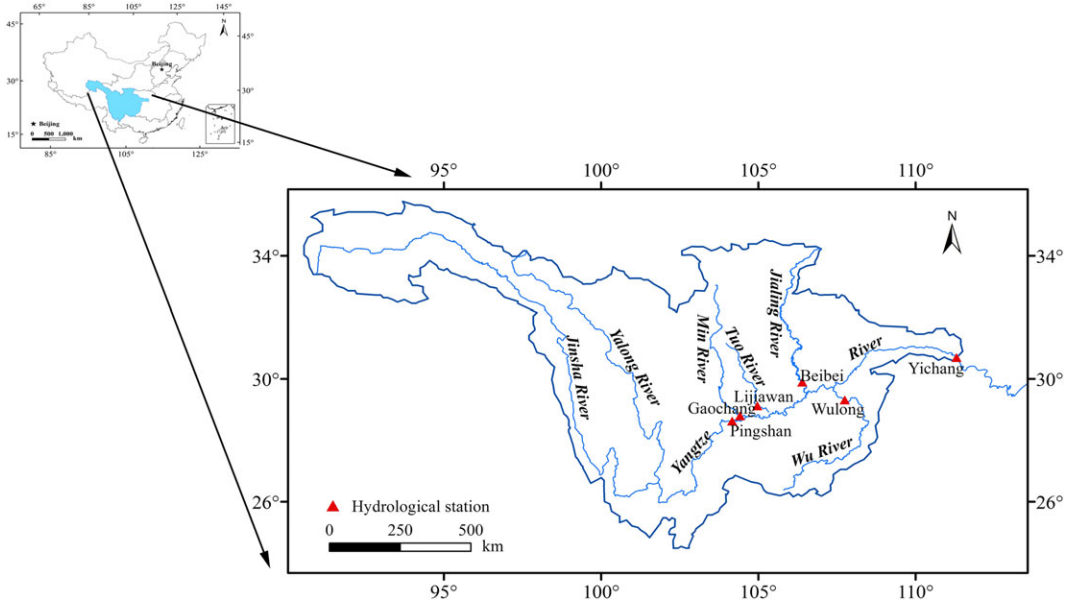

Figure 2. Locations of mainstream and regional tributary rivers as well as their control hydrological stations for the upper Yangtze River

Pingshan, Gaochang, Lijiawan, Beibei, Wulong, and Yichang station as input variables, and the runoff at the Yichang station as the output variable.

The data set used in this study consists of 55 years of daily mean streamflow at the six hydrological control stations during the flood season (June–September) and represents a wide variety of hydrological conditions. The first 35 years (1953 to 1987) were used for model calibration, and the remaining 20 years (1988 to 2007) were used for model evaluation.

ANN model architecture

Based on the copula entropy input determination method described in Chen *et al.* (2014b), the best inputs for use by an ANN model to forecast $Q_{yc,t}$ include $Q_{yc,t-1}$, $Q_{yc,t-2}$, $Q_{ps,t-1}$, $Q_{gc,t-3}$, $Q_{bb,t-2}$, $Q_{wl,t-2}$, and $Q_{ljw,t-3}$, where the subscripts *yc*, *ps*, *gc*, *bb*, *wl*, and *ljw* refer to Yichang, Pingshan, Gaochang, Beibei, Wulong, and Lijiawan hydrological stations respectively, and t represents the time step. The outputs of the ANN model are the upper and lower bounds of $Q_{yc,t}$, and the numbers of input and output neurons are set to seven and two respectively. A simple single hidden layer ANN architecture is used, as more hidden layers will increase the number of parameters and thereby increase the difficulty of parameter estimation. Based on a trial and error approach, the optimal number of hidden neurons was determined to be 3. Therefore, the number of the ANN model parameters to be optimized was 32 (number of hidden neurons + number of output neurons + number of input neurons × number of hidden neurons + number of hidden neurons × number of output neurons).

Parameter settings

The parameter settings used for the SCE algorithm [based on recommendations in Duan *et al.* (1994)] and the $CWC_{Proposed}$ cost function are shown in Table I. Selecting the prescribed PI level of confidence to be 90%, the value of μ was set to 0.9, and η_1 and η_2 were set to 35 and 15 respectively to balance the trade-off between PICP and PIARW. The parameter settings for the MOSCDE algorithm were selected based on Guo *et al.* (2013) as shown in Table II, where *dy* is the number of objective functions, here two (PIARW and PICP), and q, s, ss, S_g, S_c, and *max* were set as described in Appendix A. Please refer to Guo *et al.* (2013) for detailed settings of the MOSCE algorithm. The feasible parameter space for the ANN model parameters (the weights between the neurons and the threshold of the neurons) was set to between −10 and 10, as recommended by Kasiviswanathan *et al.* (2013).

Table I. Parameters settings for the SCE algorithm and the $CWC_{Proposed}$ cost function

Parameters	Parameter	Numerical value
SCE	q	4
	m	65
	s	260
	ss	65
	max	200
$CWC_{Proposed}$	α	0.1
	μ	0.9
	η_1	35
	η_2	15

Results for the single-objective methods

Comparison of the three single-objective LUBE methods. We first compare the results of our single-objective LUBE method (using $CWC_{Proposed}$) to those provided using the $CWC_{Original}$ and CWC_{Quan} cost functions. To ensure a fair comparison, the parameters used to construct the PIs were kept the same for all three cases as far as possible (Table I). However, while $CWC_{Proposed}$ has two parameters (η_1 and η_2) to be specified, the $CWC_{Original}$ and CWC_{Quan} cost functions have only a single η parameter, which was set to be 38.5 and 84 respectively. The η values were chosen to ensure that the PIs generated by all three CWC cost functions have similar PICPs (~94%). This allows us to focus on comparing/evaluating the PI widths generated by the various methods. Performance is compared for both calibration and evaluation periods.

Figure 3 illustrates the progress of the calibration process for the single-objective LUBE method using all three CWC cost functions, showing that all the cost functions have converged to a good result. They decreased rapidly at the beginning and then remained stable for a while before decreasing again to gradually converge to their optimal value.

The results obtained using all three single-objective methods are presented in Tables III and IV. The four measures PICP, PIARW, PINRW, and PINAW are used to assess performance in terms of both coverage probability and PI width. The best forecast value in each column is bolded. Table III shows that the coverage probability is greater than 90% for all three methods.

Table II. Parameters settings for the MOSCDE algorithm

Parameter	Numerical value
dy	2
q	5
s	300
ss	5
Sg	200
Sc	200
max	200

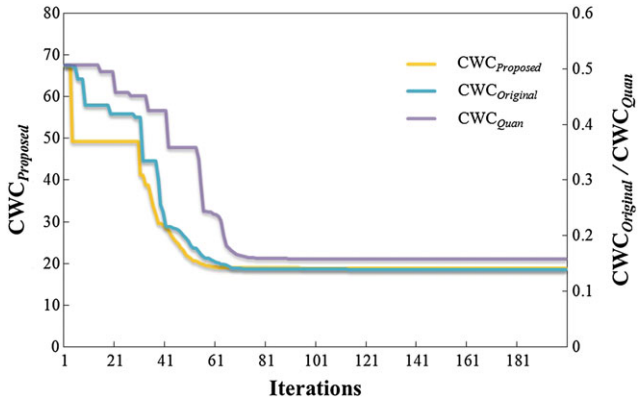

Figure 3. Progress of the calibration process for all three CWC cost functions

Table III. Evaluation indices for PIs generated using the three different CWCs during calibration period

Types of PI	PICP(%)	PINAW(%)	PINRW(%)	PIARW(%)
$PI_{Original}$	94.1	13.1	13.5	35.7
PI_{Quan}	93.4	12.8	13.1	35.4
$PI_{Proposed}$	93.9	12.1	12.8	31.6

Table IV. Evaluation indices for PIs generated using the three different CWCs during evaluation period

Types of PI	PICP(%)	PINAW(%)	PINRW(%)	PIARW(%)
$PI_{Original}$	91.1	14.4	14.8	36.2
PI_{Quan}	92.8	14.2	14.7	36.4
$PI_{proposed}$	92.8	13.2	14.1	31.9

However, the new method generally provides better PI width in terms of the four evaluation indices (Table III). Our results indicate that the new $PI_{Proposed}$ is better than PI_{Quan} in terms of all four of the evaluation indices. Although the coverage probability for $PI_{Original}$ (94.1%) is larger than for $PI_{Proposed}$ (93.9%), the difference is very small, while the remaining three indices are better for $PI_{Proposed}$. Because the prescribed confidence level was 90%, any PICP greater than 90% is valid, and so it is not reasonable to balance a slight higher PICP against a much worse PIARW.

During evaluation, the performance of the three methods is similar to that during calibration except that the coverage probability for $PI_{Proposed}$ is now the highest (Table IV). Overall, the new method provides better forecasting results in both calibration and evaluation, having a narrower width and a satisfactory coverage probability.

It is important to note that in contrast to the new method, the previous methods (using CWC_{Quan} and $CWC_{Original}$) have a tendency to generate invalid PIs when searching for an appropriate value for η. These results, presented in Table V, all have a zero coverage probability (PIs that include no observed data) and extremely narrow width (in one case equal to zero), illustrating our point that small (optimal) values for

Table V. Evaluation indices for unacceptable PIs generated using the $CWC_{Original}$ and CWC_{Quan}

CWC cost functions	η	PICP(%)	PINAW(%)	PINRW(%)	PIARW(%)	CWC	$CWC_{Proposed}$
$CWC_{Original}$	20	0	2.8×10^{-11}	3.0×10^{-11}	8.7×10^{-11}	1.9×10^{-5}	7.3×10^{5}
	25	0	2.9×10^{-10}	1.5×10^{-9}	8.4×10^{-10}	1.7×10^{-2}	7.3×10^{5}
CWC_{Quan}	10	0	0	0	0	0	7.3×10^{5}
	15	0	1.1×10^{-10}	1.1×10^{-10}	3.2×10^{-10}	7.7×10^{-5}	7.3×10^{5}

CWC_{Quan} or $CWC_{Original}$ can be achieved by having an extremely narrow PI width, even when the coverage probability is zero. The problem clearly lies in the previous formulations of the CWC cost function.

Comparison with the GLUE method. We also compare the new single-objective LUBE method with the GLUE method for uncertainty estimation and PI construction. For GLUE, the number of the behavioural parameter sets was set to 2000 and the confidence level was set to 85% to provide a coverage probability comparable with that of the LUBE method. This was achieved by selecting a behavioural threshold such that the Nash–Sutcliffe efficiency is equal to 0.55.

The GLUE results are presented in Table VI from which it can be seen that in terms of our evaluation indices, the performance of GLUE does not compare favourably with the performance of the three different LUBE methods. Overall, while the coverage probabilities for calibration and evaluation are comparable with those of the three different LUBE methods (slightly lower for calibration and slightly higher for evaluation), the GLUE-based PIs are much wider. In fact, the PIARW reached 58.1%, which is far too wide and offers limited information for decision makers. We note also that the GLUE method is time-consuming, because it is cumbersome to obtain 2000 randomly sampled parameter sets from the feasible parameter space that have Nash–Sutcliffe efficiency greater than 0.55.

Analysis of the prediction intervals. Figure 4 displays the four different PIs of the hydrograph generated using the three different CWC cost functions and GLUE method along with the observed flows. Because the evaluation period is very long, we plot the hydrograph for only the two years 1998 and 2004, in which the entire Yangtze River basin suffered from severe flooding. As can be seen, the $PI_{Original}$ and PI_{Quan} severely underestimate the high streamflow periods, with most of the high streamflow values lying outside the PIs (especially in 1988, which is the most severe flood year during the evaluation period).

The reason for this underestimation of high streamflow lies in the choice of width-based indices. The width-based indices (PINRW and PINAW) for generating $PI_{Original}$ and PI_{Quan} calculate the absolute width $(U_i - L_i)$. For periods of high streamflow, the PI bracketing the observed streamflow is usually wider than that obtained for low streamflow periods. Therefore, during the computation of PINRW and PINAW, the wider PI during high streamflow periods imposes a higher penalty than during the medium and low streamflow periods. Meanwhile, bracketing a single high streamflow value adds only 'one' to the number of covered values. Therefore, because there are far fewer high streamflow values than low values, the $PI_{Original}$ and PI_{Quat} tend to more strongly emphasize the low streamflow periods.

In contrast, as seen in Figure 4, the new method provides for better coverage of the high flood events, while providing narrower PIs than those obtained using the previous methods. Meanwhile, although the GLUE-based PIs cover most of the observed streamflow data, they tend to be much wider than provided by the LUBE methods (especially in 1998), and are therefore less informative.

Results for the multi-objective methods

Figure 5 shows the progress of the MOSCDE optimization search for the multi-objective LUBE method, in which the ANN model parameters were trained using two simultaneous objectives, PIARW and PICP. The search is terminated after 200 generations. Note that the number of Pareto optimal points increases steadily during the search, with the initial generation having only 42 unevenly distributed Pareto optimal values, while the 100th generation already has 200 Pareto optimal points (equal to the size of the global archive) evenly distributed across the front. Most of the improvement occurs during the first 50 generations, with relatively minor improvements thereafter, and it is clear that further iterations are not likely to result in any significant improvement.

The results show clearly that PIARW and PICP are conflicting objectives. Each point on the final (200th generation) Pareto-optimal front represents a different '*optimal*' solution, in the multi-objective sense that all of these solutions are '*differently good*'. The solutions are distributed evenly across the Pareto front, and a user can pick any solution that reflects a given relative priority regarding the trade-off between improving the two objectives PIARW and PICP. Note that a 95.8% PI coverage probability can be achieved by selecting the PIARW to be about 37.3% (Point A).

Table VI. Evaluation indices for PI generated by the GLUE method

Period	PICP(%)	PINAW(%)	PINRW(%)	PIARW(%)
Calibration	92.8	19.9	20.9	58.1
Evaluation	93.7	22.4	23.5	61.1

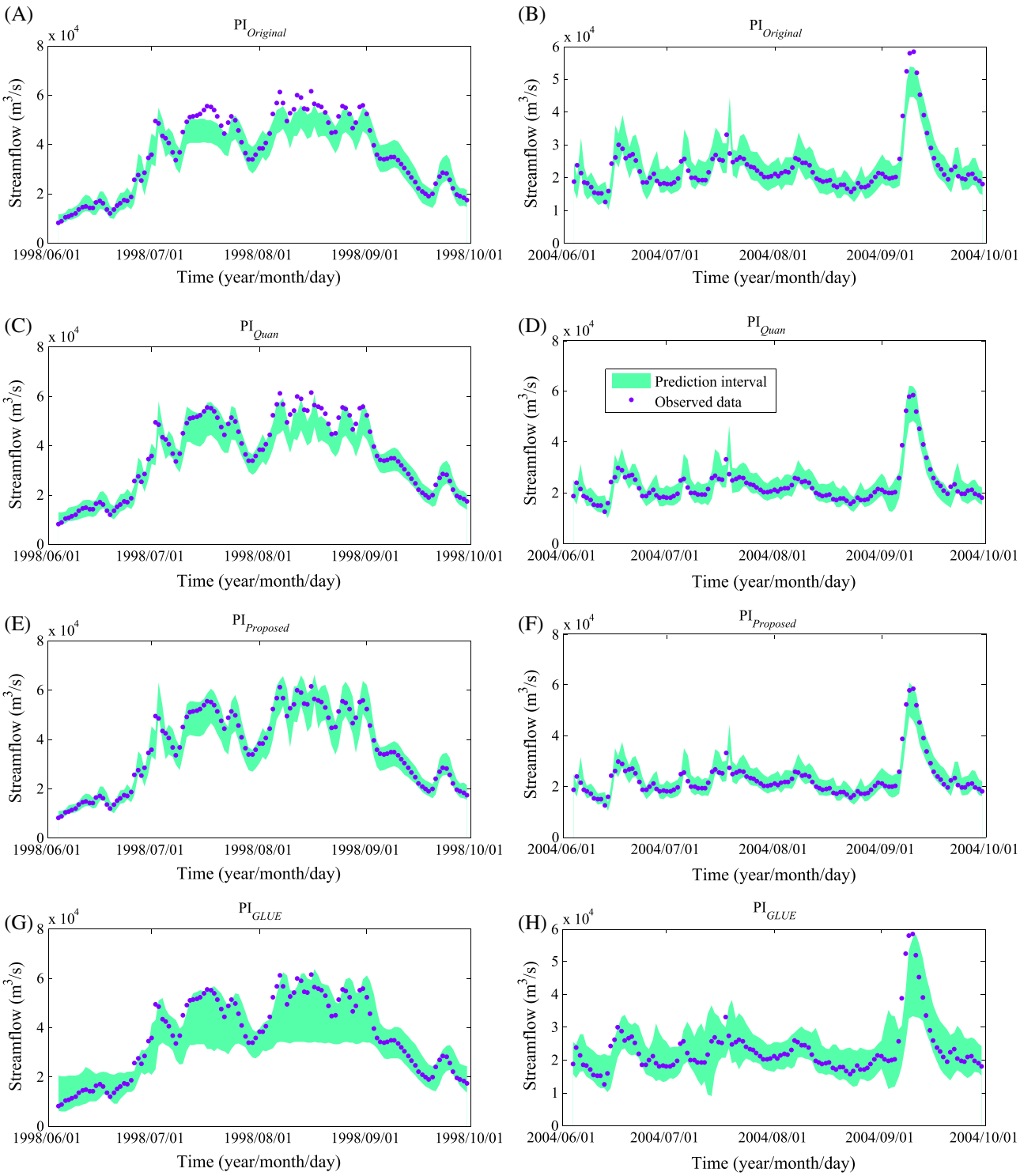

Figure 4. Four different PIs of the hydrograph generated using the three different CWC cost functions and GLUE method along with the observed flows

Figure 6 shows the PIs obtained for Point A and two other example points (Points B and C) on the Pareto front having higher and lower coverage probability (98.7% and 90.3%) respectively. While $PI_{Point\ B}$ has a smaller number of observed flows outside the prediction band than $PI_{Point\ A}$, it also has a wider PIARW. Conversely, $PI_{Point\ C}$ has a narrower PIARW, but a larger number of observed flows outside the prediction band. Because these points are indistinguishable in a multi-criteria sense, users can choose any of them according to their own preferences

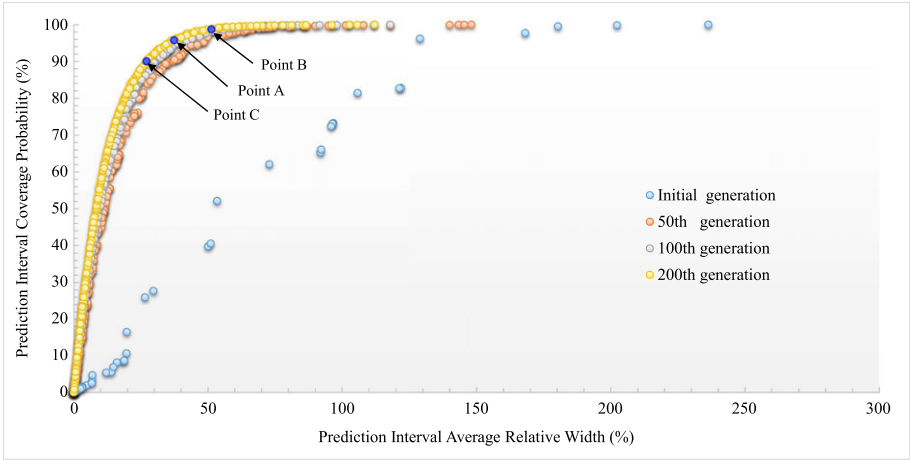

Figure 5. Variations of multi-objective Pareto optimal front during calibration

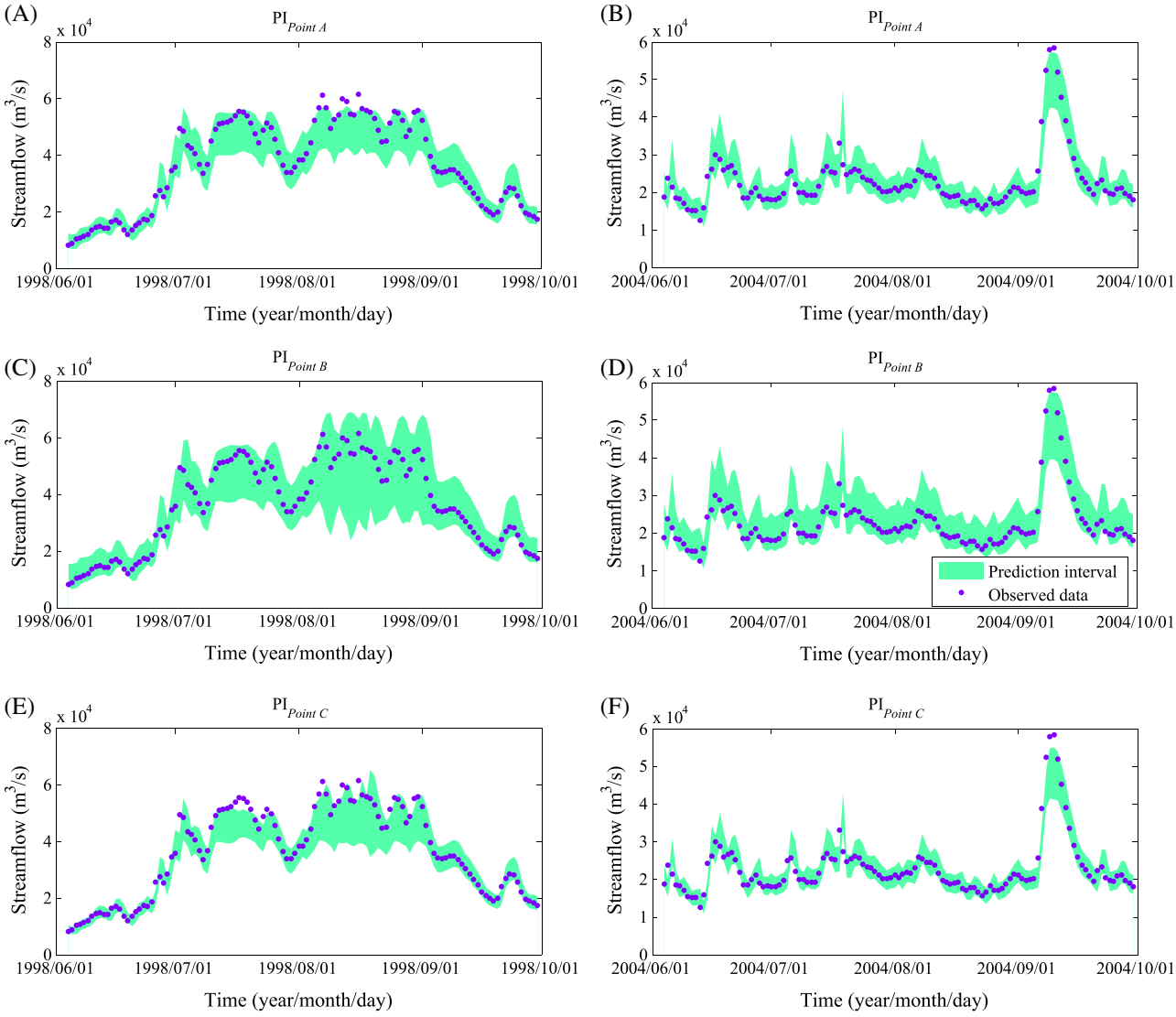

Figure 6. PIs corresponding to the three selected points along with the observed flows

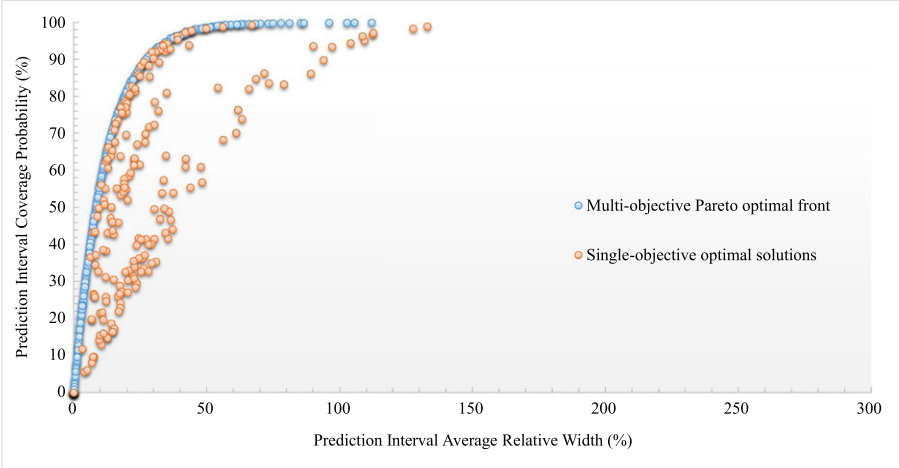

Figure 7. Comparison between multi-objective Pareto optimal front (200 points) and 200 single-objective optimal solutions

regarding the trade-off between objectives. Note that $PI_{Point\ A}$ has a medium value for PIARW and most of the observed flows fall within the PI (except for a few extremely high flow values). $PI_{Point\ A}$ can be considered comparable with $PI_{Proposed}$ in the *Results for the single-objective methods* section because they have similar PIARW and PICP. Note that for the year 1998, the $PI_{Proposed}$ appears to be slightly better, but this is misleading since, for brevity, we have only plotted two years data.

For completeness, we also compare the multi-objective results to an approach where we randomly selected 200 sets of values for the $CWC_{Proposed}$ cost function parameters (μ, η_1, and η_2), and then calibrated the ANN model parameters for each case using the SCE single-criterion global optimization algorithm. Figure 7 compares the PIARW and PICP solutions so obtained with the Pareto-optimal front (200 points) generated using the multi-objective LUBE method. As can be seen, the points generated by the single-objective approach are not evenly distributed, and only a few are close to the Pareto front. Clearly, a user would have to perform many trial-and-error attempts to obtain values for the $CWC_{Proposed}$ cost function parameters that provide optimal performance in terms of both PIARW and PICP. Further, the very real possibility of premature convergence at a local optimum could cause the single-objective CWC solutions to terminate prior to reaching the Pareto optimal front.

It is clear from this analysis that extension of the LUBE method to the multi-objective framework can provide superior results. While there is an additional complexity associated with implementation of a multi-objective optimization algorithm, the challenge of finding an appropriate optimal PI solution in terms of both PICP and PIARW is greatly reduced.

CONCLUSIONS

This paper has developed and tested two improvements to the recently proposed LUBE method for constructing PIs for flood forecasting. The method uses an ANN model having two outputs to construct upper and lower bounds of the PI.

Implementation of the LUBE method is based on the simultaneous optimization of two complementary features of any PI: the coverage probability and the width. Our first improvement is the development of a new CWC for use in single-objective implementations of the method. Our second improvement is to implement LUBE within a multi-objective framework that provides, with a single optimization run, a comprehensive set of solutions that represent different trade-offs in the simultaneous maximization of probability of coverage and minimization of width of the PIs. Our results indicate the following:

1. The newly formulated CWC overcomes some important defects associated with previous CWCs. By replacing the previous width indices with the PIARW index, we are able to more completely utilize the available information about the target variable. Further, the PIARW is better suited to streamflow forecasting as it is based on a calculation of the relative (rather than absolute) width. Further, the new formulation prevents the selection of degenerate solutions having zero PI width.
2. The improved single-objective LUBE method provides higher-quality PIs than does earlier versions of the LUBE method or the GLUE method.
3. The multi-objective implementation of LUBE makes it much easier for a decision maker to obtain a feasible PI solution that has an appropriate probability of coverage and width to suit the intended application, dramatically

reducing the effort associated with trial-and-error implementation of the single-objective approach.

Importantly, the LUBE method is generally applicable to a broad range of engineering applications, and the system model does not have to be an ANN. Further, the method does not require that any assumptions be made regarding the joint probability distribution of the data.

Finally, as with any new method, definitive conclusions regarding its superiority (or otherwise) to other approaches must be based on exhaustive testing using large samples of catchments representing a wide range of hydrometeorological and hydrogeological conditions (Gupta *et al.*, 2014). In this work we have selected the Yangtze River basin as being typical of Chinese basins, being that it covers 18.8% of the total area in China and has 55 years of available daily mean streamflow, so that it represents a wide variety of hydrological conditions. Nonetheless, our future research will involve more extensive testing of the LUBE method using a broad range of applications, model types, and physical situations.

ACKNOWLEDGEMENTS

This work was supported by the Key Program of the Major Research Plan of the National Natural Science Foundation of China (No. 91547208), the State Key Program of National Natural Science of China (No. 51239004) and the National Natural Science Foundation of China (No. 51579107, 51309105, 51309104). The third author received partial support from the Australian Research Council through the Centre of Excellence for Climate System Science (No. CE110001028) and from the EU-funded project 'Sustainable Water Action: Building Research Links Between EU and US' (INCO-20011-7.6 No. 294947). We especially thank Mr Hao Quan (National University of Singapore) and Dr Dongwei Gui (Cele National Station of Observation and Research for Desert-Grassland Ecosystem in Xinjiang) for their valuable suggestions and support.

REFERENCES

Ajami NK, Gupta H, Wagener T, Sorooshian S. 2004. Calibration of a semi-distributed hydrologic model for streamflow estimation along a river system. *Journal of Hydrology* **298**: 112–135.

Aronica G, Bates P, Horritt M. 2002. Assessing the uncertainty in distributed model predictions using observed binary pattern information within GLUE. *Hydrological Processes* **16**: 2001–2016.

Beven K, Binley A. 1992. The future of distributed models: model calibration and uncertainty prediction. *Hydrological Processes* **6**: 279–298.

Blasone R-S, Madsen H, Rosbjerg D. 2007. Parameter estimation in distributed hydrological modelling: comparison of global and local optimisation techniques. *Nordic Hydrology* **38**: 451–476.

Chen L, Singh VP, Guo S, Zhou J, Ye L. 2014a. Copula entropy coupled with artificial neural network for rainfall–runoff simulation. *Stochastic Environmental Research and Risk Assessment* **28**: 1755–1767.

Chen L, Singh VP, Guo S, Zhou J, Zhang J, Liu P. 2015a. An objective method for partitioning the entire flood season into multiple subseasons. *Journal of Hydrology* **528**: 621–630.

Chen L, Ye L, Singh VP, Zhou J, Guo S. 2014b. Determination of input for artificial neural networks for flood forecasting using the copula entropy method. *Journal of Hydrologic Engineering* **19**: 04014021.

Chen L, Zhang Y, Zhou J, Singh VP, Guo S, Zhang J. 2015b. Real-time error correction method combined with combination flood forecasting technique for improving the accuracy of flood forecasting. *Journal of Hydrology* **521**: 157–169.

Chryssolouris G, Lee M, Ramsey A. 1996. Confidence interval prediction for neural network models. *IEEE Transactions on Neural Networks* **7**: 229–232.

Cooper V, Nguyen V, Nicell J. 1997. Evaluation of global optimization methods for conceptual rainfall-runoff model calibration. *Water Science and Technology* **36**: 53–60.

De Gooijer JG, Hyndman RJ. 2006. 25 years of time series forecasting. *International Journal of Forecasting* **22**: 443–473.

Ding AA, He X. 2003. Backpropagation of pseudo-errors: neural networks that are adaptive to heterogeneous noise. *IEEE Transactions on Neural Networks* **14**: 253–262.

Duan Q, Sorooshian S, Gupta V. 1992. Effective and efficient global optimization for conceptual rainfall-runoff models. *Water Resources Research* **28**: 1015–1031.

Duan Q, Sorooshian S, Gupta VK. 1994. Optimal use of the SCE-UA global optimization method for calibrating watershed models. *Journal of Hydrology* **158**: 265–284.

Efron B, Tibshirani RJ. 1994. *An introduction to the bootstrap*. CRC press: New York.

Franchini M, Galeati G, Berra S. 1998. Global optimization techniques for the calibration of conceptual rainfall-runoff models. *Hydrological Sciences Journal* **43**: 443–458.

Gan TY, Biftu GF. 1996. Automatic calibration of conceptual rainfall-runoff models: optimization algorithms, catchment conditions, and model structure. *Water Resources Research* **32**: 3513–3524.

Goodwin P, Önkal D, Thomson M. 2010. Do forecasts expressed as prediction intervals improve production planning decisions? *European Journal of Operational Research* **205**: 195–201.

Guo J, Zhou J, Zou Q, Liu Y, Song L. 2013. A novel multi-objective shuffled complex differential evolution algorithm with application to hydrological model parameter optimization. *Water Resources Management* **27**: 2923–2946.

Gupta H, Perrin C, Bloschl G, Montanari A, Kumar R, Clark M, Andréassian V. 2014. Large-sample hydrology: a need to balance depth with breadth. *Hydrology and Earth System Sciences* **18**: 463–477.

Gupta HV, Sorooshian S, Yapo PO. 1998. Toward improved calibration of hydrologic models: multiple and noncommensurable measures of information. *Water Resources Research* **34**: 751–763.

Kasiviswanathan K, Cibin R, Sudheer K, Chaubey I. 2013. Constructing prediction interval for artificial neural network rainfall runoff models based on ensemble simulations. *Journal of Hydrology* **499**: 275–288.

Khosravi A, Nahavandi S, Creighton D, Atiya AF. 2011. Lower Upper Bound Estimation method for construction of neural network-based prediction intervals. *IEEE Transactions on Neural Networks* **22**: 337–346.

Kuczera G. 1997. Efficient subspace probabilistic parameter optimization for catchment models. *Water Resources Research* **33**: 177–185.

Li L, Xia J, Xu C-Y, Singh V. 2010. Evaluation of the subjective factors of the GLUE method and comparison with the formal Bayesian method in uncertainty assessment of hydrological models. *Journal of Hydrology* **390**: 210–221.

Madsen H. 2003. Parameter estimation in distributed hydrological catchment modelling using automatic calibration with multiple objectives. *Advances in Water Resources* **26**: 205–216.

McCulloch WS, Pitts W. 1943. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics* **5**: 115–133.

Muttil N, Jayawardena A. 2008. Shuffled Complex Evolution model calibrating algorithm: enhancing its robustness and efficiency. *Hydrological Processes* **22**: 4628–4638.

Quan H, Srinivasan D, Khosravi A. 2014. Particle swarm optimization for construction of neural network-based prediction intervals. *Neurocomputing* **127**: 172–180.

Seber G, Wild C. 1989. *Nonlinear regression*. Wiley: New York.
Storn R, Price K. 1997. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization* **11**: 341–359.
Taormina R, Chau K-W. 2015. ANN-based interval forecasting of streamflow discharges\ using the LUBE method and MOFIPS. *Engineering Applications of Artificial Intelligence* **45**: 429–440.
Tiwari MK, Chatterjee C. 2011. A new wavelet—bootstrap—ANN hybrid model for daily discharge forecasting. *Journal of Hydroinformatics* **13**: 500–519.
Yao X, Liu Y. 1996. Fast evolutionary programming. In *Evolutionary Programming*. Citeseer: Cambridge; 451–460.
Ye L, Zhou J, Zeng X, Guo J, Zhang X. 2014. Multi-objective optimization for construction of prediction interval of hydrological models based on ensemble simulations. *Journal of Hydrology* **519**: 925–933.
Zhang R, Santos CA, Moreira M, Freire PK, Corte-Real J. 2013. Automatic calibration of the SHETRAN hydrological modelling system using MSCE. *Water Resources Management* **27**: 4053–4068.

APPENDIX A. BRIEF STEP-BY-STEP DESCRIPTION OF THE PROPOSED LUBE PROCEDURE

Step 1 Split the entire data into calibration and evaluation periods.
Step 2 Select the input variables and architecture of the ANN model.
Step 3 Specify the parameters for the SCE algorithm, including q, the number of complexes; m, the number of points in each complex; s, the population size, equals to qm; ss, the number of evolution steps allowed for each complex before complex shuffling; max, the maximum number of evolutionary generations. In this paper, the parameter settings of SCE were based on the recommendations in Duan *et al.* (1994).
Step 4 Specify the parameters μ, η_1, and η_2 for $CWC_{Proposed}$ cost function.
Step 5 Define feasible ranges for the ANN model parameters and randomly generate an initial population of size s points (ANN model parameters) in the feasible parameter space. Then construct the PI for each point and evaluate the performance in terms of $CWC_{Proposed}$.
Step 6 Sort the s points in order of increasing $CWC_{Proposed}$ value and store them in a set $P = \{P_1, P_2, \cdots, P_s\}$, so that the first point represents the point with the smallest value.
Step 7 Partition the set P into q complexes, $C^1, C^2, \cdots, C^q$, each complex containing m points, such that the first complex contains every $q(j-1)+1$ ranked point, the second complex contains every $q(j-1)+2$ ranked point of P, and so forth, where $j = 1,2,\ldots,m$.
Step 8 Evolve each complex independently by taking ss evolution steps according to the competitive complex evolution (CCE) strategy. For brevity, CCE is not introduced here. Readers may refer to Duan *et al.* (1992) or Blasone *et al.* (2007) for a detailed description of CCE.
Step 9 Shuffle the q complexes into a single sample population and sort the sample population in order of increasing $CWC_{Proposed}$ value.
Step 10 Repeat step 5 to 9 until the max generations.
Step 11 The optimal parameters obtained from the preceding texts are used to construct PI for evaluation period.

The previously mentioned procedures can be repeated several times for different choices of the hyper-parameters μ, η_1, and η_2 so as to obtain a suitable result. But this would not cost many time because the calculation for LUBE method is very fast.

APPENDIX B. OUTLINE OF THE MULTI-OBJECTIVE IMPLEMENTATION OF THE LUBE METHOD

Step 1 Split the entire data into calibration and evaluation periods.
Step 2 Select the input variables and architecture of the ANN model.
Step 3 Select PICP and PIARW as cost functions.
Step 4 Select a suitable multiple-criteria optimization algorithm (Appendix C).
Step 5 Define feasible ranges for the ANN model parameters and randomly generate an initial population in the feasible parameter space.
Step 6 Construct calibration period PIs associated with each point and compute values for the two cost functions.
Step 7 Conduct the multiple-criteria optimization search (Appendix C) to generate a Pareto optimal set of solutions in the feasible parameter space.
Step 8 Construct PIs for evaluation period for each point in the Pareto optimal solution set.

APPENDIX C. BRIEF DESCRIPTION OF MOSCDE ALGORITHM

Step 1 Specify the parameters for the MOSCDE algorithm, the number of complexes q; including the population size s; the number of evolution steps allowed for each complex before complex shuffling ss; the size of global archive set S_g; the size of complex archive set S_c; and the maximum number of evolutionary generations max.

Step 2 Randomly generate s initial points in the feasible space and calculate values for the various objective functions at each point.

Step 3 Assign rank and calculate crowding distance for each point, then sort and store them in set P.

Step 4 Update the global archive set A_g with the Pareto optimal points in P.

Step 5 Partition the s points into q complexes and copy the A_g to q complex archive sets A_i.

Step 6 Apply multi objective differential evolution to each complex for ss generations independently and update complex archive set A_i.

Step 7 Shuffle q complexes into population P and sort the individuals of P based on rank and crowd distance.

Step 8 Update the global archive set A_g with the q complex archive sets A_i ($i = 1, 2, \ldots, q$).

Step 9 Apply the Cauchy mutation operator to A_g every five iterations.

Step 10 Repeat step 4 to 9 until the *max* generations.