Title
Optimal Reservoir Operation Under Inflow Uncertainty

Permalink
https://escholarship.org/uc/item/5g24f86j

Author
Li, Jinshu

Publication Date
2021

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Optimal Reservoir Operation Under Inflow Uncertainty

A dissertation submitted in partial satisfaction of the requirements for the degree

Doctor of Philosophy in Civil Engineering

by

Jinshu Li

2021

ABSTRACT OF THE DISSERTATION

Optimal Reservoir Operation Under Inflow Uncertainty

by

Jinshu Li

Doctor of Philosophy in Civil Engineering

University of California, Los Angeles, 2021

Professor William W. Yeh, Chair

Stochastic programming is a mathematical model used to resolve the uncertainty of random variables in optimization problems. In reservoir management and operation, the reservoir inflow is typically regarded as a random variable as it brings most of the operation uncertainty. Although stochastic programming has been successfully applied to many reservoir managements cases, the pursuit of the improvement on its accuracy, efficiency, and applicability never ceases. This dissertation consists of five chapters. The first introductory presents the classical stochastic model and describes the challenges. Then, the second chapter develops a statistical model that focuses on improving the distribution fitting accuracy for the monthly average inflow as the random variable. The third chapter discusses a method aiming at streamflow scenario tree reduction, which is essential for alleviating the computational burden of a two-stage stochastic programming with recourse model. The fourth chapter expands the applicability of stochastic programming model, by introducing a multi-objective, multi-stage stochastic programming with

recourse model. The final chapter offers conclusions, discussions, and potential future research opportunities.

The dissertation of Jinshu Li is approved.

Mekonnen Gebremichael

Steven Adam Margulis

Ying Nian Wu

William W. Yeh, Committee Chair

University of California, Los Angeles

2021

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 3 Figures

Chapter 4 Figures

LIST OF TABLES

Chapter 3 Tables

Chapter 4 Tables

ACKONWLEDGEMENT

Foremost, I would like to express my sincere gratitude to my advisor Professor William W-G. Yeh, for his guidance, motivation, and the tremendous support on me through my Ph.D. journey. Thank you so much, Prof. Yeh, for the fortunate opportunity and the great honor to work with you.

Then, my deepest thanks go to my parents, for their love, encouraging, and support. Without their constant support and encourage, I would never have a chance to start a PhD. Thank you, mom and dad. I am so fortunate to be your son.

Chapter 2 contains the following reprinted journal article:

Li, J., Zhou, Q. and Yeh, W.W.G., 2020. A Bayesian hierarchical model for estimating the statistical parameters in a three-parameter log-normal distribution for monthly average streamflows. *Journal of Hydrology*, 591, p.125265.

and retains the following copywrite:

Chapter 3 contains the following reprinted journal article:

Li, J., Zhu, F., Xu, B. and Yeh, W.W.G., 2019. Streamflow scenario tree reduction based on conditional Monte Carlo sampling and regularized optimization. *Journal of Hydrology*, 577, p.123943.

and retains the following copywrite:

Chapter 4 contains the following submitted journal article:

Li, J., Zhang, W., and Yeh, W.W.G., A proposed multi-objective, multi-stage stochastic programming with recourse model for reservoir management and operation, In revision, *Water Resources Research*, 2020.

and retains the following copywrite:

VITA

Education

Master of Science in Statistics at University of California, Los Angeles (2019)

Master of Science in Civil and Environmental Engineering at University of California, Los Angeles (2017)

Bachelor of Science in Water Conservancy and Hydropower Engineering, Hohai University, China (2016)

Journal Publications

[1] **Li, J.**, Zhu, F., Xu, B. and Yeh, W.W.G., 2019. Streamflow scenario tree reduction based on conditional Monte Carlo sampling and regularized optimization. Journal of Hydrology, 577, p.123943.

[2] **Li, J.**, Zhou, Q. and Yeh, W.W.G., 2020. A Bayesian hierarchical model for estimating the statistical parameters in a three-parameter log-normal distribution for monthly average streamflows. Journal of Hydrology, 591, p.125265.

[3] **Li, J.**, Zhang, W., and Yeh, W.W.G., A proposed multi-objective, multi-stage stochastic programming with recourse model for reservoir management and operation, In revision, Water Resources Research, 2020

[4] Zhang, W., **Li, J.**, Pan, L., Lei, X., Chen, J., and Yeh, W.W.G., When to start an adaptation strategy in response to climate change in reservoir system management, Submitted to Journal of Hydrology, 2020

Presentations

[1] AGU 2019 Fall meeting, oral presentation: A Bayesian hierarchical model for estimating the statistical parameters in a three-parameter log-normal distribution for monthly average streamflows, San Francisco, California, December 9-13

[2] AGU 2020 Fall meeting, poster presentation: A proposed multi-objective, multi-stage stochastic programming with recourse model for reservoir management and operation, Online Everywhere, December 1-17

Chapter 1

INTRODUCTION

Reservoir operation and management, typically referring to scheduling reservoir releases, is one of the most active areas of research in the field of water resources. A proper reservoir operation is always associated with the benefit of flood control, water supply, and hydropower generations. The traditional methods of reservoir operation are deterministic-based. These models assume that all the input factors inherited in the reservoir operation model are fixed and without uncertainty, such as reservoir inflow, reservoir capacity curve errors, and river flood routing errors. These deterministic methods can be easily implemented. However, those methods based on the deterministic input factors may produce sub-optimal operation strategy due to the accuracy of the predictions, which may incur a huge benefit loss. Hence, it is important to account the input uncertainty into the reservoir operation models.

Among all the uncertainty factors, the forecasted reservoir inflow (i.e., streamflow) is assumed as the principal uncertainty factor in reservoir operation (Zhu et al., 2017). This is because that the reservoir inflow is the primary input for reservoir operation models. Due to the limitation of the hydrology forecasting, the deterministic forecasted inflow is not always accurate, sometimes even far from the actual values (Mao et al., 2000; Tucci et al., 2003). Because of this disadvantage, stochastic programming models have been introduced and widely developed in reservoir operation. In such model, the future inflows are assumed with uncertainties, which makes the optimal operation of reservoirs as a risk-based decision-making problem (Xu et al., 2015).

The chance-constrained programming (CCP) model is a widely applied stochastic programming model in reservoir operation (Revelle et al., 1969; Houck, 1979; Yeh, 1985; Sreenivasan and Vedula, 1996), which is often used along with the linear decision rule. For a chance-constrained model, the constraint is in a probabilistic form, as shown in 1.1.

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ s.t. \quad & P[g(x,w) \geq 0] \geq \beta \end{aligned}$$

(1.1)

where $f(x)$ is a deterministic function; x is the decision variable (vector); $g(x,w)$ is a function with random variable w; and β is the reliability level. In a long-term reservoir operation model, $f(x)$ can be the energy output function, flood risk, assurance rate, etc. The decision variable x is usually the water release or the decision rule parameter. The monthly average reservoir inflow is often regarded as the random variable.

The concept of CCP model is traditional and not complicated. However, the quality of CCP model is highly subject to the accuracy of the random variable distribution. This is because that a well-estimated distribution indicates an accurate characterization of uncertainty of the CCP model, thus yields an optimal operation strategy by solving the model. As discussed, for a long-term reservoir operation model, the random variable in the CCP model is usually the monthly average streamflow of each month. Therefore, it is desired to develop a method to better estimate the distribution of the monthly average streamflow, so that the monthly release decision obtained from this model can be better optimized.

A general approach to obtain the distribution of the monthly average streamflow is via frequency analysis. Frequency analysis is a widely used statistical tool, which is implemented by

fitting an assumed probability distribution to the observed data of a random variable. In this dissertation, a new method for better fitting a distribution to monthly average streamflow is proposed. It is assumed that the monthly average streamflow follows a three-parameter log-normal distribution (LN3). A Bayesian hierarchical model (BHM) is then developed for estimating the statistical parameters in LN3. The three underlying statistical parameters in LN3 are shift, shape, and location. When estimating a parameter, such as the shape, of a given month the BHM utilizes historical observations not only from the month under consideration but also from all other months. This is different from traditional statistical parameter estimation methods that only use historical observations for the month under consideration. Using cross-validation with test data log-likelihood as the measure of performance, the results show that BHM outperforms traditional estimation methods. Also, the new method is especially suitable for fitting the streamflow distribution where historical observations are limited. This method is elaborated in Chapter 2.

Another popular stochastic programming model in reservoir operation is the two-stage stochastic programming with recourse model. In this model, people make sequential decisions, between which some of the uncertain parameters become gradually known (Xu et al., 2015; Gutjahr and Pichler, 2016). At the immediate stage (i.e., the 1st stage), the uncertain parameter w is unknown but a first decision (i.e., here-and-now decision) has to be made. Then after the realization of uncertain parameter w [i.e., the future stage or the 2nd stage, and $w = (q, J, L, h)$], a second decision (i.e., wait-and-see decision) is obtained by solving a deterministic optimization problem.

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & f(x) + \boldsymbol{E}[g(x, w)] \\ s.t. \quad & x \in X; w \in \Omega, \end{aligned} \tag{1.2}$$

3

the generic form of this model at the 1st stage is shown in 1.2, where x is the 1st stage decision variable; w is the uncertain parameter; X is the set of feasible solutions (i.e., solution space); Ω is the set of w (i.e., sample space); $\boldsymbol{E}[*]$ is the expectation operator, and $f(x)$ is a general function without uncertain parameter. $g(x,w)$ is the recourse function (also known as the second stage value function), which is the solution of the following second stage problem 1.3. For the purpose of demonstrating, it is assumed that the second stage optimization problem is a linear programming problem.

$$\begin{aligned}\underset{y}{\text{maximize}} \quad & q(w)^T y \\ s.t. \quad & J(w)x + L(w)y \leq h(w),\end{aligned} \qquad (1.3)$$

where y is the 2nd stage decision variable. The terms q, J, L, and h are the realization data of the uncertain parameter w.

The uncertain parameter w can be represented in many different ways, and the discretized scenario tree method is one of the popular ways in the literatures (Xu et al., 2015; Séguin et al., 2017; Li et al., 2019). In this method, w is represented as the discretized inflow scenario and Ω is the inflow scenario tree. An inflow scenario tree is usually generated from the historical inflow data by employing different methods, such as clustering, moment matching, and sampling. Since the generated scenario tree is used as the input to a stochastic programming with recourse model, the size of the tree directly impacts the dimensionality of the optimization model (Casey and Sen, 2005). Xu et al. (2015) showed that the CPU time of running a stochastic programming with recourse model drops more than 70%, if the scenario tree size is halved. Therefore, to improve the efficiency of the stochastic model, the number of scenarios must be reduced properly.

Compared with scenario tree generation, scenario tree reduction techniques have received less attention in the literature. One way for scenario tree reduction is to re-generate a scenario tree with a smaller tree size (Xu et al., 2015). However, the nodal values of the original full tree are altered under this method, which is not compatible with the definition of scenario tree reduction. Dupacová et al. (2003) proposed a method for selecting a scenario subset from the full tree, which is achieved by minimizing probability metric between the reduced tree's distribution and the full tree's distribution. Nevertheless, the pursuit of matching higher-order moments will be at the expense of matching the more important first-order and second-order moments in reservoir operation. Additionally, it fails to accommodate serial and spatial correlations that are critical in the operation of a cascade reservoir system. To improve these drawbacks, this dissertation proposes a new scenario tree reduction method in Chapter 3, based on a variant of the Monte Carlo sampling method and regularized optimization. The proposed method does not alter the nodal values of the full scenario tree, and it focuses on matching the first two moments and co-moments that are critical in reservoir operation.

The two-stage stochastic programming with recourse model can handle the inflow uncertainty in reservoir operation. However, it is not designed to solve the multi-objective problem, which limits its applicability. In fact, Gutjahr and Pichler (2016) pointed out in their survey paper that although stochastic optimization and multi-objective optimization are well established in the field of operations research, their interaction is less developed. In reservoir operation, decision makers often need to satisfy different reservoir operation requirements (e.g., hydropower generation, water supply and flood control), thus a multi-objective optimization problem is posed (Yeh and Becker, 1982).

Therefore, it is desired to integrate the multi-objective optimization into the two-stage stochastic programming with recourse model.

A traditional and widely used way to do that is via scalarization method. That is, assign a weight (i.e., preference coefficient) to each objective and optimize the weighted sum of all the objectives. However, in the scalarization method, the weight for each objective is typically not easy to determine and is often based on expert experience. An even more serious disadvantage of the scalarization method is that the preference coefficient for each objective is assumed to be fixed over the entire domain, which is usually not the case in the real world.

For instance, a high reservoir water level is usually beneficial to power generation but unfavorable for flood control. Suppose two conflicting objectives are maximizing hydropower output (economic objective, f1) and minimizing the reservoir water level (safety objective, f2). If the reservoir water level is low, then it is acceptable to increase the water level by 1m, in exchange of 100 kWh power output. In this case, the preference coefficient of objective f2 over f1 is $100/1 = 100$. On the contrary, if the reservoir water level is already high, then a high exchange rate is required (e.g., a 1m water level increase in exchange for 10000 kWh power output). The preference coefficient of objective f2 over f1 is now $10000/1 = 10000$. Therefore, it can be seen that the weight of each objective should changed with respect to the objective values. In economics, this phenomenon is known as the law of diminishing marginal rate of substitution (Hicks, 1939; Besada and Vázquez, 1999; Dittmer, 2005, White, 2015), which states that consumers are willing to give out less and less quantity of

one good in order to get one more additional unit of another good. This example will be re-visited in the Chapter 4.

To improve these drawbacks and provide optimal reservoir operation strategies under both inflow uncertainty and conflicting objectives, this dissertation proposes a multi-objective, multi-stage stochastic programming with recourse model for reservoir management and operation in Chapter 4. A single best compromise solution on the Pareto front is selected for the immediate stage and the model moves forward one stage and is re-optimized over a moving planning horizon of fixed duration. The selection is achieved by a proposed linear spline utility function allied with regression, which satisfies the law of diminishing marginal rate of substitution. The proposed method is demonstrated on a case study of the Three Gorges Reservoir (TGR) in China.

The following chapters are reprints of the published papers, which elaborate on the methods discussed in the introduction. Specifically, Chapter 2 has been published in its current form in Journal of Hydrology: Li, J., Zhou, Q. and Yeh, W.W.G., 2020. A Bayesian hierarchical model for estimating the statistical parameters in a three-parameter log-normal distribution for monthly average streamflows. Journal of Hydrology, 591, p.125265. Chapter 3 has been published in its current form in Journal of Hydrology: Li, J., Zhu, F., Xu, B. and Yeh, W.W.G., 2019. Streamflow scenario tree reduction based on conditional Monte Carlo sampling and regularized optimization. Journal of Hydrology, 577, p.123943. Chapter 4 is submitted and in revision with Water Resources Research in its current form.

Chapter 2

Research papers

A Bayesian hierarchical model for estimating the statistical parameters in a three-parameter log-normal distribution for monthly average streamflows

Jinshu Li[a,b], Qing Zhou[b], William W.-G. Yeh[a,*]

[a] Department of Civil and Environmental Engineering, University of California, Los Angeles, CA, USA
[b] Department of Statistics, University of California, Los Angeles, CA, USA

ARTICLE INFO

ABSTRACT

We develop a Bayesian hierarchical model (BHM) for estimating the statistical parameters for monthly average streamflows. We assume monthly average streamflow can be characterized by a three-parameter log-normal distribution (LN3). The three underlying statistical parameters are shift, shape, and location. When estimating a parameter, such as the shape, of a given month the BHM utilizes historical observations not only from the month under consideration but also from all other months. This is different from traditional statistical parameter estimation methods that only use historical observations for the month under consideration. We apply the proposed BHM for parameter estimation to eight watersheds in the United States, where historical unimpaired streamflows have been collected. We also carry out parameter estimation using traditional methods, such as the maximum likelihood estimation, the method of moments, and the L-moment method. Using cross-validation with test data log-likelihood as the measure of performance, the results show that BHM outperforms traditional estimation methods. In addition, we show that as available observation data decreases, the more the proposed method improves relative to traditional methods. Since BHM utilizes information contained in the entire data set, it is especially suited for parameter estimation where historical observations are limited. Furthermore, we conduct a comparative analysis between BHM and an autoregressive model to demonstrate the advantage of BHM.

1. Introduction

Frequency analysis is a useful statistical tool in hydrology. It is implemented by fitting an assumed probability distribution to the observed data of a random variable (Stedinger, 1980; Xiong et al., 2014). Frequency analysis has been used to estimate extreme hydrologic events, such as maximum streamflow, and it can be used for estimating average streamflow as well. In flood frequency analysis, maximum streamflow is considered a random variable, while frequency analysis for average streamflow treats the annual or monthly average streamflow as a random variable. The assumed probability distributions for the two different types of frequency analysis are generally not identical. The generalized extreme value distribution (GEV) (i.e. the generalized form of Weibull and Gumbel distributions) and the Log-Pearson Type 3 distribution (LP3) are standard choices for flood frequency analysis (Bobee, 1975; Rao, 1980; Hosking et al., 1985a, 1985b; Smith, 1987; obee and Ashkar, 1991; Rosbjerg and Madsen, 1995; Stedinger and Lu, 1995; Hosking and Wallis, 1996; Vogel and Wilson, 1996; Morrison and Smith, 2002; and Griffis and Stedinger, 2007; Singh, 2013). Compared

with flood frequency analysis, the choice of distribution for average streamflow is more flexible. Using the chi-square goodness-of-fit statistic, Markovic (1965) compared the fit of normal (N), log-normal (LN2), three-parameter log-normal (LN3), Gamma (GAM), and Pearson Type 3 (P3) distributions for the annual average streamflow. The results suggest that GAM, P3, LN2, and LN3 are all capable of fitting the distributions. Vogel and Wilson (1996) further tested the above distributions of annual average streamflow in the U.S. using an L-moment diagram. They concluded that annual average streamflows are well approximated by P3, LN2, or LN3 distributions.

Unlike annual average streamflows, literature dealing with the choice of distribution for monthly average streamflow is sparse, even though monthly average streamflow may be used more frequently than annual average streamflow for water resource management and reservoir operations (Yeh, 1985; Xu et al., 2015a; Xu et al., 2015b; Li et al., 2019). Since the time scale becomes shorter, the volatility of monthly average streamflow would be higher than annual average streamflow; therefore, a three-parameter distribution would be suitable for monthly average streamflow due to its greater flexibility. Thus, both
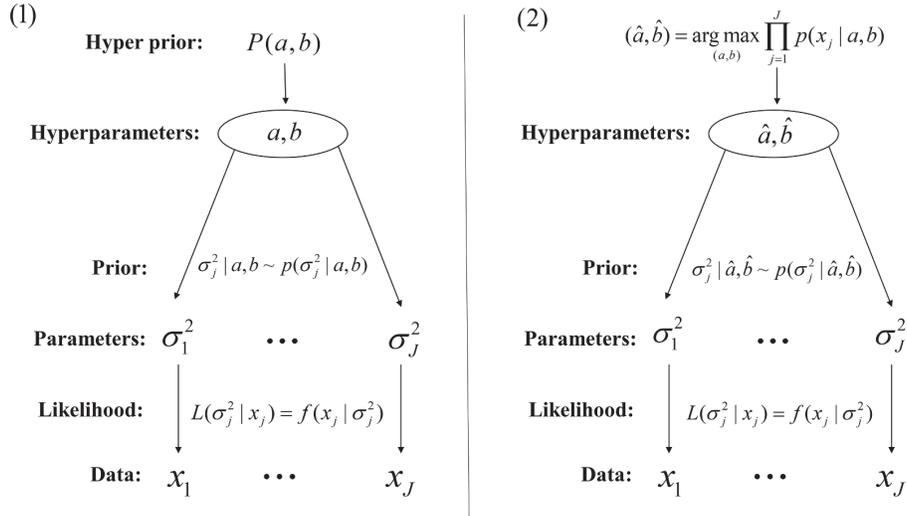
Fig. 1. The structure difference between a full BHM (1) and an empirical BHM (2).

P3 and LN3 would be good choices for monthly average streamflow. Furthermore, Sangal and Biswas (1970) suggested that LN3 can be applied successfully to the frequency analysis of both annual flows and monthly flows, and they also reported that P3 and LP3 will not be reliable when data are limited. Yue and Hashino (2007) demonstrated the ability of using LN3 to fit precipitation data. Bowers et al. (2012) also showed that river flow data are typically better fitted by a log-normal distribution than by a power law distribution. Therefore, in this study, we select three-parameter log normal distribution (LN3) to approximate monthly average streamflow. We also conduct the Shapiro-Wilk normality test to ensure this approximation is valid (Shapiro and Wilk, 1965; Royston 1995; Devineni et al., 2013).

Many traditional estimation methods have been applied to estimate LN3 parameters. Hill (1963) showed that the global maximum likelihood estimation (MLE) may approach infinity, which is inadmissible. Cohen (1951) proposed the local maximum likelihood estimation (LMLE) by equating partial derivatives of the log-likelihood function to zero. The studies of Harter and Moore (1966), Calitz (1973), Cohen and Whitten (1980), and Griffiths (1980) showed the validity of LMLE and suggested that it should be considered the prime method of estimating LN3 parameters. Method of moments estimation (MME) is a simple method that estimates parameters by equating the first, second, and third sample moments with the population values. However, this method may suffer from a large sampling error, introduced by the second and third sample moments (Cohen, 1988; Johnson et al., 1994). Moreover, the log-normal distribution may not be uniquely determined by its moments (Heyde, 1963). Inspired by the fact that the first order statistic in the sample exerts much greater influence on the shift parameter, Cohen and Whitten (1980) proposed the modified method of moments estimation (MMME), which is similar to MME but substitutes a function of the first order statistic. The L-moment method (L-MOM), first introduced by Hosking (1990), is also a popular way to estimate parameters. It matches the linear combinations of order statistics (i.e. L moments) for parameter estimation, which is analogous to MME but more robust to outliers. Combining the idea of MME and MLE, Griffiths (1980) introduced the zero-skewness method (Zeros). This method first computes the shift parameter by equating the sample skewness of log-transformed data to zero, then estimates the other two parameters (i.e. the shape and location parameters) by MLE. Thus, the estimation of the shift parameter and the other two parameters in this method are separate steps (this will be utilized and further discussed in the next section). Other popular methods for estimating LN3 parameters include

the median method (Sangal and Biswas, 1970), method of least squares (Snyder and Wallace, 1974), and method of entropy (Singh et al., 1986; Singh and Singh, 1987; Singh et al., 1990).

All of the above-mentioned traditional methods can be applied to estimate LN3 parameters of monthly average streamflow for a given month; however, they only use historical observations for the month under consideration. It is likely that the monthly average streamflow for different months are related, because of the time correlations of streamflow. In other words, some useful information contained in the other months (i.e. "between-group" information) is not utilized, since traditional methods only can use the "within-group" information. Therefore, the goal of this paper is to develop a new LN3 estimation method for monthly average streamflow that uses both "within-group" and "between-group" information.

Bayesian hierarchical models (BHMs) are a class of statistical models known for modeling "many statistical applications involving multiple parameters that can be regarded as related or connected in some way by the structure of the problem" (Gelman et al., 2013). This makes BHM a perfect model to capture the "between-group" information when estimating the parameters of monthly average streamflow for different months. Commonly used statistical models often face the following difficulty: With few parameters, the models cannot fit data accurately, whereas with a lot of parameters, the overfitting problem arises. However, as pointed out by Gelman et al. (2013): "BHMs have enough parameters to fit data well, while using a population distribution to structure some dependence into the parameters, thereby avoiding overfitting."

A typical full BHM model structure is shown in Fig. 1 (1). Consider a set of J groups of experiments. The data layer contains the data x_j (or statistics extracted from the data) for each group j, while the parameters layer includes the parameters that we want to estimate for all groups. These two layers usually are linked with an implied likelihood function. As a Bayesian method, prior distributions are needed for parameters (parameters are modeled as random variables in Bayesian inference), which reflect the belief ones have about the parameters. People often choose conjugate priors to facilitate derivations (i.e. the posterior and the prior distributions are in the same family of distributions). Assume the prior distributions are $\sigma_j^2 \stackrel{iid}{\sim} p(\sigma_j^2|a, b)$, in which there are two additional parameters (a, b). These parameters are called hyperparameters and they determine the shape of the priors. Thus, we further use a hyperprior to specify the prior distributions of the hyperparameters. Note that the variables of the hyperpriors are

hyperparameters [e.g. (a, b)], while the variables of the priors are the parameters that we want to estimate (e.g. σ_j^2). Therefore, the difference between prior and hyperprior is that they are prior distributions for model parameters (σ_j^2) and hyperparameters (a, b), respectively. In terms of the model structure, they are at different layers in the hierarchical models.

Additionally, BHM allows flexible inference approaches that come with many versions and variations, even though all the variations share the same principle of using several modeling layers. There are two main versions of BHM: 1) full Bayesian inference under the hierarchical model [i.e. full BHM, shown in Fig. 1 (1)], and 2) empirical Bayes method [i.e. empirical BHM, shown in Fig. 1 (2)]. The full BHM is the complete and rigorous model that contains data parameters, hyperparameters, and a hyperprior. With a hyperprior in a full BHM, we can first derive the posterior distribution of hyperparameters as $p(a, b|x)$, then use the MCMC method to draw from this posterior distribution, which accounts for the uncertainty introduced from the hyperparameters (a, b).

In contrast, the empirical BHM does not specify a hyperprior. Instead, it uses a point estimate for the hyperparameters (a, b) [e.g. Maximum Likelihood Estimator (MLE) in Eq. (1)]

$$(\hat{a}, \hat{b}) = \arg\max_{(a,b)} \prod_{j=1}^{J} p(x_j|a, b) \tag{1}$$

In other words, in an empirical BHM, the hyperparameter will be fixed after it is estimated, which simplifies the estimating process, although it loses some flexibility and uncertainties. Also, in some cases, it is not even feasible to estimate hyperparameters by MLE, since some optimization problems just don't have an optimum or are very difficult to solve.

Many BHM application studies in hydrology have been published: the BHM of multisite daily rainfall based on the binomial distribution and Gaussian process (Lima and Lall, 2009); the spatio-temporal BHM for extreme precipitation based on GEV and copula (Sang and Gelfand, 2009; Ghosh et al., 2011); the BHM in assessing streamflow response to a climate change based on GEV and generalized Pareto distribution (GPD) (Wei et al., 2012); the spatial BHM of extreme precipitation and extreme runoff based on GEV and GPD (Cooley et al., 2007; El Adlouni and Ouarda, 2009; Najafi and Moradkhani, 2013; Yan and Moradkhani, 2015; Steinschneider and Lall, 2015; Bracken et al., 2016); and the BHM of annual peak snow water equivalent (SWE), annual peak flow, and annual peak reservoir elevation based on GEV (Bracken et al., 2018). Renard (2011) estimated annual maximum rainfall in a regional context by constructing a BHM based on GEV and a regression model known as the "link function" between data and parameters. Devineni et al. (2013) directly applied a BHM on a linear regression model to reconstruct the average summer streamflow at five gauges in the Delaware River basin using eight regional tree-ring chronologies. Results of all the aforementioned studies show the advantages of BHM over traditional methods, since information contained in the data from different groups (mostly locations) are combined with the help of BHM.

However, most existing BHM studies in hydrology focus on extreme hydrological variables (i.e. maximum streamflow and rainfall). This is because frequency analysis of extreme events is more useful than average streamflow for designing flood control structures. For this purpose, the most well-known BHM framework is based on the generalized extreme value distribution (i.e. BHM-GEV), which is designed to model extreme random variables. (Cooley et al., 2007; EL Adlouni and Ouarda, 2009; Sang and Gelfand, 2009; Ghosh et al., 2011). However, we believe that normal events (e.g. the monthly-average streamflow) are also very important, especially for reservoir management and operation (Yeh, 1985; Xu et al., 2015a; Xu et al., 2015b; Li et al., 2019). For example, a better estimation of the monthly-average streamflow certainly will lead to better reservoir operations for the purpose of hydropower production or water supply.

Thus, in this paper we propose a new BHM [i.e. (BHM-LN3)] for estimating the statistical parameters for monthly average streamflows based on LN3 distribution. The proposed BHM also can be applied to any random variables that follow a LN3 distribution. Also, when constructing a full BHM, there can be two types of hyperpriors used: the noninformative hyperprior (i.e. "flat" hyperprior) and the informative hyperprior. A flat hyperprior is widely used when prior knowledge is not available for hyperparameters, and most of the BHM application studies in hydrology adopt this type of hyperprior (e.g. EL Adlouni and Ouarda (2009), Lima and Lall (2009), Renard (2011), Ghosh et al. (2011), Yan and Moradkhani (2015)). In contrast, the use of an informative hyperprior is much less frequent, and most informative hyperpriors found in the literature are empirical. For example, in building a BHM for estimating extreme precipitation, Cooley et al. (2007) assume a uniform [0.001, 0.02] distribution on a hyperparameter based on empirical information. In their BHM modeling of extreme precipitation, Bracken et al. (2016) use a weakly informative normal hyperprior centered at zero with an empirical standard deviation. Thus, in our paper, we also develop a new informative hyperprior that is expected to extract information from the data skewness. The proposed BHM under this informative hyperprior is compared with the frequently used flat hyperprior. We employ a 4-folds cross validation (CV) to compare the proposed BHM with traditional LN3 estimation methods. The test data log-likelihood is used as the criterion for model selection and comparison. The results show that our proposed BHM produces better parameter estimates than the selected traditional parameter estimation methods for all data sizes (long, medium, and short). The fewer the observation data, the more the proposed method improves over the traditional methods.

We outline this paper as follows: Section 2.1 introduces the data structure and the division of two seasons. Section 2.2 describes the zero-skewness method for estimating the shift parameter. Section 2.3 proposes the new BHM for estimating the shape parameter (variance BHM) and develops a new informative hyperprior. Section 2.4 discusses BHM for estimating the location parameter (mean BHM). Section 2.5 summarizes the estimated three-parameter log-normal distribution (LN3) based on BHM. Section 2.6 introduces log-likelihood as the goodness of fit metric, as well as illustrates the K-folds CV and the model selection criteria. Section 3 applies the proposed method to eight watersheds in the United States. Section 4 compares BHM with an autoregressive model and investigates the potential loss of uncertainty. We then provide final remarks and conclusions in Section 5.

2. Methodology

2.1. Data preparation

2.1.1. Data structure and assumed data distribution

The data used to conduct this research is the historical monthly average streamflow and it is structured as Fig. 2:

where N is the total number of historical data series; the monthly average streamflow x_j is modeled as a random variable that follows a f distribution (e.g. f can be a normal distribution , log-normal distribution, etc.); x_{ij} is a sample value of random variable x_j; and par_j is a set of parameters of the distribution of x_j [i.e. the distribution of x_j is $f(x_j|par_j)$].

In this study, we select a three-parameter log-normal distribution (LN3) for f, because 1) LN3 is a flexible distribution with a general skewness that can be used successfully for analyzing hydrologic data; and 2) LN3 is highly related to a normal distribution. By subtracting the shift parameter γ and then taking the log transformation, a LN3 variable can be transformed into a normal variable, which is the desired property for constructing BHM.

The probability density function (pdf) of the LN3 distribution is

Fig. 2. An example of historical data structure.

$$f(x|par) = \frac{1}{(x-\gamma)\sigma\sqrt{2\pi}} \exp\left\{-\frac{(\ln(x-\gamma)-\theta)^2}{2\sigma^2}\right\}, \tag{2}$$

where x is a random variable that follows the LN3 distribution [i.e. $f(x)$]; par is a set of parameters [i.e. $par = (\gamma, \sigma^2, \theta)$]; γ is the shift parameter, satisfying the constraint of $\gamma < x < \infty$; σ^2 is the shape parameter, with $\sigma > 0$; and θ is the location parameter. The goal is to estimate the distribution $f(x_j|par_j)$ for the monthly average streamflow random variable x_j for each month j ($j = 1, 2, ..., 12$).

2.1.2. Division of data into a dry season and wet season

The advantage of BHM is that it can make use of the between group's information. However, if information in the two groups are intrinsically "far away" from each other, then Bayesian analysis may not be able to improve the results. For example, the mean value of streamflow of July may be highly correlated with the mean value of streamflow of August, yet it will be largely uncorrelated with the mean value of streamflow of January. In many regions, streamflow in July may be much lower than that of January. Therefore, to maximize the advantage of BHM, we divide the year into two seasons: dry season and wet season. We will construct a hierarchical model for each of the two seasons. In our case study, we consider June to November as the dry season and December to May as the wet season. Note that the way to divide the data is not fixed, and this is not even required.

2.2. Zero-skewness method for estimating the shift parameter γ

The zero-skewness method (Griffiths, 1980; Royston, 1992) is a traditional method for estimating LN3 parameters. It is a combination of moment method estimation (MME) and maximum likelihood estimation (MLE). Note that "zero-skewness" refers to y (i.e. the transformed data) after we remove the skewness in x (i.e. the original streamflow data). And the only assumption made is that the original data x follows LN3. In this study, we adopt this zero-skewness method to estimate shift parameter γ as follows:

1) Since the original streamflow data (i.e. x) is assumed to follow LN3 distribution, we first carry out a log-transformation on $(x - \gamma)$. Note here the shift parameter γ is unknown and will be estimated. And the log-transformed result is y [i.e. $y = \log(x - \gamma)$]. Then, the transformed data y should follow a normal distribution.

2) Since the skewness for normal distribution (i.e. y) is zero, then

$$Skewness(y) = 0. \tag{3}$$

Using the estimator of the population skewness, Eq. (3) can be future expressed as

$$\frac{\frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y})^3}{\left[\frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y})^2\right]^{3/2}} = 0, \tag{4}$$

where $y_i = \log(x_i - \gamma)$ and $\bar{y}$ is the mean value of y. When estimating a normal distribution, the skewness estimator used in Eq. (4) is unbiased and consistent (Kendall and Stuart, 1969; Cramer, 1997; Joanes and Gill, 1998; Doane and Seward, 2011).

3) Since Eq. (4) has only one unknown (i.e. γ), solving Eq. (4) (Kowarik 0000; Millard, 2013), yields the γ estimate.

The reason we estimate γ by the zero-skewness method is because the estimation of γ is independent of the estimation of θ and σ^2 in this method. This independence provides an opportunity to improve the estimation of θ and σ^2 with the proposed BHM.

After estimating γ by the zero-skewness method, we will then construct our BHMs to estimate θ and σ^2. We first perform a log transformation on $x_{ij} - \gamma_j$ (i.e. the original samples following LN3 minus γ) to get y_{ij}, as Fig. 3 shows. We now refer to our data as y_{ij}. To ensure that y_j follows a normal distribution for each month j, we conduct the Shapiro-Wilk normality test (Shapiro and Wilk, 1965; Royston 1995; Devineni et al., 2013). If y_j passes the Shapiro-Wilk normality test, then y_j is normally distributed with parameters θ_j and σ_j^2, and x_j follows the three-parameter log-normal distribution.

2.3. Bayesian hierarchical model for estimating σ^2 (variance BHM)

2.3.1. Structure of variance BHM

A BHM is constructed for estimating the shape parameter σ^2, as

Fig. 3. Log-transformation on the original data.
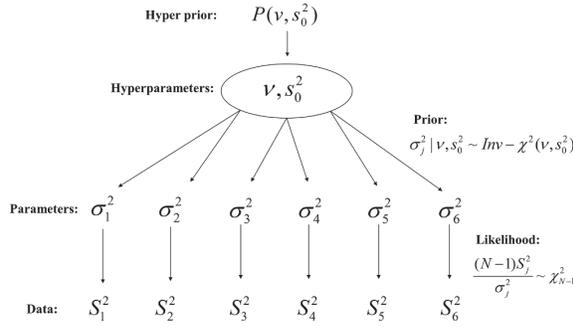
Fig. 4. Variance BHM for estimating σ^2.

shown in Fig. 4. After the log transformation, the data now are y_{ij}, where $i = 1, ..., N$ and $j = 1, ..., J = 6$ (6 months/groups in each season). Our goal is to estimate the parameter σ_j^2, which parameterizes the distribution of y_j:

$$y_j = \log(x_j - \gamma_j) \sim N(\theta_j, \sigma_j^2), \tag{5}$$

where σ_j^2 is the unknown shape parameter for month j; and θ_j is the unknown location parameter for month j.

A sufficient statistic is widely used in the Bayesian analysis to facilitate derivations (Fraser, 1963; Kholevo, 2001; Steel, 2007). In our proposed BHM model, a sufficient statistic for the parameter σ_j^2 is the sample variance S_j^2 for each group j:

$$S_j^2 = \frac{1}{N-1} \sum_{i=1}^{N} (y_{ij} - \bar{y}_j)^2. \tag{6}$$

Based on the distribution (Ross, 2006), we formulate

$$\frac{d \times S_j^2}{\sigma_j^2} \sim \chi_d^2, \tag{7}$$

where $d = N - 1$ (i.e. constant) and χ_d^2 is the chi-squared distribution with d (i.e. $N - 1$) degrees of freedom.

A conjugate prior is used for estimating σ_j^2 (Hoff, 2009):

$$\sigma_j^2 | v, s_0^2 \overset{iid}{\underset{\sim}{}} Inv - \chi^2(v, s_0^2), \quad j = 1, ..., J = 6, \tag{8}$$

where $Inv - \chi^2(v, s_0^2)$ is an inverse chi-squared distribution, parameterized by v and s_0^2 (hyperparameters). It is also the conjugate prior in Bayesian statistics when estimating a normal variance with unknown mean (Raiffa and Schlaifer, 1961; Gelman et al., 2013).

Based on Eqs. (7) and (8), the conditional posterior distribution of σ_j^2 given the hyperparameters (v, s_0^2) can be derived as (see Appendix A)

$$P(\sigma_j^2 | v, s_0^2, S_j^2) \sim Inv - \chi^2 \left(v + d, \frac{v s_0^2 + d S_j^2}{v + d} \right), \quad j = 1, ..., 6. \tag{9}$$

If we can draw a sufficient number of posterior hyperparameter samples [i.e. $(v, s_0^2 | S^2)$], then the posterior of σ_j^2 can be obtained by sampling from Eq. (9). Therefore, the posterior distribution of hyperparameters $P(v, s_0^2 | S^2)$ needs to be derived.

To perform a full Bayesian analysis, a hyperprior $P(v, s_0^2)$ is needed for $P(v, s_0^2 | S^2)$, where S^2 is the set of all S_j^2 (i.e. $S^2 = \{S_j^2 | j = 1, ..., 6\}$). For different seasons (i.e. dry and wet), we propose different hyperpriors. For now, we assume the hyperprior is $P(v, s_0^2)$. Thus, the posterior distribution of hyperparameters can be derived as (see Appendix B)

$$P(v, s_0^2 | S^2) \propto P(v, s_0^2) \times \prod_{j}^{J} \frac{1}{Beta(v/2, d/2)} \cdot \frac{(s_0^2 v/2)^{v/2}}{[(v s_0^2 + d S_j^2)/2]^{(v+d)/2}}, \tag{10}$$

where $Beta(x, y)$ is the beta function [i.e.

$Beta(x, y) = \Gamma(x)\Gamma(y)/\Gamma(x + y)]$ and $\Gamma(x)$ is the Gamma function. Note that the posterior distribution $P(v, s_0^2 | S^2)$ is conditioned on $S^2 = (S_1^2, ..., S_j^2)$ rather than a single S_j^2, since information from all the months/groups is being used to estimate hyperparameters (v, s_0^2). This is how BHM incorporates "between group" information into estimating a certain group's parameter σ_j^2 (Gelman et al., 2013).

Based on the distributions in Eqs. (9), (10) and the law of total expectation (Wolpert, 2010), we are now able to calculate the Bayesian estimator $\hat{\sigma}_j^{2B}$ (superscript "B" represents "Bayesian") by the posterior mean of σ_j^2:

$$\hat{\sigma}_j^{2B} = E[\sigma_j^2 | S^2] = E[E(\sigma_j^2 | v, s_0^2, S^2) | S^2] = E[\bar{\sigma}_j^2(v, s_0^2, S_j^2) | S^2], \tag{11}$$

where $\bar{\sigma}_j^2$ is the conditional posterior mean of σ_j^2, given (v, s_0^2), i.e.

$$\bar{\sigma}_j^2(v, s_0^2, S_j^2) = E(\sigma_j^2 | v, s_0^2, S^2) = \frac{v s_0^2 + d S_j^2}{v + d - 2} = \frac{v-2}{v+d-2} \cdot \left(\frac{v s_0^2}{v-2} \right) + \frac{d}{v+d-2} \cdot S_j^2$$
$$= B_{\sigma^2} \cdot \left(\frac{v s_0^2}{v-2} \right) + (1 - B_{\sigma^2}) \cdot S_j^2. \tag{12}$$

The second equality comes from the fact that the expected value [i.e. $E(\sigma_j^2 | v, s_0^2)$] of $Inv - \chi^2(v, s_0^2)$ (i.e. the inverse chi-squared distribution) is $v s_0^2 / (v - 2)$. The ratio $(v - 2)/(v + d - 2)$ is the shrinkage factor $B_{\sigma^2} \in (0, 1)$ for this variance BHM, which indicates the magnitude of shrinkage from MLE (i.e. S_j^2) to the prior expected value [i.e. $E(\sigma_j^2 | v, s_0^2)$]. Once we have simulated a large sample $(v_{(k)}, s_{0(k)}^2)$, $k = 1, ..., n$, we can approximate the expectation of Eq. (11) by Monte Carlo simulation.

We summarize the algorithm for estimating $\hat{\sigma}_j^{2B}$ in the following steps:

Step 1: Draw n (a large number) samples of $(v_{(k)}, s_{0(k)}^2)$ from the posterior distribution of hyperparameters $P(v, s_0^2 | S^2)$ [i.e. Eq. (10)], based on the Metropolis-Hastings algorithm, a MCMC method (Metropolis et al., 1953).

Step 2: For each $(v_{(k)}, s_{0(k)}^2)$, calculate the expected value using Eq. (12).

Step 3: Calculate the mean of $\bar{\sigma}_j^2$: $\hat{\sigma}_j^{2B} \approx \frac{1}{n} \sum_{k=1}^{n} \bar{\sigma}_j^2(v_{(k)}, s_{0(k)}^2, S_j^2)$.

According to the weak law of large numbers, when n is sufficiently large, $\hat{\sigma}_j^{2B}$ converges in probability to the Bayesian estimator $E[\sigma_j^2 | S^2]$ (derived from the proposed BHM model). A key advantage of our proposed BHM is that $\hat{\sigma}_j^{2B}$ is estimated not only using the "within-group" information, but also the "between-group" information. Thus, it is expected that the results from BHM will have an expected mean square error (MSE) that is less than or equal to the traditional MLE [i.e. $E(MSE(\hat{\sigma}_j^{2B})) \leqslant E(MSE(\hat{\sigma}_j^{2MLE}))$], due to its shrinkage (Efron and Morris, 1977; Chaloner, 1987; Berger, 2013).

2.3.2. Selection of hyperprior for variance BHM

BHM requires a hyperprior $P(v, s_0^2)$ to provide the prior knowledge of hyperparameters [see Eq. (10)]. Though how to select this hyperprior $P(v, s_0^2)$ is an interesting question, the choice of $P(v, s_0^2)$ has negligible impact on the results, since it merely represents the prior knowledge of the hyperparameters. In this study, we suggest two different hyperpriors for the variance of BHM. The first is the "non-informative hyperprior" (or "flat hyperprior") and the second is the "informative hyperprior".

A flat hyperprior is widely used when prior knowledge is not available for either hyperparameter (v, s_0^2) or parameter σ^2 itself. In other words, a flat hyperprior will lead to the same possibility for every σ^2 value, instead of focusing on any particular σ^2 value. It is expected to work well unless the number of groups J is low (Gelman, 2006). We derive our flat hyperprior as follows:

First we assume v and s_0^2 are independent in the prior:

$$P(v, s_0^2) = P(v)P(s_0^2). \tag{13}$$

Then we assume v is uniformly distributed on $(0, 15d)$ where $d = N - 1$. Thus we arrive at

$$P(v, s_0^2) \propto P(s_0^2), \quad v \in (0, 15d). \tag{14}$$

To make a flat hyperprior, we further assume $P(s_0) \propto 1$. By a change of variable, we can derive the flat hyperprior as

$$P(v, s_0^2) \propto \frac{1}{\sqrt{s_0^2}}, \quad v \in (0, 15d). \tag{15}$$

A fairly wide but finite support [i.e. $(0, 15d)$] for hyperparameter v is specified to ensure: 1) the posterior of the hyperparameters [i.e. Dis. (10)] is always proper (i.e. integrable to a finite value) based on the proposed flat hyperprior [i.e. Dis. (15)] (Gelman et al., 2013); and 2) the shrinkage factor B_{σ^2} is sufficiently large when v approaches its maximum. This will not limit the degree of shrinkage of the model. For example, when v approaches $15d$, $B_{\sigma^2} \in (0, 1)$ will be close to 15/16, which is sufficiently large for most applications.

Using a flat hyperprior is safe, but rather conservative. So, if some additional information on parameter σ^2 can be extracted from the data or by expertise, an informative hyperprior should be considered. However, very few BHM applications in hydrology have developed informative hyperpriors. In this study, we suggest a possible way of constructing an informative hyperprior by extracting prior information from the data skewness.

First, the skewness of LN3 is defined as $skew_j$, and it is a function of only one parameter σ_j^2:

$$skew_j = (e^{\sigma_j^2} + 2) \times \sqrt{e^{\sigma_j^2} - 1}. \tag{16}$$

Based on the idea of MME, the sample skewness of historical streamflow records b_j ($j = 1, ..., 6$) can be calculated by

$$b_j = \frac{N\sqrt{N-1}}{N-2} \cdot \frac{\sum_{i=1}^{N} (x_{ij} - \bar{x}_j)^3}{\left[\sum_{i=1}^{N} (x_{ij} - \bar{x}_j)^2\right]^{3/2}}, \tag{17}$$

where x_{ij} is the monthly averaged historical streamflow in year i for month j, $\bar{x}_j$ is the mean of historical streamflow records for month j, and N is total number of years of historical streamflow records. Eq. (17) is also known as "the adjusted Fisher–Pearson standardized moment coefficient", which is adopted by Excel, SAS, and SPSS as the default way to calculate skewness (Doane and Seward, 2011). Although it is a biased estimator, it provides an adjustment for sample size, which makes it closer to an unbiased estimator (Joanes and Gill, 1998; Doane and Seward, 2011). Thus, Eq. (17) is a good skewness estimator that can help alleviate the bias problem common to most skewness estimators, as pointed out by Wallis et al. (1974). Another potential issue for the estimators of skewness is the boundedness (Kirby, 1974); that is, the estimators may reach bound due to smaller sample size or hydroclimatic variability. Using the L-moment estimator would alleviate this concern.

Letting $skew_j = b_j$ (i.e. the exact moment matching with σ_j^2 unknown), we can obtain the prior estimate of σ_j^2 from skewness in the data, denoted as $\tilde{\sigma}_j^2$. Then we calculate the average of $\tilde{\sigma}_j^2(j = 1, ...,6)$:

$$Ave(\tilde{\sigma}^2) = \frac{1}{6} \sum_{j=1}^{6} \tilde{\sigma}_j^2. \tag{18}$$

Now we have $Ave(\tilde{\sigma}^2)$ as the prior knowledge of σ^2, and we proceed to extract the prior information of the hyperparameters (v, s_0^2) (i.e. determine hyperprior $P(v, s_0^2)$). We want our hyperprior to reflect the prior knowledge. Recall that our prior is an inverse chi-squared distribution:

$$\sigma_j^2 | v, s_0^2 \sim Inv - \chi^2(v, s_0^2). \tag{19}$$

The mode and expected value of this inverse chi-squared distribution are:

$$Mode = \frac{vs_0^2}{v + 2}, \tag{20}$$

$$Expected\ value = \frac{vs_0^2}{v - 2}, \quad v > 2. \tag{21}$$

Since we want every σ_j^2 sampled from Eq. (19) to be close to $Ave(\tilde{\sigma}^2)$, we set the mode or the expected value of (19) equal to $Ave(\tilde{\sigma}^2)$, which leads to

$$\frac{vs_0^2}{v \pm 2} = Ave(\tilde{\sigma}^2). \tag{22}$$

Reformulating Eq. (22), we arrive at

$$vs_0^2 - vAve(\tilde{\sigma}^2) \pm 2Ave(\tilde{\sigma}^2) = 0. \tag{23}$$

We expect samples from the hyperprior $P(v, s_0^2)$ have a high chance of satisfying Eq. (23). Our goal is then to find a distribution as our informative hyperprior. This distribution should have its highest probability density on the points (v, s_0^2) that satisfy Eq. (23). We suggest a simple informative hyperprior that can achieve this goal in the form of

$$P(v, s_0^2) \propto \frac{1}{(vs_0^2 - vAve(\tilde{\sigma}^2) \pm 2Ave(\tilde{\sigma}^2))^k}, \tag{24}$$

where k is a constant coefficient and $k > 0$. The choice of value for k, as well as the choice of a positive or negative sign, is not restricted.

In this process, the hyperprior parameter $\tilde{\sigma}_j^2$ in Eq. (24) is estimated only from the data skewness, not from any other data information (i.e. sample variance and mean) used in the BHM model. Therefore, using $\tilde{\sigma}_j^2$ as the prior information of σ_j^2 will not cause over-fitting (i.e. no reuse of the same data information). After that, we calculate $Ave(\tilde{\sigma}^2)$ (i.e. the average of six $\tilde{\sigma}_j^2$) and equate it to the mean/mode of the prior distribution, from which we derive Eq. (23). This step specifies the prior information of (v, s_0^2) from the skewness-based estimates $\tilde{\sigma}_j^2$. Based on that equation, we finally propose Eq. (24) as our informative hyperprior.

Note that the derived informative hyperprior is based on the assumption that all six months' prior information of σ_j^2 (i.e. $\tilde{\sigma}_j^2$) can be represented by $Ave(\tilde{\sigma}^2)$, which requires all six $\tilde{\sigma}_j^2$ to be concentrated on $Ave(\tilde{\sigma}^2)$ [i.e. $Var(\tilde{\sigma}_j^2)$ (the variance of $\tilde{\sigma}_j^2$) is small]. This requirement may only be satisfied for the months in the wet season, since the months in the dry season typically have high skewness, and thus large $\tilde{\sigma}_j^2$ values, which often may lead to large $Var(\tilde{\sigma}_j^2)$ values. Therefore, although the choice of using the "flat" or "informative" hyperprior is rather flexible, we only use the flat hyperprior for the months of the dry season. For the months in the wet season, we can either be conservative and use the flat hyperprior, or adopt the informative hyperprior as proposed. In sum, for each watershed, the following two scenarios of hyperprior will be compared: (1) Use the flat hyperprior for both the dry and wet season; (2) Use the flat hyperprior for the dry season and adopt the proposed informative hyperprior for the wet season.

2.4. Bayesian hierarchical model for estimating θ (mean BHM)

Another BHM (mean BHM) is used for estimating the location parameter θ, as shown in Fig. 5. This mean BHM originally was discussed in Gelman et al. (2013), where the authors assumed the true variances σ_j^2 are known. However, in this study, we first estimate $\hat{\sigma}_j^{2B}$ from section 2.3 and then build the mean BHM conditioning on $\hat{\sigma}_j^{2B}$.

We first represent the data y_{ij} by its sample mean $\bar{y}_j$ for each group j, which is also a sufficient statistic for θ_j:

$$\bar{y}_j = \frac{1}{N} \sum_{i=1}^{N} y_{ij}, \tag{25}$$

where N is the total number of historical records in each month.

Then the likelihood function can be derived from

$$\bar{y}_j | \theta_j, \sigma_j^2 \sim N(\theta_j, \sigma_j^{2*}), \tag{26}$$

where we define $\sigma_j^{2*} = \hat{\sigma}_j^{2B}/N_j$, and assume $N_1 = N_2, ..., N_j = N$ for
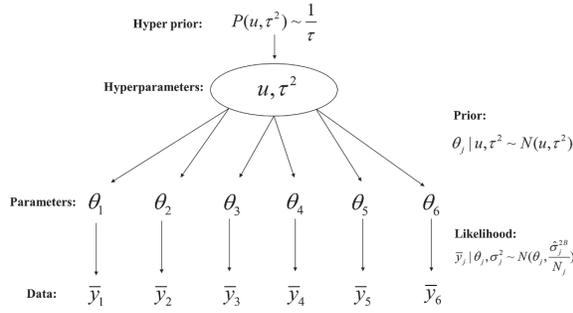
Fig. 5. Mean BHM for estimating θ.

simplicity.

We use a conjugate prior to provide the prior information for estimating θ_j:

$$\theta_j | u, \tau^2 \overset{iid}{\underset{\sim}{}} N(u, \tau^2), \quad j = 1, ..., J = 6, \tag{27}$$

where $N(u, \tau^2)$ is a normal distribution, parameterized by u and τ^2 (hyperparameters). It is also the conjugate prior in Bayesian Statistics when estimating the normal mean with known variance.

Based on Eqs. (26) and (27), the conditional posterior distribution of θ_j given hyperparameters (u, τ^2) can be formulated as

$$\theta_j | u, \tau^2, y, \sigma_j^{2*} \sim N(\hat{\theta}_j, V_j), \tag{28}$$

where

$$\hat{\theta}_j = \frac{\frac{1}{\sigma_j^{2*}} \bar{y}_j + \frac{1}{\tau^2} u}{\frac{1}{\sigma_j^{2*}} + \frac{1}{\tau^2}} = \frac{\frac{1}{\tau^2}}{\frac{1}{\sigma_j^{2*}} + \frac{1}{\tau^2}} \cdot u + \frac{\frac{1}{\sigma_j^{2*}}}{\frac{1}{\sigma_j^{2*}} + \frac{1}{\tau^2}} \cdot \bar{y}_j = B_\theta \cdot u + (1 - B_\theta) \cdot \bar{y}_j, \tag{29}$$

and

$$V_j = \frac{1}{\frac{1}{\sigma_j^{2*}} + \frac{1}{\tau^2}}. \tag{30}$$

The ratio $\frac{1}{\tau^2} / \left(\frac{1}{\sigma_j^{2*}} + \frac{1}{\tau^2} \right)$ is the shrinkage factor $B_\theta \in (0, 1)$ for this mean BHM, which indicates the magnitude of shrinkage from MLE (i.e. $\bar{y}_j$) to the prior expected value u [i.e. $E(\theta_j | u, \tau^2)$].

If sufficient posterior hyperparameter samples [i.e. $(u, \tau^2 | \bar{y})$] are available, then the posterior of θ_j can be obtained by sampling from Eq. (28). In such instances, the posterior distribution of hyperparameters $P(u, \tau^2 | \bar{y})$ needs to be derived.

We assume the hyperprior is $P(u, \tau^2)$. The posterior distribution of the hyperparameters [i.e. $P(u, \tau^2 | \bar{y})$] can be formulated as

$$P(u, \tau^2 | \bar{y}) \propto P(u, \tau^2) \times \prod_j^J N(\bar{y}_j | u, \sigma_j^{2*} + \tau^2), \tag{31}$$

where $N(\bar{y}_j | u, \sigma_j^{2*} + \tau^2)$ represents the probability density function (pdf) of the normal distribution of $\bar{y}_j$ with mean u and variance $\sigma_j^{2*} + \tau^2$, and $\bar{y}$ is the set of all $\bar{y}_j$ [i.e. $\bar{y} = (\bar{y}_1, ..., \bar{y}_6)$]. Also, for the mean BHM, we choose the flat hyperprior because the location parameter θ, unlike the shape parameter σ^2, often is dispersed widely for different months. Gelman et al. (2013) derived a flat hyperprior in the mean BHM as

$$P(u, \tau^2) \propto \frac{1}{\tau}. \tag{32}$$

To improve sampling stability, we propagate $P(u, \tau^2 | \bar{y})$ in factored form based on Bayes' rule:

$$P(u, \tau^2 | \bar{y}) = P(\tau^2 | \bar{y}) \cdot P(u | \tau^2, \bar{y}). \tag{33}$$

Compute the posterior distribution of τ^2 [i.e. $P(\tau^2 | \bar{y})$] as

$$P(\tau^2 | \bar{y}) = \int P(u, \tau^2 | \bar{y}) du$$
$$\propto \frac{1}{\tau} \cdot V_u^{\frac{1}{2}} \cdot \prod_{j=1}^{J=6} (\sigma_j^{2*} + \tau^2)^{-\frac{1}{2}} \cdot \exp \left[-\frac{(\bar{y}_j - \hat{u})^2}{2(\sigma_j^{2*} + \tau^2)} \right], \tag{34}$$

where

$$\hat{u} = \frac{\sum_{j=1}^{J} \frac{1}{\sigma_j^{2*} + \tau^2} \bar{y}_j}{\sum_{j=1}^{J} \frac{1}{\sigma_j^{2*} + \tau^2}}, \tag{35}$$

and

$$V_u = \left(\sum_{j=1}^{J} \frac{1}{\sigma_j^{2*} + \tau^2} \right)^{-1}. \tag{36}$$

Then the posterior distribution of u given τ^2 [i.e. $P(u | \tau^2, \bar{y})$] also can be calculated as

$$P(u | \tau^2, \bar{y}) \propto P(u, \tau^2 | \bar{y}) \propto \prod_{j=1}^{J} N(\bar{y}_j | u, \sigma_j^{2*} + \tau^2) \propto N(\hat{u}, V_u), \tag{37}$$

where $N(\hat{u}, V_u)$ represents the pdf of the normal distribution with mean $\hat{u}$ and variance V_u; $\hat{u}$ and V_u are defined in Eqs. (35) and (36).

With Eqs. (34) and (37), we are able to calculate the Bayesian estimator $\hat{\theta}_j^B$ (superscript "B" represents "Bayesian") under the l_2 loss function:

$$\hat{\theta}_j^B = E[\theta_j | \bar{y}] = E[E[\theta_j | u, \tau^2, \bar{y}] | \bar{y}] = E[\hat{\theta}_j(u, \tau^2, \bar{y}) | \bar{y}], \tag{38}$$

where $\hat{\theta}_j$ [i.e. the conditional posterior mean of θ_j given (u, τ^2)] is given in Eq. (29).

We summarize the above procedure of estimating $\hat{\theta}_j^B$ with the following algorithm:

Step 1: Draw n (a large number) samples of $\tau_{(k)}^2$ [i.e. $(k) = 1, ..., n$] from the posterior distribution of τ^2 [i.e. $P(\tau^2 | \bar{y})$], based on the Metropolis-Hastings algorithm.

Step 2: For each $\tau_{(k)}^2$, sample $u_{(k)}$ based on the posterior distribution of u given τ^2 [i.e. $P(u | \tau_{(k)}^2, \bar{y})$].

Step 3: For each pair of $(u_{(k)}, \tau_{(k)}^2)$, compute $\hat{\theta}_j(u_{(k)}, \tau_{(k)}^2, \sigma_j^{2*}, \bar{y}_j)$.

Step 4: Calculate the mean of $\hat{\theta}_j$: $\hat{\theta}_j^B = \frac{1}{n} \sum_{k=1}^{n} \hat{\theta}_j(u_{(k)}, \tau_{(k)}^2, \sigma_j^{2*}, \bar{y}_j)$.

According to the weak law of large numbers, when n is large enough, $\hat{\theta}_j^B$ converges to the Bayesian estimator $E[\theta_j | \bar{y}]$. This is a clear advantage of BHM over traditional estimators (such as MLE).

2.5. Estimated three-parameter log-normal distribution

After obtaining $\hat{\theta}_j^B$ and $\hat{\sigma}_j^{2B}$ from BHM, the results of our proposed estimating method based on BHM can be summarized as

$$y_j \sim N(\hat{\theta}_j^B, \hat{\sigma}_j^{2B}). \tag{39}$$

And its equivalent form:

$$x_j \sim LN3(\gamma_j, \hat{\theta}_j^B, \hat{\sigma}_j^{2B}). \tag{40}$$

With the pdf:

$$f(x_j) = \frac{1}{(x_j - \gamma_j) \hat{\theta}_j^B \sqrt{2\pi}} \exp \left\{ -\frac{(\ln(x_j - \gamma_j) - \hat{\theta}_j^B)^2}{2\hat{\sigma}_j^{2B}} \right\}. \tag{41}$$

2.6. Log-likelihoods and K-folds cross validation

In this study, we choose log-likelihoods as the measure of performance (i.e. goodness of fit metrics), because we want to evaluate the

difference between two distributions (i.e. the estimated LN3 and the true test data distribution). Log-likelihoods is one of the most widely used and well-acknowledged metrics for the goodness-of-fit of different statistical models (Fisher, 1992; Edwards, 1992; Berger and Wolpert, 1988). It is especially useful for evaluating the performance of the estimated distributions (Myung, 2003). When new observation data is collected, its log-likelihood value describes how well this distribution fits this new data (or how likely this new observation data is drawn from this distribution). Thus, the distribution with the highest log-likelihood value can be regarded as the best distribution to explain the observation data, and it is then the best one to characterize the variable (monthly streamflow in this study). This is a consequence of the fact that the Kullback-Leibler divergence (i.e. KL divergence) is minimized when an estimated distribution is identical to the true distribution (Kullback and Leibler, 1951; Kullback, 1997).

Most of goodness-of-fit statistics are essentially log-likelihood or built upon log-likelihood. The well-known AIC (Akaike information criterion; Akaike, 1998) and BIC (Bayesian information criterion; Schwarz, 1978) also are formulated as the sum of log-likelihood and minus penalties for complexity. Note that they are calculated with training data (not test data) and therefore use the model complexity penalty to approximate test data log-likelihood. In our case, we have both the training and the test dataset, so we believe using the test data log-likelihood as the metric is the most interpretable and standard way to evaluate the distribution performance among different methods.

K-folds cross-validation (K-folds CV) is a widely used technique for assessing statistical model performance (Kohavi, 1995; Refaeilzadeh et al., 2009). Compared with the traditional hold-out method (i.e. simply splitting the observation dataset into a training and a test set), K-folds CV is less affected by how the observation dataset is split, since every observation data point gets to be used as the test data exactly once (McLachlan et al., 2005). Thus, in this study, we employ a 4-folds CV and the steps are summarized as follows:

Step 1. Randomly divide the entire N years of observations into four folds (i.e. each fold contains $N/4$ years of observations).

Step 2. For the first test (i.e. $t = 1$), combine folds 1, 2, and 3 as the training dataset (i.e. the dataset that is used to estimate $\hat{\theta}_j^B$ and $\hat{\sigma}_j^{2B}$), while fold 4 is used as the test dataset in the test process. Since the log-likelihood is used as the metric in the test process, for an estimated distribution $\hat{f}_j$, the log-likelihood based on the test dataset is computed as

$$L_j^{t=1} = \log\left(\prod_{i \in V^t} \hat{f}_j(x_{ij})\right) = \sum_{i \in V^t} \hat{f}_j(x_{ij}),$$

(42)

where $V^t = \{i | i \in t^{th}\ test\ dataset\}$. Repeat the above procedure for $t = 2$, $t = 3$, and $t = 4$, where the corresponding test sets are fold 3, fold 2, and fold 1, as Fig. 6 shows. Calculate the test data log-likelihood $L_j^{t=2}$, $L_j^{t=3}$, and $L_j^{t=4}$.

Step 3. Compute the mean of L_j^t ($t = 1,2,3,4$) as L_j; it is the final test data log-likelihood for the estimated distribution $\hat{f}_j$.

Step 4. Compare the performance of different estimated distributions $\hat{f}_j$, $\hat{g}_j$, ..., and $\hat{h}_j$ ($j \in 1, ...,12$), by comparing their $L_j^{f_j}$, $L_j^{g_j}$, and $L_j^{h_j}$. The estimated distribution with the largest L_j is considered the best for month j.

Step 5. Compare different estimating methods f, g, ..., and h, by comparing their cumulative test data log-likelihood over 12 months (i.e. $\sum_{j=1}^{12} L_j^{f_j}$, $\sum_{j=1}^{12} L_j^{g_j}$, and $\sum_{j=1}^{12} L_j^{h_j}$). The estimating method with the largest $\sum_{j=1}^{12} L_j$ is considered the best.

The magnitude of $\sum_{j=1}^{12} L_j$ is related to the number of observation years (i.e. number of data N). To avoid the influence of the number of data N, when comparing two different estimating methods f and g, we calculate the "relative improvement" (*RI*) using Eq. (43):

Folds

	$t=1$	$t=2$	$t=3$	$t=4$
1	Train	Train	Train	Test
2	Train	Train	Test	Train
3	Train	Test	Train	Train
4	Test	Train	Train	Train
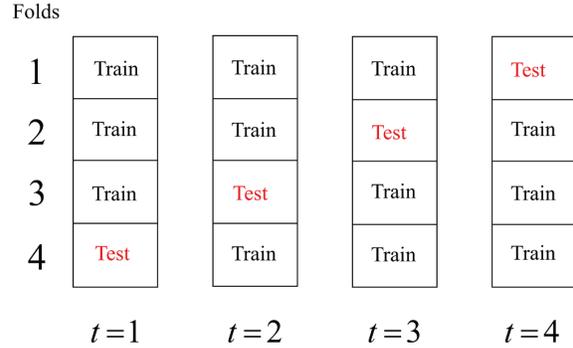
Fig. 6. The 4-folds cross-validation illustrative diagram.

$$RI = \frac{\sum_{j=1}^{12} L_j^{f_j} - \sum_{j=1}^{12} L_j^{g_j}}{\left|\sum_{j=1}^{12} L_j^{f_j}\right|} \times 100\%,$$

(43)

where f is selected as the base method. If *RI* is positive, then f is *RI*% improvement over g.

3. Case study

3.1. Data information summary

We select eight watersheds in the United States for our case study. Key information related to the selected sites is summarized in Table 1. Fig. 7 shows the location map. We collect the last 100 years (the information may vary slightly due to availability) of historical unimpaired monthly streamflow data from each site. All data and the site information are collected from the USGS official website (https://waterdata.usgs.gov/nwis/uv/?referred_module = sw).

3.2. Shapiro-Wilk normality test on the transformed data

After estimating the shift parameter γ_j by the zero-skewness method, we transform x_{ij} into y_{ij} by subtracting γ_j and taking the log. If y_j follows a normal distribution, then our assumption that x_j follows a three-parameter, log-normal distribution is valid. Therefore, we employ the Shapiro-Wilk normality test (Shapiro and Wilk, 1965) to determine whether y_j follows a normal distribution: First, from the 100 years of data, we randomly sample a dataset of 75 years of observations. Then, with a significance level $\alpha = 0.05$, we calculate the test statistic and p-values for each month and each watershed. If the p-value is less than 0.05, then we reject the null hypothesis that y_j follows a normal distribution. Table 2 records the data's p-values for each month and watershed.

We see from Table 2 that in only 6 out of 96 months (12 months per watershed) y_j may not have followed the normal distribution, which is insignificant. Therefore, we assume that the original data, , follows the three-parameter, log-normal distribution.

3.3. Example of CV results of Schoharie watershed

Using the 100-year dataset and the flat hyperpriors for both seasons, we plot the log-likelihood (L_j) results from the 4-folds CV for each month in Fig. 8. We use the Schoharie watershed as an example (other watersheds yield similar results). We compare the results from BHM with results obtained from the following five selected traditional parameter estimation methods: local maximum likelihood estimation (LMLE), L-moment (L-MOM), method of moment estimation (MME), modified method of moment estimation (MMME), and zero-skewness (Zeros). In Fig. 8, we choose the BHM results as the baseline, and plot

Table 1
Site information for the selected eight watersheds.

Site ID	Site Name	Drainage area (km²)	Latitude	Longitude	State
03,069,500	Cheat River (i.e. Cheat)	1856.853	39.122884	−79.681174	WV
01,543,000	Driftwood Creek (i.e. Driftwood)	705.4974	41.413396	−78.196952	PA
03,488,000	Holston River (i.e. Holston)	578.206	36.896781	−81.746229	VA
01,548,500	Pine Creek (i.e. Pine)	1557.047	41.521736	−77.44748	PA
01,350,000	Schoharie Creek (i.e. Schoharie)	612.5148	42.319528	−74.436537	NY
01,532,000	Towanda Creek (i.e. Towanda)	553.941	41.70702	−76.484665	PA
11,266,500	Yosemite Merced River (i.e. Yosemite)	833.0817	37.716871	−119.666279	CA
08,190,000	Nueces River (i.e. Nueces)	1961.433	29.428567	−99.997287	TX

the difference of log-likelihood between BHM and the other traditional methods (i.e. $L_j^B - L_j^f$, where B represents BHM; f represents other traditional methods). In other words, for a traditional method, if the difference for a given month is greater than zero, then BHM provides better estimates.

From Fig. 8, we see that BHM outperforms all five traditional methods, since the log-likelihood differences are positive in almost all months. The five traditional methods behave irregularly in different months, including the obvious outliers provided by the MME in the dry season. The results also show that BHM is more stable. For instance, the LMLE method behaves similarly as BHM in December, but it is clearly worse in January. In other words, the results from BHM can be regarded as the flat horizontal line of $y = 0$, which is not only below the results from other methods, but also exhibits the lowest volatility (i.e. it can be trusted in all months).

Kernel density estimation is a non-parametric data smoothing technique that estimates the probability density function of a random variable based on a finite data sample (Rosenblatt, 1956; Parzen, 1962). In Fig. 9, we compare the LN3 density estimations from BHM and several representative methods [i.e. Red: BHM; Cyan: MME; Purple (outermost): Zeros-skewness] against the real test data kernel density estimation [i.e. Green (innermost): real test data]. We see in Fig. 8 that, in January, the log-likelihood difference for MME is 2, while the log-likelihood difference for Zeros is around 1. In Fig. 9, we see that the LN3 distribution estimated from BHM best fits the real data kernel density. As the figure shows, the BHM estimation captures the peak most accurately, followed by the Zeros estimation, while the MME estimation clearly deviates from the correct peak position. Furthermore, both the left and right tails of the real data distribution are best matched by the BHM estimation, shown from the green area on both sides. Therefore, Fig. 9 enables us to easily interpret the log-likelihood difference (i.e. Fig. 8, Table 5–7). It is also a demonstration of the

advantage of our BHM over the selected traditional methods.

In addition, we also calculate the moment information (i.e. mean and variance) for each LN3 estimated by the BHM and Zeros methods. The formulas used are shown in Eqs. (44) and (45) and the moment values are shown in Tables 3.

$$Mean = e^{(\theta + \frac{\sigma^2}{2})} + \gamma \tag{44}$$

$$Varaince = (e^{\sigma^2} - 1)e^{(2\theta + \sigma^2)} \tag{45}$$

For most months, the variance differs significantly, and the mean is also distinguishable. Also, these results are just the mean (i.e. first order moment) and variance (i.e. second order moment) calculated as an example, not to mention the larger difference in the higher order moments such as skewness and kurtosis. Therefore, the differences between the two distributions in Fig. 9 [i.e. red (BHM) and purple (Zeros)] are significant.

Table 4 shows the estimated parameter values (i.e. $\hat{\theta}_j^B$ and $\hat{\sigma}_j^{2B}$) from the 4th test based on the proposed BHM and zero-skewness method (Zeros). In addition to the point estimates, we use the equal-tail method to calculate the 95% Bayesian interval estimation for each parameter. We see that the difference in $\hat{\sigma}_j^2$ is larger than the difference in $\hat{\theta}_j$. This indicates that the proposed BHM provides more shrinkage on the shape parameter σ_j^2 than the location parameter θ_j. We further investigate this by calculating the shrinkage factors (i.e. B_{σ^2} and B_θ) for BHM, and the result shows that the average B_{σ^2} is 0.29, while the average B_θ is only 0.02. On one hand, due to the relatively large average B_{σ^2} value, the result clearly explains why the shape parameter σ_j^2 can be estimated better by BHM. In addition, it is evident that BHM allows one to determine whether there is useful "between group" information to be extracted. If this is not the case, such as with the estimation of $\hat{\theta}_j$ in this example, BHM automatically assigns a small shrinkage factor B_θ that barely shrinks the MLE results.
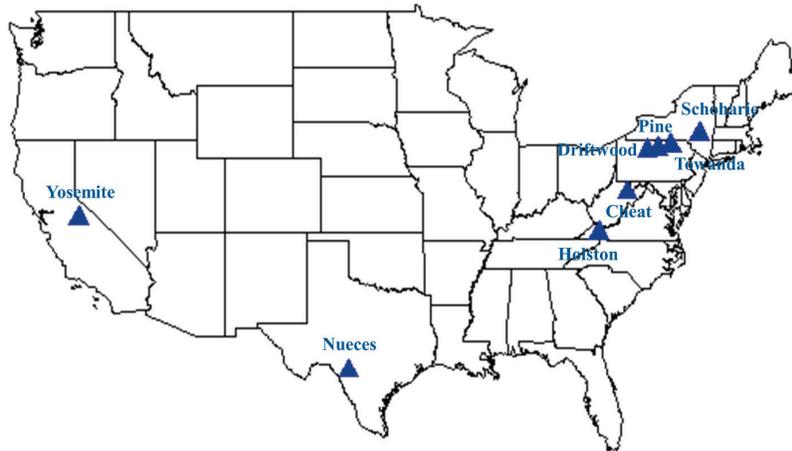
Fig. 7. The location map of eight gauging stations.

Table 2
P-values of the Shapiro-Wilk normality test.

Month	Cheat	Driftwood	Holston	Pine	Schoharie	Towanda	Yosemite	Nueces
Jun	0.654	0.882	0.199	0.941	0.696	0.662	0.169	0.034
Jul	0.366	0.441	0.979	0.514	0.787	0.340	0.417	0.841
Aug	0.621	0.991	0.138	0.507	0.132	0.681	0.998	0.423
Sep	0.974	0.019	0.237	0.999	0.568	0.367	0.832	0.839
Oct	0.380	0.310	0.707	0.114	0.019	0.145	0.066	0.962
Nov	0.788	0.083	0.017	0.181	0.294	0.522	0.351	0.851
Dec	0.472	0.382	0.002	0.254	0.590	0.281	0.895	0.987
Jun	0.309	0.746	0.607	0.344	0.436	0.580	0.962	0.968
Feb	0.555	0.293	0.763	0.745	0.401	0.278	0.951	0.015
Mar	0.671	0.911	0.960	0.384	0.671	0.777	0.941	0.547
Apr	0.149	0.996	0.838	0.672	0.966	0.474	0.652	0.125
May	0.397	0.763	0.372	0.315	0.566	0.873	0.729	0.542
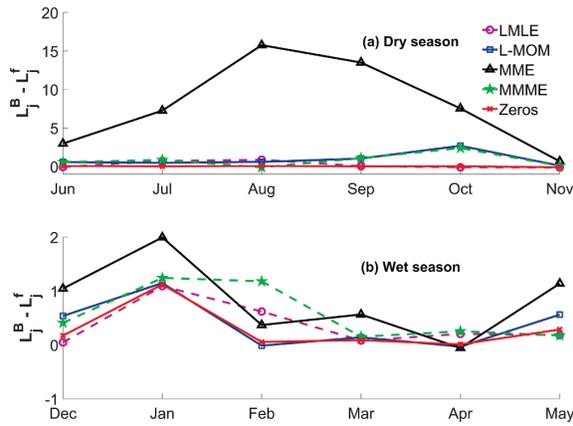
Fig. 8. Log-likelihood difference ($L_j^B - L_j^f$) for different traditional methods for (a) dry season months (b) wet season months.
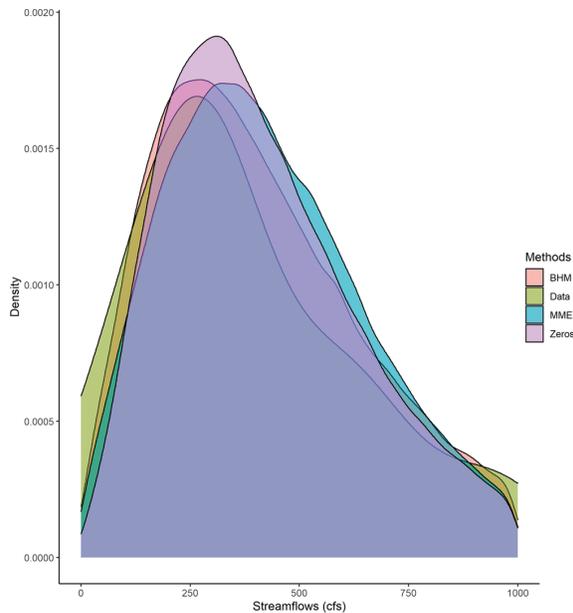
Fig. 9. A density plot, using the January results from Fig. 8.

Table 3
The mean and variance of two LN3 estimated by BHM and Zeros in two seasons.

	Mean (BHM)	Mean (Zeros)	Variance (BHM)	Variance (Zeros)
Dry				
Jun	338.27	336.62	147,346	141,308
Jul	176.14	174.44	77,951	75,760
Aug	115.47	113.45	46,158	44,559
Sep	222.94	221.3	354,222	355,303
Oct	326.26	326.3	309,827	307,578
Nov	529.42	519.48	174,266	143,686
Unit	(cfs)	(cfs)	(cfs)^2	(cfs)^2
Wet				
Dec	500.31	507.92	134,158	155,470
Jan	472.7	453.25	118,641	83,118
Feb	431.94	441.63	94,952	121,204
Mar	853.1	855.06	250,195	244,861
Apr	1098.28	1087.91	420,213	338,423
May	578.29	576.35	124,008	123,811
Unit	(cfs)	(cfs)	(cfs)^2	(cfs)^2

It is also evident that the shrinkage factor B_{σ^2} of the wet season (0.554) is much higher than B_{σ^2} of the dry season (0.026). This clearly shows that the model achieves substantial shrinkage for variance estimation in the wet season, also reflected by the observation that the Bayesian estimates across the six months in the wet season (Dec.-May) were quite close to each other. The variances in the dry season and the mean estimates exhibit almost no shrinkage, so they were very close to the MLEs, confirming the notion that BHMs can achieve data-driven shrinkage automatically.

In sum, the proposed BHM method rarely performs worse than the traditional methods based on MLE. This is due to the fact that if abundant "between group" information is detected, BHM will take full advantage and shrink the MLE results (e.g. the $\hat{\sigma}_j^2$ in Table 4).

3.4. CV results of tests with different dataset sizes

To further evaluate the performance of BHM under different dataset sizes, we choose two different datasets: the entire 100 years of observations, a large size that represents a case where the data are sufficient; and the most recent 60 years of observations, a medium size that is available for most hydrological stations. We use the flat hyperprior for both seasons, and 4-folds CV. The cumulative log-likelihood and relative improvement [Eq. (43) with BHM as the base model] results for each method are shown in Table 5.

Table 5 shows, for both the 100-year and 60-year tests, that BHM outperforms all other traditional methods in all eight watersheds (except a similar performance of BHM and LMLE in the 100-year test). This is evident because the cumulative log-likelihood of BHM is greater than the traditional methods, and the relative improvements (*RI*) over the traditional methods are all positive. Additionally, comparing the 60-

Table 4
The estimated parameters based on two different methods and shrinkage factors.

Season	Month	$\hat{\theta}_j^B$ (BHM)	95% S	95% E	B_θ	$\hat{\sigma}_j^{2B}$ (BHM)	95% S	95% E	B_{σ^2}	$\hat{\theta}_j$ (Zeros)	$\hat{\sigma}_j^2$ (Zeros)
Dry	Jun	5.3767	5.3678	5.3833	0.0096	0.8517	0.8336	0.8724	0.026	5.3808	0.8336
	Jul	4.4867	4.4794	4.5012	0.0146	1.3036	1.2712	1.3285		4.4797	1.2975
	Aug	3.8587	3.8408	3.8881	0.0181	1.6213	1.5751	1.6547		3.8384	1.6237
	Sep	4.3174	4.2993	4.3442	0.0236	2.1344	2.0636	2.1840		4.3018	2.1504
	Oct	5.1505	5.1390	5.1604	0.0148	1.3257	1.2924	1.3511		5.1534	1.3202
	Nov	6.5501	6.5440	6.5537	0.0028	0.2458	0.2225	0.2829		6.5546	0.2116
Wet	Dec	6.1652	6.1577	6.1763	0.0291	0.3491	0.3133	0.3843	0.554	6.1600	0.3862
	Jan	6.2938	6.2883	6.3002	0.0227	0.2695	0.2297	0.3112		6.2928	0.2078
	Feb	5.9397	5.9277	5.9596	0.0310	0.3736	0.3288	0.4230		5.9269	0.4411
	Mar	6.5577	6.5465	6.5648	0.0263	0.3136	0.2872	0.3435		6.5635	0.3066
	Apr	6.8921	6.8723	6.9040	0.0238	0.2827	0.2506	0.3202		6.9056	0.2375
	May	6.1908	6.1842	6.2004	0.0268	0.3204	0.2929	0.3494		6.1867	0.3218

Note: 95% S represents the start of 95% Bayesian interval, while 95% E is the end of 95% interval.

Table 5
CV results of tests with different dataset sizes.

100-year dataset

	LMLE	L-MOM	MME	MMME	Zeros	BHM
Cumulative log-likelihood						
Cheat	−2539.7	−2536.2	−2542.0	−2537.9	−2535.3	−2534.2
Driftwood	−2104.9	−2111.3	−2158.5	−2111.5	−2105.1	−2104.9
Holston	−1798.3	−1805.9	−1830.4	−1802.3	−1799.6	−1798.6
Pine	−2185.5	−2192.1	−2224.9	−2190.5	−2184.7	−2184.1
Schoharie	−2282.8	−2287.1	−2332.0	−2287.5	−2281.1	−2279.3
Towanda	−1963.6	−1972.8	−2022.5	−1964.2	−1959.7	−1959.5
Yosemite	−1977.4	−1979.2	−2019.8	−1979.0	−1976.9	−1976.8
Nueces	−1659.6	−1679.2	−1756.4	−1665.3	−1661.7	−1660.9
Relative improvement						
Cheat	0.22	0.08	0.31	0.14	0.04	0
Driftwood	0.00	0.30	2.55	0.31	0.01	0
Holston	−0.02	0.40	1.77	0.20	0.06	0
Pine	0.06	0.37	1.87	0.29	0.03	0
Schoharie	0.15	0.34	2.31	0.36	0.08	0
Towanda	0.21	0.68	3.21	0.24	0.01	0
Yosemite	0.03	0.12	2.13	0.11	0.00	0
Nueces	−0.08	1.09	5.44	0.26	0.05	0

60-year dataset

	LMLE	L-MOM	MME	MMME	Zeros	BHM
Cumulative log-likelihood						
Cheat	−1465.7	−1466.4	−1474.9	−1466.0	−1465.5	−1464.3
Driftwood	−1223.2	−1229.8	−1248.7	−1224.9	−1223.3	−1222.9
Holston	−1136.9	−1137.2	−1153.8	−1137.1	−1136.2	−1134.5
Pine	−1343.3	−1345.1	−1369.2	−1342.1	−1340.5	−1340.0
Schoharie	−1247.3	−1252.9	−1276.0	−1250.0	−1246.4	−1245.5
Towanda	−1152.4	−1161.9	−1183.5	−1150.5	−1148.5	−1148.0
Yosemite	−1217.2	−1215.7	−1238.7	−1229.4	−1213.3	−1212.4
Nueces	−1084.8	−1092.2	−1135.9	−1084.0	−1083.8	−1082.2
Relative improvement						
Cheat	0.10	0.15	0.72	0.12	0.08	0
Driftwood	0.03	0.57	2.12	0.17	0.04	0
Holston	0.21	0.24	1.70	0.23	0.16	0
Pine	0.25	0.38	2.18	0.16	0.04	0
Schoharie	0.15	0.60	2.46	0.36	0.07	0
Towanda	0.38	1.21	3.09	0.21	0.04	0
Yosemite	0.40	0.27	2.13	1.38	0.08	0
Nueces	0.24	0.91	4.72	0.17	0.14	0

year test to the 100-year test, we can see the *RI* increases for most watersheds. This indicates that BHM can provide more improvement over the traditional methods in the test with a smaller data size. Furthermore, comparing BHM with "Zeros" (zero-skewness), we see that BHM produces better results for all watersheds, and the *RI* increases as the data quantity decreases. This is a direct demonstration of the advantage of BHM, since Zeros has the same shift parameter γ estimation as BHM, but Zeros estimates the shape and location parameters by MLE

instead of BHM.

3.5. CV results of the test with the proposed informative hyperprior

To evaluate the performance of the proposed informative hyperprior, we conduct the same experiments for the 100-year wet season dataset for each watershed, where the results of the proposed informative hyperprior and the flat hyperprior are compared. The parameter of the proposed informative hyperprior [k and the sign in Eq. (24)] for each watershed is obtained by fine tuning. The informative hyperpriors used for the wet season for each watershed are listed below.

For Cheat, Pine, and Towanda:

$$P(v, s_0^2) \propto \frac{1}{\sqrt{s_0^2 v - Ave(\tilde{\sigma}^2) \cdot v - 2 \cdot Ave(\tilde{\sigma}^2)}}. \tag{46}$$

For Holston, Driftwood and Nueces:

$$P(v, s_0^2) \propto \frac{1}{(s_0^2 v - Ave(\tilde{\sigma}^2) \cdot v - 2 \cdot Ave(\tilde{\sigma}^2))^4}. \tag{47}$$

For Schoharie:

$$P(v, s_0^2) \propto \frac{1}{(s_0^2 v - Ave(\tilde{\sigma}^2) \cdot v - 2 \cdot Ave(\tilde{\sigma}^2))^{\frac{1}{8}}}. \tag{48}$$

For Yosemite:

$$P(v, s_0^2) \propto \frac{1}{(s_0^2 v - Ave(\tilde{\sigma}^2) \cdot v + 2 \cdot Ave(\tilde{\sigma}^2))^{\frac{3}{2}}}. \tag{49}$$

We note that the proposed hyperprior parameter for each watershed may not be optimal. That is, a better parameter may exist, but how to find the optimal hyperprior parameter is beyond the scope of this paper. We also note that, compared to the data, the influence of the hyperprior on BHM is minimal.

To see the difference between the flat hyperprior and the informative hyperprior, we plot the samples from the posteriors of the hyperparameters [i.e. $P(v, s_0^2|S^2)$] based on two different hyperpriors in Fig. 10, using the 100-year Cheat watershed wet season dataset as the example. A total of 200,000 points are sampled from each posterior, and the first 3000 samples are dropped as the burn-in period in the MCMC process. From Fig. 10, we see both posteriors are proper without any outliers. And the informative hyperprior can provide a wider range of s_0^2 sample values than the flat hyperprior. Table 6 shows the CV results based on both hyperpriors.

Table 6 shows that first, for each watershed, results generated from the traditional methods based on two hyperpriors are exactly the same. This implies that changing the hyperprior will only impact the performance of BHM, since only BHM requires a hyperprior. Second, for each watershed, BHM performs better than the traditional methods under both hyperpriors, as the cumulative log-likelihood values are larger.
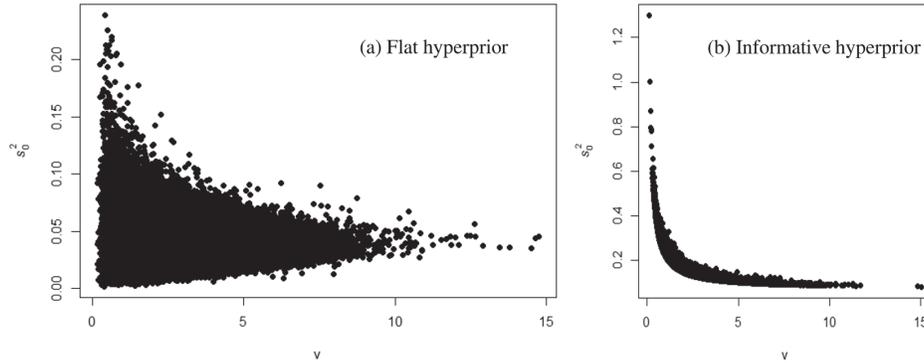
Fig. 10. (a) Samples from the posterior of hyperparameters based on the flat hyperprior (Cheat wet season); (b) Samples from the posterior of hyperparameters based on the informative hyperprior (Cheat wet season).

Table 6
CV test results of flat hyperprior and the proposed informative hyperprior.

Flat hyperprior (100-year wet season)						
	LMLE	L-MOM	MME	MMME	Zeros	BHM
Cumulative log-likelihood						
Cheat	−1321.63	−1316.22	−1313.72	−1318.62	−1316.84	−1316.25
Driftwood	−1132.10	−1132.38	−1133.90	−1131.87	−1131.92	−1131.75
Holston	−984.32	−987.04	−987.96	−984.70	−984.88	−984.17
Pine	−1169.71	−1170.35	−1172.73	−1170.75	−1169.21	−1168.61
Schoharie	−1208.67	−1208.79	−1211.50	−1209.86	−1208.18	−1206.45
Towanda	−1066.37	−1061.11	−1064.11	−1061.65	−1060.99	−1060.82
Yosemite	−1036.83	−1040.38	−1052.36	−1037.55	−1036.85	−1036.72
Nueces	−801.25	−804.61	−813.89	−803.90	−801.74	−801.57
Informative hyperprior (100-year wet season)						
Cumulative log-likelihood						
Cheat	−1321.63	−1316.22	−1313.72	−1318.62	−1316.84	−1316.17
Driftwood	−1132.10	−1132.38	−1133.90	−1131.87	−1131.92	−1131.70
Holston	−984.32	−987.04	−987.96	−984.70	−984.88	−983.99
Pine	−1169.71	−1170.35	−1172.73	−1170.75	−1169.21	−1168.56
Schoharie	−1208.67	−1208.79	−1211.50	−1209.86	−1208.18	−1206.43
Towanda	−1066.37	−1061.11	−1064.11	−1061.65	−1060.99	−1060.70
Yosemite	−1036.83	−1040.38	−1052.36	−1037.55	−1036.85	−1036.71
Nueces	−801.25	−804.61	−813.89	−803.90	−801.74	−801.56

Table 7
Relative improvement (*RI*) results of 28-year and 60-year tests for each method (BHM is the base model).

	LMLE	L-MOM	MME	MMME	Zeros	BHM
60-year (dry season)						
Cheat	0.11	0.19	1.32	0.21	0.11	0
Driftwood	−0.07	1.11	4.32	0.26	0.04	0
Holston	0.36	0.26	3.37	0.41	0.27	0
Pine	−0.12	0.67	4.27	0.12	0.02	0
Schoharie	0.20	0.99	4.59	0.82	0.08	0
Towanda	0.66	2.54	6.08	0.43	0.03	0
Yosemite	0.15	0.47	3.82	2.09	−0.02	0
Nueces	0.47	1.41	6.88	0.24	0.14	0
28-year (dry season)						
Cheat	0.30	0.39	1.84	0.16	0.30	0
Driftwood	2.31	0.10	5.26	0.75	0.05	0
Holston	1.76	2.35	7.78	4.48	1.91	0
Pine	0.19	1.45	4.85	0.59	0.21	0
Schoharie	−0.14	0.76	5.16	1.18	0.10	0
Towanda	0.26	2.02	6.98	0.76	0.07	0
Yosemite	1.25	1.73	8.05	2.04	0.39	0
Nueces	1.88	1.99	11.86	1.90	1.45	0

Third, comparing the flat hyperprior results with the informative hyperprior results, we see that for each watershed, the latter's log-likelihood value is a little higher than the former, but the improvements are insignificant. This indicates that adopting the informative hyperprior can still benefit BHM; however, compared to the data layer, the improvements of using different hyperpriors on BHM should be minimal.

3.6. CV results of the test with an extremely small dataset size

Not all rivers have a sufficiently long historical observation series (i.e. 100 years or 60 years of observations). In fact, for many watersheds, the historical observation series is very short, and the data are very limited. To compare BHM with the traditional methods based on a short time series, we use the most recent 28 years (i.e. small size) of historical observations for each watershed. Then for the most recent 60-year and 28-year datasets, we conduct the tests for each watershed and each method. Because the 28-year series is too short for many watersheds, the months in the wet season (i.e. Dec.–May) may have negative sample skewness, which prohibits any method from estimating LN3 parameters. Thus, in this section, we use dry season months (i.e. Jun.–Nov.) as the example for comparison. To test if BHM performs

Table 8
Comparison of joint log-likelihoods between AR(1) and BHM.

Model	Cheat	Driftwood	Holston	Pine	Schoharie	Towanda	Yosemite	Nueces
AR(1)	−1.06	−80.23	−63.02	−106.40	−164.86	−120.96	−17.01	−108.24
BHM	1.53	−78.51	−61.90	−106.29	−163.19	−121.68	−46.34	−181.94

better than the selected traditional methods, we calculate the relative improvements for each method. Table 7 shows the test results. The flat hyperprior is used in the test.

It is evident from Table 7 that most *RI*'s in the 28-year test are greater than the corresponding *RI* in the 60-year test. This can be interpreted as: 1) The base model (BHM) outperforms the selected traditional methods in most watersheds for both datasets; and 2) Compared with the medium size dataset (60 years), the advantage of BHM over the traditional methods is more noticeable in the small size data set (28 years), particularly relative to the MME, MMME, and Zeros methods. This is because BHM utilizes other months' information, which "expands" the information contained in the limited data. Thus, BHM is especially suited for parameter estimation where historical observations are limited.

4. Discussion

4.1. Comparison of BHM with an autoregressive model

BHMs are used to account for correlation (dependence) between different months. It would be of interest to compare the performance of BHM with an autoregressive (AR) model, as AR models are used to account for temporal dependence.

Recall in our BHM that the temporal dependence is captured from the log-transformed data y in different months [i.e. $y = \log(x - \gamma)$]. We therefore use the same data to build an AR model and compare results. A typical AR(q) model can be expressed as

$$y_t = \beta_0 + \beta_1 \cdot y_{t-1} + \beta_2 \cdot y_{t-2} + \dots \beta_q \cdot y_{t-q} + \varepsilon. \tag{50}$$

This model only can capture linear dependence. However, our proposed BHM is a data-driven model, which means the data itself can decide the relation among different months and the correlation (dependence) can be of any type, not limited to linear relation.

The AR model also defines a joint distribution over $y_1, \dots y_{12}$, and we can compare it with our BHM by the test data likelihood. Consider the wet season (i.e. $y_{12}, \dots y_5$) of a given test watershed. We propose the following steps for a comparative analysis:

1. For simplicity, we build an AR(1) model for the wet season (i.e. Dec, Jan,…, May) with training dataset (e.g. $y_5 = \beta_0 + \beta_1 \cdot y_4 + \varepsilon$). Assume the error term ε follows $N(0, \sigma^2)$ and the variance σ^2 can be estimated by an unbiased estimator s^2 [i.e. $s^2 = SSE/(N_{tratin} - 2)$, where SSE is the sum square error and N_{train} is the number of observations in the training dataset].

2. Based on the AR(1) model, the example conditional density distribution can be expressed as $y_5|y_4 \sim N(y_5|\hat{\beta}_0 + \hat{\beta}_1 \cdot y_4, s^2)$, and similarly for $y_{12}|y_{11} \dots y_4|y_3$.

3. We calculate test likelihood for the AR model using the test dataset. For example, the test likelihood of $p(y_5|y_4)$ is $p(y_5|y_4) = \prod_i^{Ntest} N(y_5(i)|\hat{\beta}_0 + \hat{\beta}_1 \cdot y_4(i), s^2)$, where N_{test} is the observation numbers in the test dataset, and similarly for $p(y_{12}|y_{11}) \dots p(y_4|y_3)$ as $p(y_5|y_4)$.

4. We use the training data to estimate the first month in the wet season (i.e. December) as a normal distribution [i.e. $y_{12} \sim N(\hat{\theta}_{12}, \hat{\sigma}_{12}^2)$]. Note this is a marginal density distribution. We then use the test dataset to calculate its likelihood: $p(y_{12}) = \prod_i^{Ntest} N(y_{12}(i)|\hat{\theta}_{12}, \hat{\sigma}_{12}^2)$.

5. We calculate the joint probability density $p(y_{12}, y_1, \dots y_5)$ based on the product rule: $p(y_{12}, y_1, \dots y_5)_{AR} = p(y_{12}) \cdot p(y_1|y_{12}) \cdot \dots \cdot p(y_5|y_4)$. This

is also the joint likelihood of the AR model.

6. From our BHM, we have obtained $y_j \sim N(\hat{\theta}_j^B, \hat{\sigma}_j^{2B})$ from the training dataset for all six months. And these distributions are independent, so their joint probability density can be calculated as $p(y_{12}, y_1, \dots y_5)_{BHM} = p(y_{12}) \cdot p(y_1) \cdot \dots \cdot p(y_5)$. Then we use the test dataset to calculate the test likelihood of BHM, which is also the joint likelihood of BHM.

7. We compare two joint (log) likelihoods between AR(1) and BHM. The method with the larger joint log-likelihood would be a better model.

Note that we calculate the joint likelihood for each method for comparison because we cannot directly compare $y_5|y_4 \sim N(y_5|\hat{\beta}_0 + \hat{\beta}_1 \cdot y_4, s^2)$ from the AR(1) model with $y_5 \sim N(\hat{\theta}_5^B, \hat{\sigma}_5^{2B})$ from the BHM. This is because the former is a conditional distribution, but the latter is a marginal distribution, and they are not comparable. Thus, we need to convert them to the joint distributions for the comparison (i.e. as shown in steps 5 and 6).

Based on this procedure, we have calculated the joint log-likelihoods for AR(1) and BHM for all eight watersheds. The results are shown in the Table 8.

From Table 8, we see that the BHM performs better than AR(1) for most of the watersheds, because the joint log-likelihood of BHM is larger. However, for Yosemite and Nueces, the AR(1) model has a larger joint log-likelihood. This implies that, for these two watersheds, monthly flows indeed follow an AR(1) model. Therefore, despite its limitations, the AR model is indeed a good model accounting for temporal dependence, and it can perform very well in some watersheds. However, our proposed BHM is a more general data-driven model and requires fewer assumptions.

4.2. Potential uncertainty loss due to sequential estimation

Our methodology of construing BHM follows a sequential approach. We first estimated shift parameter γ and fixed it. We then estimated σ^2 and also fixed it, and finally we estimated θ. A potential shortcoming of this approach is that not all parameter uncertainties may be properly reflected in the final estimates. A full Bayesian analysis based on the joint posterior [i.e. $P(\gamma, \sigma^2, \theta|y)$] would be most ideal; however, it would be very difficult to carry this out in practice.

We investigate how much uncertainty we may lose with our sequential approach when estimating γ. In the Table 9, we calculate the 90% confidence interval of γ estimates, using Schoharie dry season 100-year data as an example.

Table 9
90% confidence interval of γ estimates.

Confidence Interval	Jun	Jul	Aug	Sep	Oct	Nov
γ 90% upper bound	26.5	10.1	10.4	6.3	0	−63.6
γ estimates	7.1	5.6	8.8	3.4	−8.4	−261.3
γ 90% lower bound	−12.2	1.3	7.2	4.9	−16.9	−459
100-year average streamflow (x)	330.3	166.7	119.2	200.1	302.3	517.3
1/(x-γ_lower)	0.0029	0.0060	0.0089	0.0051	0.0031	0.0010
1/(x-γ_upper)	0.0033	0.0064	0.0092	0.0052	0.0033	0.0017
ln(x-γ_lower)	5.84	5.11	4.72	5.27	5.77	6.88
ln(x-γ_upper)	5.72	5.05	4.69	5.27	5.71	6.36

[Table 9](#) shows that the loss of uncertainty for γ is negligible, when compared with the magnitude of each month's real monthly-average streamflow (i.e. x). According to the probability density function of LN3 [i.e. Eq. [(2)](#)], the potential uncertainty loss for γ can be measured by the uncertainty losses of $1/(x - \gamma)$ and $\ln(x - \gamma)$. Because of the small differences in the last four rows in [Table 9](#), these uncertainty losses are insignificant, especially for $1/(x - \gamma)$. Although some uncertainty loss indeed occurred in $\ln(x - \gamma)$, the loss is minimum and negligible for most months.

In addition, even though our sequential approach loses some uncertainty, it doesn't lose information on shrinkage (i.e. the shrinkage for estimating σ^2 and θ). In fact, the variance parameter (i.e. shape parameter σ^2) is most likely the one that needs to be stabilized. In other words, σ^2 will benefit the most from the shrinkage (i.e. BHM) among all the parameters. This is also the reason why our proposed variance BHM is important. So, if we assume γ is perfectly estimated, then there is no uncertainty loss for the σ^2 estimation since it is first estimated in our sequential approach (i.e. estimated before θ).

5. Conclusion

In this paper, we proposed a Bayesian hierarchical model (BHM) for estimating the statistical parameters for monthly average streamflows. We assumed that the monthly average streamflow follows a three-parameter, log-normal distribution (LN3). The three underlying statistical parameters associated with LN3 are shift, shape, and location.

We selected eight watersheds in the United States, where historical unimpaired streamflows have been collected. We first conducted a Shapiro-Wilk normality test to validate the LN3 assumption. We then developed the BHM and proposed an informative hyperprior. We used Cross-Validation (CV) to evaluate the performance of BHM against selected traditional parameter estimation method methods, including local maximum likelihood estimation (LMLE), L-moment (L-MOM), method of moment estimation (MME), modified method of moment estimation (MMME), and zero-skewness (Zeros). We calculated the log-likelihood, cumulative log-likelihood, and relative improvement for all methods. We performed comparative analyses using different dataset sizes and different hyperpriors.

The results show that the proposed BHM produced better parameter estimates than the selected traditional parameter estimation methods for all data sizes (long, medium, and short). The fewer the observation

data, the more the proposed method improves compared with the traditional methods. The two reasons that BHM produced better parameter estimates are: 1) BHM utilizes historical observations not only from the month under consideration but also from all other months, and 2) Due to shrinkage, the Bayesian estimator from BHM is guaranteed to be no worse than the MLE-based estimator. The proposed BHM is especially suited for parameter estimation where historical observations are limited. In addition, when compared to an AR model, the proposed BHM shows its advantage as a data-driven model with fewer assumptions.

We have demonstrated the advantages of the proposed method for parameter estimation. A disadvantage of the method is that it takes much longer running time to get the results (primarily due to MCMC simulation). For example, the Zeros method may take just a few seconds, while the proposed method takes around 30 min per run. The proposed sequential estimation approach is easy to carry out in practice, but may incur loss of uncertainty. However, the loss is insignificant and negligible.

Appendix A

Derivation of the posterior distribution of σ_j^2 [Dist. [(9)](#)]
From Bayes rule:

$$P(\sigma_j^2|v, s_0^2, S_j^2) \propto P(\sigma_j^2, v, s_0^2, S_j^2) = P(v, s_0^2) \cdot P(\sigma_j^2|v, s_0^2) \cdot P(S_j^2|\sigma_j^2). \tag{1a}$$

which is equivalent to:

$$P(\sigma_j^2|v, s_0^2, S_j^2) \propto P(\sigma_j^2|v, s_0^2) \cdot P(S_j^2|\sigma_j^2). \tag{2a}$$

Based on the Eq. [(8)](#) and a change of variable, we can derive:

$$P(S_j^2|\sigma_j^2) = \frac{\left(\frac{d}{2\sigma_j^2}\right)^{d/2}}{\Gamma(d/2)} \cdot S_j^{2(\frac{d}{2}-1)} \cdot e^{-\frac{dS_j^2}{2\sigma_j^2}} \propto S_j^{2(\frac{d}{2}-1)} \cdot \sigma_j^{2\frac{-d}{2}} \cdot e^{-\frac{dS_j^2}{2\sigma_j^2}}. \tag{3a}$$

The pdf of $Inv - \chi^2(v, s_0^2)$ is

$$P(\sigma_j^2|v, s_0^2) = \frac{(s_0^2 v/2)^{v/2}}{\Gamma(v/2)} \cdot \sigma_j^{2-(\frac{v}{2}+1)} \cdot e^{\frac{-s_0^2 v}{2\sigma_j^2}}. \tag{4a}$$

Then based on Eq. [(2a)](#), we have:

$$P(\sigma_j^2|v, s_0^2, S_j^2) \propto \sigma_j^{2-(\frac{v+d}{2}+1)} \cdot e^{-\left(\frac{s_0^2 v + dS_j^2}{2\sigma_j^2}\right)} \tag{5a}$$

which is equivalent to:

$$P(\sigma_j^2|v, s_0^2, S_j^2) \sim Inv - \chi^2\left(v + d, \frac{vs_0^2 + dS_j^2}{v + d}\right).$$

(6a)

This is exactly Eq. (9).

Appendix B

Derivation of the posterior distribution of hyperparameters $P(v, s_0^2|S^2)$ [i.e. Eq. (10)]

Based on the Bayes rule and the fact that S_j^2 are independent with each other ($j = 1, ...,6$), we have:

$$P(v, s_0^2|S^2) \propto P(v, s_0^2, S^2) \propto P(v, s_0^2) \cdot P(S^2|v, s_0^2) = P(v, s_0^2) \cdot \prod_{j=1}^{J=6} P(S_j^2|v, s_0^2).$$

(1b)

From Eq. (3a) we already have:

$$P(S_j^2|\sigma_j^2) = \frac{\left(\frac{d}{2\sigma_j^2}\right)^{d/2}}{\Gamma(d/2)} \cdot S_j^{2\left(\frac{d}{2}-1\right)} \cdot e^{-\frac{dS_j^2}{2\sigma_j^2}}.$$

(2b)

Then we claim that:

$$P(S_j^2|\sigma_j^2) = P(S_j^2|\sigma_j^2, v, s_0^2).$$

(3b)

This is because once σ_j^2 is given, S_j^2 can be determined only by σ_j^2, so it is independent of the parameters v and s_0^2. This condition can be summarized as:

$$S_j^2 \perp (v, s_0^2)|\sigma_j^2,$$

(4b)

which implies Eq. (3b).

Bayes rule with Eq. (3b) implies:

$$P(S_j^2, \sigma_j^2|v, s_0^2) = P(S_j^2|\sigma_j^2) \cdot P(\sigma_j^2|v, s_0^2).$$

(5b)

Based on Eqs. (5b) and (4a), we can derive:

$$P(S_j^2, \sigma_j^2|v, s_0^2) = \frac{\left(\frac{d}{2\sigma_j^2}\right)^{d/2}}{\Gamma(d/2)} \cdot S_j^{2\left(\frac{d}{2}-1\right)} \cdot e^{-\frac{dS_j^2}{2\sigma_j^2}} \cdot \frac{(s_0^2 v/2)^{v/2}}{\Gamma(v/2)} \cdot \sigma_j^{2^{-\left(\frac{v}{2}+1\right)}} \cdot e^{\frac{-s_0^2 v}{2\sigma_j^2}}.$$

(6b)

Integrating σ_j^2 out from Eq. (6b), we have:

$$P(S_j^2|v, s_0^2) = \int P(S_j^2, \sigma_j^2|v, s_0^2)\, d\sigma_j^2$$

$$= \frac{(s_0^2 v/2)^{v/2}}{\Gamma(v/2)} \cdot \frac{\left(\frac{d}{2}\right)^{d/2}}{\Gamma(d/2)} \cdot S_j^{2\left(\frac{d}{2}-1\right)} \cdot \frac{\Gamma\left(\frac{v+d}{2}\right)}{\left[\frac{(vs_0^2 + dS_j^2)}{2}\right]^{\frac{v+d}{2}}}.$$

(7b)

Substituting Eq. (7b) into Eq. (1b), we have:

$$P(v, s_0^2|S^2) \propto P(v, s_0^2) \cdot \prod_{j=1}^{J=6} \frac{(s_0^2 v/2)^{v/2}}{\Gamma(v/2)} \cdot \frac{\Gamma\left(\frac{v+d}{2}\right)}{\left[\frac{(vs_0^2 + dS_j^2)}{2}\right]^{\frac{v+d}{2}}}.$$

(8b)

Simplifying (8b) by introducing the Beta function, we get

$$P(v, s_0^2|S^2) \propto P(v, s_0^2) \cdot \prod_{j=1}^{J=6} \frac{1}{Beta\left(\frac{v}{2}, \frac{d}{2}\right)} \cdot \frac{(s_0^2 v/2)^{v/2}}{\left[\frac{(vs_0^2 + dS_j^2)}{2}\right]^{\frac{v+d}{2}}}.$$

(9b)

This is exactly Eq. (10).

References

Akaike, H., 1998. Information theory and an extension of the maximum likelihood principle. In: Selected Papers of Hirotugu Akaike. Springer, New York, NY, pp. 199–213.

Bobee, B., 1975. The log Pearson type 3 distribution and its application in hydrology. Water Resour. Res. 11 (5), 681–689.

Berger, J.O., Wolpert, R.L., 1988. The Likelihood Principle. IMS.

Bobee, B., and Ashkar, F., 1991. The gamma family and derived distributions applied in hydrology, Water Resour. Publ. Colo (No. GB656. 2. M34. B63 1991.).

Bowers, M.C., Tung, W.W., Gao, J.B., 2012. On the distributions of seasonal river flows: lognormal or power law? Water Resour. Res. 48 (5).

Berger, J.O., 2013. Statistical Decision Theory and Bayesian Analysis. Springer Science & Business Media.

Bracken, C., Rajagopalan, B., Cheng, L., Kleiber, W., Gangopadhyay, S., 2016. Spatial Bayesian hierarchical modeling of precipitation extremes over a large domain. Water Resour. Res. 52 (8), 6643–6655. https://doi.org/10.1002/2016WR018768.

Bracken, C., Holman, K.D., Rajagopalan, B., Moradkhani, H., 2018. A Bayesian hierarchical approach to multivariate nonstationary hydrologic frequency analysis. Water Resour. Res. 54 (1), 243–255.

Cohen, A.C., 1951. Estimating parameters of logarithmic-normal distributions by

maximum likelihood. J. Am. Stat. Assoc. 46, 206–212.

Calitz, F., 1973. Maximum likelihood estimation of the parameters of the three-parameter lognormal distribution–a reconsideration. Aust. J. Stat. 15 (3), 185–190.

Cohen, A.C., B.J. Whitten,, 1980. Estimation in the three-parameter lognormal distribution. J. Am. Stat. Assoc. 75, 399–404.

Chaloner, K., 1987. A Bayesian approach to the estimation of variance components for the unbalanced one-way random model. Technometrics 29 (3), 323–337.

Cohen, A.C., 1988. Three-parameter estimation. In Crow, E.L., Shimizu, K. (Eds.) Lognormal Distributions: Theory and Applications. Marcel Dekker, New York (Chapter 4).

Cooley, D., Nychka, D., Naveau, P., 2007. Bayesian spatial modeling of extreme precipitation return levels. J. Am. Stat. Assoc. 102 (479), 824–840. https://doi.org/10.1198/016214506000000780.

Duncan Cramer, 1997. Fundamental Statistics for Social Research. Routledge. ISBN 9780415172042, p. 85.

Doane, D.P., Seward, L.E., 2011. Measuring skewness: a forgotten statistic? J. Stat. Educ. 19 (2).

Devineni, N., Lall, U., Pederson, N., Cook, E., 2013. A tree-ring-based reconstruction of Delaware River basin streamflow using hierarchical Bayesian regression. J. Clim. 26 (12), 4357–4374. https://doi.org/10.1175/JCLI-D-11-00675.1.

Efron, B., Morris, C., 1977. Stein's paradox in statistics. Sci. Am. 236 (5), 119–127.

Edwards, M., 1992. Likelihood. Johns Hopkins.

El Adlouni, S., Ouarda, T.B., 2009. Joint Bayesian model selection and parameter estimation of the generalized extreme value model with covariates using birth-death Markov chain Monte Carlo. Water Resour. Res. 45 (6).

Fraser, D.A.S., 1963. On the sufficiency and likelihood principles. J. Am. Stat. Assoc. 58 (303), 641–647.

Fisher, R.A., 1992. Statistical methods for research workers. In: Breakthroughs in Statistics. Springer, New York, NY, pp. 66–70.

Griffiths, D.A., 1980. Interval estimation for the three-parameter lognormal distribution via the likelihood function. Appl. Stat. 29, 58–68.

Griffis, V.W., Stedinger, J.R., 2007. Log-Pearson Type 3 distribution and its application in flood frequency analysis. I: distribution characteristics. J. Hydrol. Eng. 12 (5), 482–491.

Ghosh, S., Mallick, B.K., Wiley, C.J., 2011. A hierarchical Bayesian spatio-temporal model, pp. 192–204. https://doi.org/10.1002/env.1043.

Gelman, A., Stern, H.S., Carlin, J.B., Dunson, D.B., Vehtari, A., Rubin, D.B., 2013. Bayesian Data Analysis. Chapman and Hall/CRC.

Heyde, C.C., 1963. On a property of the lognormal distribution. J. R. Stat. Soc. B 25, 392–393.

Hill, B.M., 1963. The three-parameter lognormal distribution and Bayesian analysis of a point-source epidemic. J. Am. Stat. Assoc. 58, 72–84.

Harter, H.L., A.H. Moore,, 1966. Local-maximum-likelihood estimation of the parameters of three-parameter lognormal populations from complete and censored samples. J. Am. Stat. Assoc. 61, 842–851.

Hosking, J.R.M., Wallis, J.R., Wood, E.F., 1985a. An appraisal of the regional flood frequency procedure in the UK Flood Studies Report. Hydrol. Sci. J. 30 (1), 85–109.

Hosking, J.R.M., Wallis, J.R., Wood, E.F., 1985b. Estimation of the generalized extreme-value distribution by the method of probability-weighted moments. Technometrics 27 (3), 251–261.

Hosking, J.R.M., 1990. L-moments: analysis and estimation of distributions using linear combinations of order statistics. J. R. Stat. Soc.: Ser. B (Methodol.) 52 (1), 105–124.

Hosking, J.R.M., Wallis, J.R., 1996. Regional frequency analysis of floods in central Appalachia. Res. Rep, RC20349.

Hoff, P.D., 2009. A First Course in Bayesian Statistical Methods, vol. 580 Springer, New York.

Johnson, N.L., Kotz, S., Balakrishnan, N., 1994. Continuous Univariate Distributions, vol. 1, second ed. John Wiley and Sons, New York.

Joanes, D.N., Gill, C.A., 1998. Comparing measures of sample skewness and kurtosis. J. R. Stat. Soc.: Ser. D (The Statistician) 47 (1), 183–189.

Kowarik, A, Elnorm3 Package, EnvStats v2.3.1, R Documentation, https://www.rdocumentation.org/packages/EnvStats/versions/2.3.1/topics/elnorm3.

Kullback, S., Leibler, R.A., 1951. On information and sufficiency. Ann. Math. Stat. 22 (1), 79–86.

Kendall, M.G.; Stuart, A., 1969. The advanced theory of statistics. In: Distribution Theory, vol. 1, third ed., Griffin ISBN 0-85264-141-9 (Ex 12.9).

Kirby, W., 1974. Algebraic boundedness of sample statistics. Water Resour. Res. 10 (2), 220–222.

Kohavi, R., 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. In Ijcai 14 (2), 1137–1145.

Kullback, S., 1997. Information Theory and Statistics. Courier Corporation.

Kholevo, A.S., 2001. Sufficient statistic. Encyclopedia of Mathematics, Springer, Berlin/Heidelberg.

Lima, R.H.C., Lall, U., 2009. Hierarchical bayesian modeling of multisite daily rainfall occurrence (Ph.D dissertation, Ph.D Thesis). Colombia University, New York. https://doi.org/10.1029/2008WR007485.

Li, J., Zhu, F., Xu, B., Yeh, W.W.G., 2019. Streamflow scenario tree reduction based on conditional Monte Carlo sampling and regularized optimization. J. Hydrol. 577, 123943.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E., 1953. Equation of state calculations by fast computing machines. J. Chem. Phys. 21 (6), 1087–1092.

Markovic, R.D., 1965. Probability functions of best fit to distributions of annual precipitation and runoff. Hydrology papers (Colorado State University); no. 8.

Morrison, J.E., Smith, J.A., 2002. Stochastic modeling of flood peaks using the generalized extreme value distribution. Water Resour. Res. 38 (12).

Myung, I.J., 2003. Tutorial on maximum likelihood estimation. J. Math. Psychol. 47 (1), 90–100.

McLachlan, G.J., Do, K.A., Ambroise, C., 2005. Analyzing Microarray Gene Expression Data, vol. 422. John Wiley & Sons.

Millard, 2013. EnvStats: An R Package for Environmental Statistics. Springer, ISBN 978-1-4614-8455-4, http://www.springer.com/book/9781461484554.

Najafi, M.R., Moradkhani, H., 2013. Analysis of runoff extremes using spatial hierarchical Bayesian modeling. Water Resour. Res. 49 (10), 6656–6670. https://doi.org/10.1002/wrcr.20381.

Parzen, E., 1962. On estimation of a probability density function and mode. Ann. Math. Stat. 33 (3), 1065–1076.

Rosenblatt, M., 1956. Remarks on Some Nonparametric Estimates of a Density Function. Ann. Math. Stat. 27 (3), 832–837. https://doi.org/10.1214/aoms/1177728190.

Raiffa, H., Schlaifer, R., 1961. Applied Statistical Decision Theory.

Rao, D.V., 1980. Log Pearson Type 3 distribution: a generalized evaluation. J. Hydraul. Div. 106(ASCE, 15391).

Royston, J.P., 1992. Estimation, reference ranges and goodness of fit for the three-parameter log-normal distribution. Stat. Med. 11, 897–912.

Rosbjerg, D., Madsen, H., 1995. Uncertainty measures of regional flood frequency estimators. J. Hydrol. 167 (1–4), 209–224.

Royston, P., 1995. Remark AS R94: a remark on algorithm AS 181: the W-test for normality. J. R. Stat. Soc.: Ser. C (Appl. Stat.) 44 (4), 547–551.

Ross, S.M., 2006. A First Course in Probability, vol. 7 Pearson Prentice Hall, Upper Saddle River, NJ.

Refaeilzadeh, P., Tang, L., Liu, H., 2009. Cross-validation. Encycl. Database Syst. 532–538.

Renard, B., 2011. A Bayesian hierarchical approach to regional frequency analysis. Water Resour. Res. 47 (11). https://doi.org/10.1029/2010WR010089.

Shapiro, S.S., Wilk, M.B., 1965. An analysis of variance test for normality (complete samples). Biometrika 52 (3/4), 591–611.

Sangal, B.P., Biswas, AK., 1970. Three parameter lognormal distribution and its applications in hydrology. Water Resour. Res. 6(2), 505–515.

Snyder, W.M., Wallace, I.R., 1974. Fitting a three-parameter lognormal distribution by least squares. Nordic Hydrol. 5, 129–145.

Schwarz, G., 1978. Estimating the dimension of a model. Ann. Stat. 6 (2), 461–464.

Stedinger, J.R., 1980. Fitting lognormal distributions to hydrologic data. Water Resour. Res. 16 (3), 481–490.

Singh, V.P., Rajagopal, A.K., Singh, K., 1986. Derivation of some frequency distributions using the principle of maximum entropy. Adv. Water Resour. 9, 91–106.

Singh, V.P., Singh, K., 1987. Parameter estimation for TPLN distribution for flood frequency analysis. Water Resour. Bull. 23(6), 1185–1191.

Smith, J.A., 1987. Estimating the upper tail of flood frequency distributions. Water Resour. Res. 23 (8), 1657–1666.

Singh, V.P., Cruise, I.F., Ma, M., 1990. A comparative evaluation of the estimators of the three-parameter lognormal distribution by Monte Carlo simulation. Comput. Stat. Data Anal. 10, 71–85.

Stedinger, J.R., Lu, L.H., 1995. Appraisal of regional and index flood quantile estimators. Stochastic Hydrol. Hydraul. 9 (1), 49–75.

Steel, D., 2007. Bayesian confirmation theory and the likelihood principle. Synthese 156 (1), 53–77.

Sang, H., Gelfand, A.E., 2009. Hierarchical modeling for extreme values observed over space and time. Environ. Ecol. Stat. 16 (3), 407–426. https://doi.org/10.1007/s10651-007-0078-0.

Singh, V.P., 2013. Entropy-based Parameter Estimation in Hydrology, vol. 30. Springer Science & Business Media.

Steinschneider, S., Lall, U., 2015. A hierarchical Bayesian regional model for nonstationary precipitation extremes in Northern California conditioned on tropical moisture exports. Water Resour. Res. 51 (3), 1472–1492. https://doi.org/10.1002/2014WR016664.

Vogel, R.M., Wilson, I., 1996. Probability distribution of annual maximum, mean, and minimum streamflows in the United States. J. Hydrol. Eng. 1 (2), 69–76.

Wallis, J.R., Matalas, N.C., Slack, J.R., 1974. Just a moment!. Water Resour. Res. 10 (2), 211–219.

Wolpert, R.L., 2010. Conditional Expectation. University Lecture.

Wei, W.U., Clark, J.S., Vose, J.M., 2012. Application of a full hierarchical Bayesian model in assessing streamflow response to a climate change scenario at the Coweeta Basin, NC, USA. J. Resour. Ecol. 3(2), 118–129. https://doi.org/10.5814/j.issn.1674-764x.2012.02.003.

Xiong, L., Yu, K.X., Gottschalk, L., 2014. Estimation of the distribution of annual runoff from climatic variables using copulas. Water Resour. Res. 50 (9), 7134–7152.

Xu, B., Zhong, P.A., Zambon, R.C., Zhao, Y., Yeh, W.W.G., 2015a. Scenario tree reduction in stochastic programming with recourse for hydropower operations. Water Resour. Res. 51 (8), 6359–6380.

Xu, B., Zhong, P.A., Stanko, Z., Zhao, Y., Yeh, W.W.G., 2015b. A multiobjective short-term optimal operation model for a cascade system of reservoirs considering the impact on long-term energy production. Water Resour. Res. 51 (5), 3353–3369.

Yeh, W.W.G., 1985. Reservoir management and operations models: A state-of-the-art review. Water Resour. Res. 21 (12), 1797–1818.

Yue, S., Hashino, M., 2007. Probability distribution of annual, seasonal and monthly precipitation in Japan. Hydrol. Sci. J. 52 (5), 863–877.

Yan, H., Moradkhani, H., 2015. A regional Bayesian hierarchical model for flood frequency analysis. Stochastic Environ. Res. Risk Assess. 29 (3), 1019–1036. https://doi.org/10.1007/s00477-014-0975-3.

23

Chapter 3

Research papers

Streamflow scenario tree reduction based on conditional Monte Carlo sampling and regularized optimization

Jinshu Li[a], Feilin Zhu[a,b], Bin Xu[b], William W.-G. Yeh[a,*]

[a] Department of Civil and Environmental Engineering, University of California, Los Angeles, CA, USA
[b] College of Hydrology and Water Resources, Hohai University, Nanjing, China

ABSTRACT

Streamflow scenario tree reduction is essential for alleviating the computational burden of a stochastic programming with recourse model. This paper develops a new streamflow scenario tree reduction method aimed at preserving important statistical moment information and maintaining streamflow scenario probability. Specifically, we first employ a neural gas algorithm for scenario tree generation, then establish a stepwise conditional Monte Carlo sampling method for systemically reducing the number of scenarios from the full tree. We then develop a regularized optimization model based on ridge regression and moment matching to determine the posterior scenario probability. We apply the proposed method to the Qingjiang cascade reservoir system in China. The results show that the reduced tree with 35% reduction level can still maintain robust moment preservations, including the mean, variance, lag-one covariance, cross-site covariance, and scenario probability. Additionally, the stability test indicates that the proposed conditional Monte Carlo sampling method is stable and converges within a reasonable number of scenario combinations.

1. Introduction

Handling streamflow uncertainty is a major issue in reservoir management and operation. Accordingly, the stochasticity associated with long-term streamflow prediction must be considered in an analysis. However, methods based on deterministic streamflow prediction for reservoir operations suffer from the risk of making inappropriate release decisions, which may incur heavy losses of benefit (Li et al., 2009; Zhao et al., 2011; Zhu et al., 2018). To account for streamflow uncertainty, the multistage stochastic programming with recourse model has been developed and applied to solve water resources management problems in which streamflow is modeled as a stochastic process (Birge and Louveaux, 2011; Yeh, 1985; Zhu et al., 2017). This model requires an input of a discretized streamflow scenario tree that can most represent the future random streamflow and the occurrence probability of each scenario. The goal of the model is then to minimize the expected loss or maximize the expected benefit.

Existing scenario tree generation methods can be categorized in three broad ways: those based on simulation, on clustering, and on optimization. Kaut and Wallace (2007) discussed methods based on conditional simulation that sample the nodal values or scenarios from known distributions of random variables. These methods are implemented easily and able to preserve transition probability, but they only can generate uncorrelated random vectors and, generally, suffer from a problem of dimensionality because of the size of the random vector. Methods based on clustering, however, do not require random variable distributions. Instead, these methods pursue the most representative scenarios (i.e., centroids of observation data) by clustering observation data (Hansen and Jaumard, 1997; Šutiene et al., 2010). A unique feature of the clustering methods is that the mean value of the generated scenario tree always equals the mean of the observation data. The neural gas clustering method is a neural network algorithm originally used for vector quantization in topology (Martinetz and Schulten, 1991; Martinetz et al., 1993). It requires a fixed pre-specified tree structure and uses a distance-based iterative method to update nodal values to centroids. Latorre et al. (2007) compared four popular clustering techniques (including the conditional clustering method, neural gas method, node clustering method, and progressive clustering method) and conducted numerical experiments to test their performance by calculating the quantization error for each method. The error measures how well the resulting scenario tree fits the original distribution. Based on the results from the numerical experiments, they concluded that the neural gas method performs best in the application to hydro inflow. In general, a key advantage of clustering-based methods is that the sample mean can be preserved very well, and the computational cost is minimum. However, clustering-based methods

usually fail to preserve other statistical moments and co-moments other than the mean. This introduces a significant loss of important sample information from the generated scenario tree, such as high and low flows (Xu et al., 2015).

Methods based on optimization mostly refer to moment matching, which aims to match different statistical moments and co-moments between the scenario tree and historical data series (Høyland and Wallace, 2001). In general, there are two approaches to implementing moment matching: the sequential approach that generates the tree node by node, and the overall approach that generates the entire tree in a single optimization. However, both approaches have drawbacks (Høyland and Wallace, 2001). The sequential approach requires that the first-stage tree satisfy the first-stage statistical properties. This may result in conditional second-stage properties that are impossible to match, and consequently, it may produce a suboptimal tree. Also, since the sequential approach updates the nodal values and probability at each stage, for a large multi-stage scenario tree generation problem it will require solving a large number of optimization problems and is computationally expensive. In contrast, the overall approach accounts for the entire tree's statistical property and only requires one single optimization for multi-stage problems. However, the degree of non-convexity increases significantly with the scale of the optimization problem, and it is very difficult to obtain a good match for large-scale problems. In addition, Hochreiter and Pflug (2007) provided an example of four distributions that coincide with the first four moments. The results show that the moment matching method may not be able to match the target distribution.

Despite the shortcomings of the original moment matching method, the method has been widely applied to streamflow scenario tree generation (Vitoriano et al., 2000) and portfolio analysis (Boender, 1997; Gülpinar et al., 2004a). Furthermore, some variants of the moment matching method have been proposed to reduce the number of random variables and expedite the algorithm. Høyland et al. (2003) developed a heuristic algorithm for accelerating the original moment matching, but convergence cannot be guaranteed. Rubasheuski et al. (2014) described a method that combines moment matching with a forward selection reduction technique to expedite the tree construction process. Gülpinar et al. (2004b) introduced a hybrid method for price assessment scenario tree generation, which combines the main ideas of clustering and moment matching. In this hybrid method, prices (the scenario nodal values) are obtained from clustering and the scenario probabilities are determined by solving a moment matching problem. However, in the field of streamflow prediction, the probability of a fixed nodal-values scenario is required to be fixed, so this hybrid approach may not be appropriate. This is because the probability of a fixed nodal-values scenario under such a hybrid method might be subject to change, due to the non-uniqueness of the optimization problem, if only scenario probabilities are used as decision variables. Moreover, changes of preference coefficients for different moments also can lead to a large probability change of the fixed nodal-values scenario. Thus, the result of this hybrid method may yield good model parameters but they cannot be regarded as the scenario probabilities. Note that the original moment matching method does not suffer from this drawback, since both the nodal value and probability are used as decision variables.

In addition to those three categories, there are other scenario tree generation methods. Pflug (2001) presented a method that can be used to approximate a scenario tree for a given stochastic process based on minimizing the Wasserstein distance between the original stochastic process and the generated scenario tree. Da Costa et al. (2006) presented a method for producing a parsimonious multivariate scenario tree by using Principal Component Analysis (PCA), but it may encounter computational difficulties when generating a scenario tree with many stages. Heitsch and Römisch (2009a) developed a theory-based heuristic method that generates scenario trees out of an initial set of scenarios, based on forward or backward algorithms for tree generation, including recursive scenario reduction and bundling steps. Pflug

and Pichler (2015a,b) presented a method for the dynamic generation of a scenario tree based on random vectors, which are drawn from conditional distributions given the past and on sample trajectories. These two methods are both capable of generating scenario trees that are good approximations of the historical series. However, the structure of the generated scenario tree for both methods cannot be determined beforehand; it is dynamically adjusted with respect to a distance criterion. Dupacová et al. (2000) examined different scenario tree generation methods and concluded that the choice among these methods should depend on specific problems and the amount of information available.

Since the generated scenario tree is used as the input to a stochastic programming with recourse model, the size of the tree directly impacts the dimensionality of the optimization model (Casey and Sen, 2005). To avoid high computation cost, the number of scenarios (size of the tree) must be reduced properly (Pan et al., 2015; Séguin et al., 2017). The most important issue when conducting scenario tree reduction is the tradeoff between information loss from the full tree and the computational burden (Høyland and Wallace, 2001; King and Wallace, 2012).

Compared with scenario tree generation, scenario tree reduction techniques have received less attention in the literature. In fact, scenario tree generation and scenario tree reduction are often intertwined. There are some common features in scenario tree generation and scenario tree reduction, and some scenario tree generation techniques also can be used for tree reduction. However, directly applying tree generation methods to scenario tree reduction is not always the best choice, since the reduced tree will inherit the drawbacks of the selected tree generation method. For instance, Xu et al. (2015) applied a clustering method to reduce the scenario tree by adopting new pre-specified tree structures with smaller sizes. This is very easy to implement, and the mean of the full tree can be preserved very well. However, the reduced tree cannot preserve any higher moments due to the inherent limitations of the clustering method. Moreover, applying tree generation methods to tree reduction typically involves generating a new tree of smaller size, which cannot be classified as "reduction from a given scenario tree" if the strict definition of our objective of scenario tree reduction is "to determine a subset of the initial scenario tree and assign new probability to the reduced scenarios" (Growe-Kuska et al., 2003).

Most existing scenario tree reduction methods generate the reduced tree by minimizing the probability metric between the reduced tree's distribution and the full tree's distribution, and then calculating the new scenario probability by adding the deleted scenarios' probabilities according to some rules. The goal of these methods can be summarized as follows:

$$\min \quad \text{distance}\left(\text{distribution}_{full\ tree}, \text{distribution}_{reduced\ tree}\right), \quad (1)$$

which is equivalent to minimizing the following norm:

$$\min \sum_{i}^{\infty} ||\text{moment}_{full\ tree}^{ith} - \text{moment}_{reduced\ tree}^{ith}||. \quad (2)$$

Dupacová et al. (2003) first proposed a "backward reduction and forward selection" scenario reduction algorithm for distribution matching. Heitsch and Römisch (2003) then improved the computational efficiency of the algorithm. Subsequently, many studies were carried out focusing on either improving the algorithm or applying it to different fields (Heitsch et al., 2005; Heitsch and Römisch, 2007; Heitsch and Römisch, 2009a; Heitsch and Römisch, 2009b; De Oliveira et al., 2010).

By minimizing the probability metric between the reduced tree's distribution and the full tree's distribution, it is possible to generate a reduced tree with similar distribution as that of the full tree at some loss of information. However, for water resource planning and management, the higher-order moments [i.e. the third-order moment (skewness), the fourth-order moment (kurtosis), etc.] are not as important as the first-order moment (mean) and the second-order moment

(variance), particularly in reservoir management and operations (Yeh, 1985). This is because the mean streamflow is closely related to power generation, and fluctuation around the mean is important to flood control. Therefore, we believe the methods based on matching the distribution may not be the best for tree reduction for reservoir operations. Because of the property of scalarization of vector optimization (Boyd and Vandenberghe, 2004), the pursuit of matching higher-order moments (i.e. the moment^{ith}, $i \geqslant 3$ in Eq. (2)) will be at the expense of matching the first-order and second-order moments. Moreover, methods based on distribution matching are not guaranteed to match lag-one (i.e. serial) and cross-site (i.e. spatial) correlations. These correlations are important in the operation of a cascade reservoir system (Hao and Singh, 2013; Chen et al, 2015).

In our study, we propose a new scenario tree reduction method based on a variant of the Monte Carlo sampling method and regularized optimization. Distinct from all previous studies, our proposed method reduces the scenario tree in a systematic way based on the prior probability and conditional sampling. It is different from the traditional distribution matching and is specifically designed for matching statistical moments that are important to the management and operation of hydro systems. The objective of our proposed method is to minimize the following norm:

$$\min \left(\sum_i^r w_i \cdot ||(Co)\text{moment}_{historical}^{ith} - (Co)\text{moment}_{reduced\ tree}^{ith}|| \right) + \lambda$$

$$\cdot ||\beta_{historical} - \beta_{reduced\ tree}||, \qquad (3)$$

where r indicates the first r_{th} moments selected for matching; w_i is the weighting parameter for the (Co) moment i; λ is a penalty parameter; and β is the scenario probability vector. Comparing Eq. (3) with Eq. (2), it is evident that our method focuses on matching the first r_{th} important moments. It is worth noting that our method does not alter the nodal values of the full scenario tree. This is more compatible with the definition of "scenario tree reduction". Moreover, we formulate a model to optimize a weighted multiple objective function, where we can assess the tradeoff between probability matching and moment matching. We also test and ensure the stability of our method.

We outline this paper as follows: Section 2.1 introduces the neural gas method and emphasizes the probability formula that we use later. Section 2.2 illustrates the developed stepwise conditional Monte Carlo random sampling procedure. Section 2.3 uses the established probability formula to find the "historical probability" for each reduced tree candidate. Section 2.4 introduces five important criteria related to moment matching. Section 2.5 applies regularized optimization to determine posterior probability. Section 2.6 selects the best reduced tree. Section 3 applies the proposed method to the Qingjiang cascade system of reservoirs in China. We provide final remarks and conclusions in Section 4.

2. Methodology

A flowchart of the proposed method is shown in Fig. 1. First, we employ the neural gas clustering algorithm to generate a full scenario tree using historical inflow series. Then, based on the prior probability of each scenario obtained from the neural gas procedure, we apply stepwise conditional Monte Carlo random sampling to generate a tractable number of scenario combinations (subsets of the full tree), serving as the reduced tree candidates for further selection. We calculate the "historical scenario probability" for each reduced tree candidate by reusing the neural gas probability formula. Then we formulate a regularized optimization model, which is a combination of moment matching and modified ridge regression. The model generates the posterior scenario probability for each reduced tree candidate. Lastly, we evaluate all the reduced tree candidates and select the one that performs best under different subjective preferences. Each procedure will be elaborated in the following subsections.

2.1. Neural gas method for scenario tree generation

There are many ways to represent a stochastic process, and a scenario tree is widely used to model streamflow stochasticity. A scenario tree consists of a finite number of outcomes and their corresponding probabilities at each stage. Each scenario is basically a path from the root outcome to the leaves outcome, and its probability is the product of the outcomes' probabilities on that path. Specifically, a streamflow scenario tree is a tree generated from the historical observed streamflow series.

We use the notation $\{S_i\}$, $i \in [1, ...,P]$ to define a streamflow scenario tree in which S_i is the scenario i and P is the total number of scenarios. β_i is defined as the scenario probability of scenario i. We let t be the stage and T be the total number of stages. Then we define a sequence of nodes $N_{i,t}$, $i \in [1, P]$, $t \in [1, T]$, which consists of the scenario S_i from stage 1 (root node) to stage T (leaves node). Note that $N_{i,t} = N_{i*,t}$ if scenario S_i and scenario S_{i*} have a common node $N_{i,t}$ (or $N_{i*,t}$) at stage t, in which case they are interchangeable. Moreover, the nodal value can be a vector instead of a scalar, which means $N_{i,t,j} = (N_{i,t,1}, N_{i,t,2}, N_{i,t,3}, ...,N_{i,t,M})^T$, $j \in [1, M]$, and M is the length of the node vector, which, in the case of a streamflow scenario tree, is the total number of reservoirs in a reservoir system. Fig. 2 is an example of a three-stage streamflow scenario tree with four scenarios, in which $N_{1,2} = N_{2,2}$ and $N_{3,2} = N_{4,2}$ (even though $N_{2,2}$ and $N_{3,2}$ are not shown in Fig. 2).

The neural gas method, a well-known artificial neural network algorithm, generates the representative vectors from the known vector sets (Martinetz and Schulten, 1991; Martinetz et al., 1993). It is a probabilistic generalization of the k-means algorithm in which each vector is assigned a probability of being in each cluster that is proportional to its distance from that cluster's mean. It has been shown that the neural gas method outperforms other clustering methods when generating a streamflow scenario tree for hydro inflows data (Latorre et al., 2007), so we adopt the neural gas algorithm for streamflow scenario tree generation. The neural gas method extracts several representative sequences from historical streamflow series as different scenarios to form a streamflow scenario tree. The steps for performing the neural gas method are detailed in previous studies (Latorre et al., 2007; Melato et al., 2007; Xu et al., 2015). However, we outline some key steps below.

2.1.1. Pre-specified scenario tree structure

The neural gas method requires a pre-specified scenario tree structure to determine a) the number of scenarios in the scenario tree; and b) the nodal structure of the tree. The tree structure can be described by a scenario tree nodal partition matrix. The column vector of a partition matrix represents a scenario. The element value of the partition matrix has no meaning, but if two scenarios share one node, the element values of that node in each scenario should be the same. For example, the partition matrix of the tree structure in Fig. 2 can be [1,1,1,1;2,2,3,3;4,5,6,7]. Because every scenario in the tree shares the common root node in the tree, the first-row elements of the partition matrix must be the same. Similarly, the last-row elements of the partition matrix must be different from each other, since the leaves node cannot be the shared node. Latorre et al. (2007) suggested that the initial tree structure "should be wide enough not to limit how the scenario tree represents the series, ignoring whether the resulting tree will be too large or not".

2.1.2. Nodal value initialization

We initialize the nodal value of each scenario in the scenario tree by randomly selecting a historical streamflow series (i.e., an observed sample vector):

$$N_{i,t} = H_{rand,t} \quad i \in [1, P], \ t \in [1, T], \qquad (4)$$

where $H_{v,t}$ is the streamflow value of historical streamflow series v at
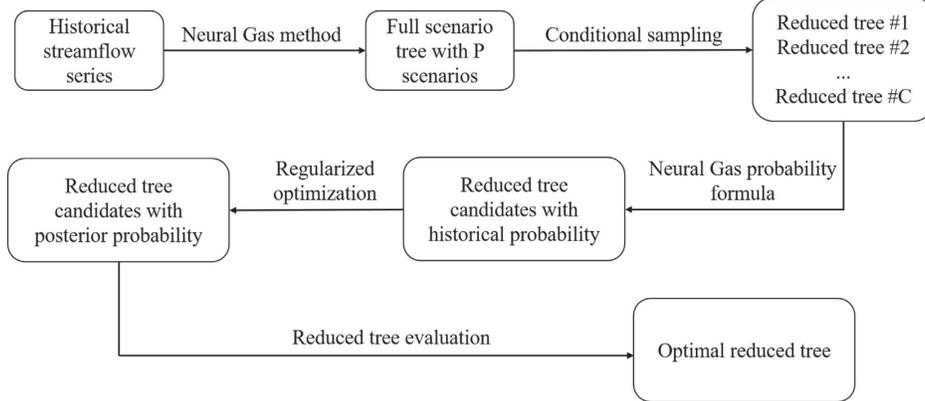
Fig. 1. The flowchart of the proposed method.
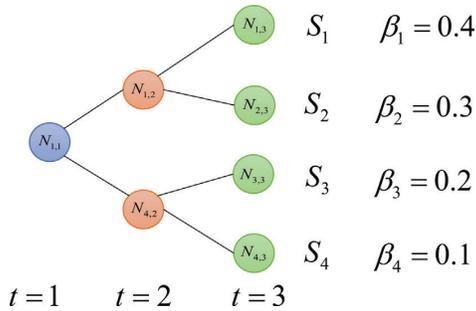
Fig. 2. An example of three-stage streamflow scenario tree structure (t is the stage; S_i is the scenario i; β_i is the probability of scenario i).
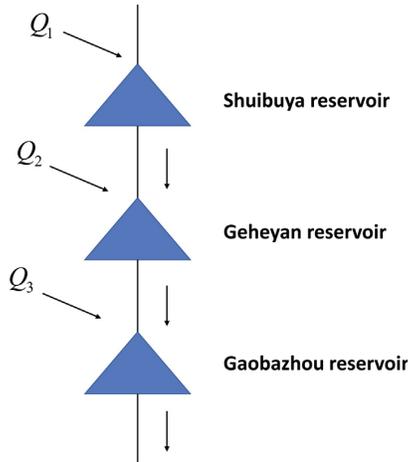
Fig. 3. Schematic diagram of the Qingjiang reservoir system.

stage t, $\upsilon \in [1, K]$; *rand* is a random integer within 1 to K; and K is the total number of historical streamflow series (each series is a set period, in this case a year, and stage t corresponds to month).

Then we average the values of the node shared by two or more scenarios (i.e., the nodes with the same partition matrix elements). This step is for accelerating the convergence:

$$N_{i,t} = \text{Average}\,(N_{i,t}, N_{j,t}, ..., N_{z,t}). \tag{5}$$

2.1.3. Iteration to update nodal values

Altering nodal values to match the historical data series is an important step of the neural gas method, which involves the following operations:

1) Select a series ξ randomly from the historical data series set before each iteration and calculate the Euclidean distance between this series and every initialized scenario in the tree. Then sort the distances in array D in an ascending order and generate an array O to record the distance rank of each scenario:

$$D_{i,\xi} = \sqrt{\sum_{t}^{T} \sum_{m}^{M} (H_{\xi,t,m} - N_{i,t,m})^2}, \qquad i \in [1, P], \tag{6}$$

$$O = order\,(D), \tag{7}$$

where $D_{i,\xi}$ is the Euclidean distance between scenario i and randomly picked series ξ; $H_{\xi,t,m}$ is the nodal value of randomly picked series ξ at stage t for reservoir m; and $N_{i,t,m}$ is the nodal value of scenario S_i at stage t for reservoir m.

2) Define the iteration step size functions $\varepsilon(j)$, $\lambda(j)$, and adaptation function $h(O_i, \lambda(j))$:

$$\varepsilon(j) = \varepsilon_0 \cdot \quad (\varepsilon_f / \varepsilon_0)^{j/j_m}, \tag{8}$$

$$\lambda(j) = \lambda_0 \cdot \quad (\lambda_f / \lambda_0)^{j/j_m}, \tag{9}$$

$$h(O_i, \lambda(j)) = e^{(-O_i/\lambda(j))} \qquad i \in [1, P], \tag{10}$$

where ε_0 and ε_f are step size parameters to reduce from ε_0 to ε_f after each iteration; j represents the iteration time from 0 to the maximum iteration time j_m; λ_0 and λ_f are the adaptation parameters; O_i is the distance rank of scenario S_i; $\varepsilon(j)$, $\lambda(j)$ and $h(O_i, \lambda(j))$ are used to change the nodal values at each iteration, as described in Step 3; $\varepsilon(j)$ is the step size function (a.k.a. "learning rate") that decides the general step size for every scenario's change; and $\lambda(j)$ is another step size function (a.k.a. "adaptation radius") that determines the step size of change for each individual scenario. Both step size functions decrease as the iterations proceed to convergence. As a learning algorithm, the convergence parameters ε_0, ε_f, λ_0, λ_f, and j_m can be determined by fine-tuning for each case. However, in this study, we adopt the values suggested by Latorre et al. (2007) and Xu et al. (2015) as follows: maximum iteration time $j_m = 3000$; step size parameters $\varepsilon_0 = 0.5$ and $\varepsilon_f = 0.05$; and adaptation parameters $\lambda_0 = 10$ and $\lambda_f = 0.01$. The adaptation function, $h(O_i, \lambda(j))$, provides the adaption value for each scenario, based on its distance order to the randomly picked series.

3) Reduce the distance between the historical data series and the

scenarios by changing the node value of $N_{i,t}$ by $\Delta N_{i,t}$, where $\Delta N_{i,t}$ is determined by Eq. (11):

$$\Delta N_{i,t} = \varepsilon(j) \cdot \sum_{i^*=1|N_{i,t} \in S_{i^*}}^{P} h(O_{i*}, \lambda(j)) \cdot \left(H_{\xi,t} - N_{i^*,t}\right) / \sum_{i^*=1|N_{i,t} \in S_{i^*}}^{P} 1, \tag{11}$$

$i \in [1, P], \quad t \in [1, T].$

Note that the notation $i^*=1, 2, ..., P|N_{i,t} \in S_{i^*}$ refers to every index i^* whose corresponding scenario S_{i^*} consists of the node $N_{i,t}$.

Update the nodal value:

$$N_{i,t}^{j+1} = N_{i,t}^{j} + \Delta N_{i,t}. \tag{12}$$

Return to procedure 1) until $j = j_m$.

By iterating on nodal values, the scenario tree is moved toward the selected series. The closer a scenario is to the picked series, the more significant change this scenario will experience. Specifically, Eq. (11) indicates that the change of a given node is determined both by step size and the weighted average distance from every scenario that contains this node to the picked series. The entire scenario tree gradually will move and finally converge to the centroid of the historical streamflow series set. We provide an illustrative figure (Fig. 5) in the case study section. The centroid is a set of P scenarios within the pre-specified tree structure (i.e. a scenario tree with P scenarios) in the higher-dimensional space, with minimum total distances to each historical streamflow series.

2.1.4. Determine the probability of each scenario

The probability of scenario S_i is proportional to the number of historical flows whose closest scenario is S_i. This is described mathematically by Eq. (13):

$$\beta_i = Count\{\zeta' \in [1, K], \quad \zeta'|D_{i,\zeta'} = \min_{i' \in [1,P]}\{D_{i',\zeta'}\}\}/K, i \in [1, P], \tag{13}$$

where β_i is the probability of scenario S_i; $D_{i',\zeta'}$ is the Euclidean distance between scenario $S_{i'}$ and historical series ζ', and Count{} is the counting function. The probability calculated by Eq. (13) is based on the likelihood of occurrence in the history.

2.2. Search for scenario subsets based on conditional Monte Carlo sampling

After applying the neural gas algorithm, we obtain P scenarios with corresponding probabilities in the full scenario tree. The full scenario tree refers to the scenario tree without reduction. By sampling the scenario from the full tree, C scenario combinations (i.e., subsets) are generated, and each combination contains R scenarios ($R < P$). This sampling is based on the full tree scenario probability obtained from the neural gas algorithm.

2.2.1. Define the reduction level, combination size R and total combination number C

Suppose the full tree contains P scenarios and the associated probabilities are denoted by β_i, $i \in [1, P]$. The reduction level for the reduced tree is defined as $(P - R)/P \times 100\%$, where R is the number of scenarios in the reduced tree. A scenario combination is the combination of the sampled R scenarios. Therefore, a scenario combination also can be regarded as a reduced tree candidate with R scenarios.

The number of combinations is represented by the total combination number C. The reason that we specify this number C, instead of going through all possible combinations, is that the number of all possible combinations can be extremely large. For example, if we sample 24 scenarios from a full tree with 48 scenarios, then the number of all possible combinations is $\binom{48}{24} \approx 3.22 \times 10^{13}$, which is intractable. This motivates us to find an efficient sampling strategy to form scenario combinations that are most likely to occur (namely C combinations and $C \ll \binom{P}{R}$).

2.2.2. Sample and formulate scenario combinations

We use the prior scenario probability obtained from the neural gas algorithm to formulate scenario subsets by multi-steps conditional Monte Carlo sampling.

1) Prior probability distribution d_0

Before sampling a scenario from the full scenario tree, the prior probability distribution is known, since the neural gas algorithm calculates the probability of each scenario in the full tree. Let d_0 be a discrete probability distribution that dictates the probability for each scenario in the full tree. The Probability Mass Function (PMF) of d_0 can be written as

$$Pr_{d_0}(S = S_i) = \beta_i, \quad i \in [1, P], \tag{14}$$

where β_i are the prior probabilities obtained from the neural gas algorithm.

2) The first step sample and the first conditional probability distribution d_1

Under the d_0 distribution, we sample one scenario from the full tree, $Sample_1 \sim d_0$, so the probability distribution from which it is selected needs to be updated from d_0 to d_1. Suppose scenario S_u is selected. Then the PMF of d_1 can be written as

$$Pr_{d_1}(S = S_i) = \frac{\beta_i}{1 - \beta_u} = \frac{\beta_i}{\sum_i^{P/\{u\}} \beta_i}, \quad i \in [1, P], \quad i \neq u, \tag{15}$$

where $P/\{u\}$ indicates all scenarios except the selected scenario S_u. This equation basically indicates that when S_u is picked-out, we delete β_u from d_0 and adjust other scenarios' probabilities to form d_1.

3) The second step sample and the second conditional probability distribution d_2

Under the d_1 distribution, we sample the second scenario from the remaining scenarios (i.e., the full tree scenarios with S_u deleted), $Sample_2 \sim d_1$. Suppose scenario S_l is selected. Then the second conditional probability distribution d_2 can be expressed as

$$Pr_{d_2}(S = S_i) = \frac{\beta_i}{1 - \beta_u - \beta_l} = \frac{\beta_i}{\sum_i^{P/\{u,l\}} \beta_i}, \quad i \in [1, P] \quad i \neq u, l. \tag{16}$$

4) The R step sample and R step conditioned probability distribution d_R

Follow the same procedure and keep sampling scenarios from the remaining scenarios until sampling-out the R scenario, under the $R - 1$ conditioned probability distribution d_{R-1}, $Sample_R \sim d_{R-1}$. The R conditional probability distribution d_R can be written as

$$Pr_{d_R}(S = S_i) = \frac{\beta_i}{\sum_i^{P/\{u,l...v\}} \beta_i}, \quad i \in [1, P], \quad i \neq u, l. ..v, \tag{17}$$

where Count$\{u, l. ..v\} = R$, with R the combination size. Now the sampling procedure over R stages for this combination is terminated, resulting in one scenario combination, $Comb_1 = \{S_u, S_l, ..., S_v\}$. Repeat the procedures C times to obtain C combinations.

Back to the original prior probability distribution d_0, we implement the same method to sample R scenarios for one combination. Repeat C times to form C combinations. These scenario combinations serve as candidates for the final reduced tree's scenarios, and new probabilities for each scenario in each combination will be determined.

As demonstrated above, this proposed stepwise sampling method is developed to find a tractable number of scenario combinations that are most likely to occur. Since the sampled combination will serve as the reduced tree candidate, so it is required that the sampled combination

only contains different scenarios. The proposed sampling method ensures that the sampled scenarios are different, and the sampling process is always based on the prior scenario probability distribution. Additionally, the proposed sampling procedure is guaranteed to terminate in R steps.

2.3. The "historical probability" of the sampled scenario for each combination

After sampling R scenarios from the full tree, the original probabilities of the sampled scenarios obtained from the neural gas algorithm no longer can be used. It is necessary to determine the scenario probability for each of scenario combinations (reduced tree candidates).

In this step, we still use the probability formula of the neural gas method to re-compute the probability of each scenario in each reduced tree candidate. In other words, we re-calculate the distance array D in Eq. (6) and then reuse probability Eq. (13) to determine the probabilities of the scenarios for each combination. This probability, calculated by reusing the neural gas probability formula Eq. (13), is named "historical probability", or "β^{hist}". Note that the probability is calculated based on the distance to the historical data series, so it represents the likelihood of occurrence in the history. This β^{hist} should be preserved for each scenario in each reduced tree candidate. However, since we take some scenarios out of the full tree, if this β^{hist} is used as the scenario probability in the reduced tree candidates, the statistical moments of the reduced tree candidates will be different from the full tree. The extent of moment deviation depends on the scenario combination in each reduced tree candidate. Therefore, for the reduced tree candidate k, it is preferable to find another posterior scenario probability "β^{k}", which is sufficiently close to $\beta^{k\ hist}$ and, at the same time, minimizes the deviation of statistical moments between the reduced tree and full tree. We then assign this posterior probability as the scenario probability of the reduced tree k.

2.4. Five criteria to evaluate the statistical moment deviation

We define five criteria to measure the statistical moment deviation: Total Mean Deviation Squared (TMDS), Total Variance Deviation (TVD), Total Lag-one Co-Variance Deviation (TLCVD), Total Cross-site Co-Variance Deviation (TCCVD), and Mean Squared Error (MSE). TMDS, TVD, TLCVD, and TCCVD are used in the optimization step, but MSE serves as a reference. These criteria are explained below.

2.4.1. Total mean deviation Squared (TMDS)
This subsection describes the calculation of the mean values for different scenario trees and the historical data series. The definition of TMDS also is provided.

Mean value of the reduced tree k for reservoir m at stage t:

$$\mu_{m,t}^k = \sum_{i=1}^{R} N_{i,m,t}^k \cdot \beta_i^k. \tag{18}$$

Mean value of the historical data series for reservoir m at stage t:

$$\mu_{m,t}^{historical} = \sum_{i=1}^{K} H_{i,m,t}/K. \tag{19}$$

Mean value of the full tree for reservoir m at stage t:

$$\mu_{m,t}^{fulltree} = \sum_{i=1}^{P} N_{i,m,t}^{fulltree} \cdot \beta_i^{fulltree}, \tag{20}$$

where $\beta_i^{fulltree}$ is the probability of scenario i of the full tree; β_i^k is the probability of scenario i of the reduced tree k; $N_{i,m,t}^k$ is the node value of scenario i in the reduced tree k for reservoir m at stage t; $H_{i,m,t}$ is the value of historical data series i for reservoir m at stage t; $N_{i,m,t}^{fulltree}$ is the node value of scenario i in the full tree for reservoir m at stage t; R is the combination size (i.e., the number of scenarios in a reduced tree); K is the total number of historical data series; P is the number of scenarios in the full tree; C is the total number of reduced tree candidates; M is the length of the node vector (i.e. the total number of reservoirs in a reservoir system); and $m \in [1, M]$, $t \in [1, T]$, $k \in [1, C]$.

Total Mean Deviation Squared (TMDS) of reduced tree k

The term TMDS is used to evaluate the deviation from the reduced tree's mean from the historical series' mean across all reservoirs and stages. A low TMDS value for a reduced tree indicates a good match between a reduced tree's mean and the historical mean, which is desirable. TMDS can be summarized as

$$TMDS_k = \sum_{m=1}^{M} \gamma_m \cdot \sum_{t=1}^{T} (\mu_{m,t}^{historical} - \mu_{m,t}^k)^2, \tag{21}$$

where γ_m is the weight assigned to reservoir m.

2.4.2. Total variance deviation (TVD)
This subsection describes the calculation of the variance values for different scenario trees and the historical data series. The definition of TVD also is provided.

Variance of the reduced tree k for reservoir m at stage t:

$$\sigma_{m,t}^{2\ k} = \sum_{i=1}^{R} (N_{i,m,t}^k - \mu_{m,t}^k)^2 \cdot \beta_i^k. \tag{22}$$

Variance of the historical data series for reservoir m at stage t:

$$\sigma_{m,t}^{2\ historical} = \sum_{i=1}^{K} (H_{i,m,t} - \mu_{m,t}^{historical})^2/(K-1). \tag{23}$$

Variance of the full tree for reservoir m at stage t:

$$\sigma_{m,t}^{2fulltree} = \sum_{i=1}^{P} (N_{i,m,t}^{fulltree} - \mu_{m,t}^{fulltree})^2 \cdot \beta_i^{fulltree}. \tag{24}$$

Total Variance Deviation (TVD) of reduced tree k:

The term TVD represents a criterion that estimates the deviation of the reduced tree's variance from the historical series' variance across all reservoirs and stages. A low TVD value for a reduced tree implies that the reduced tree reproduces the historical variance well and is able to capture the historical extreme values. TVD can be written as

$$TVD_k = \sum_{m=1}^{M} \gamma_m \cdot \sum_{t=1}^{T} |\sigma_{m,t}^{2\ historical} - \sigma_{m,t}^{2\ k}|. \tag{25}$$

2.4.3. Total Lag-one Co-Variance deviation (TLCVD)
The serial correlation between two consecutive stages in the historical streamflow series can be represented by the lag-one covariance.

Lag-one Co-Variance of the reduced tree k for reservoir m at stage t:

$$Cov_{m,t}^k = \sum_{i=1}^{R} \{(N_{i,m,t}^k - \mu_{m,t}^k) \times (N_{i,m,t-1}^k - \mu_{m,t-1}^k)\} \cdot \beta_i^k. \tag{26}$$

Lag-one Co-Variance of the historical data series for reservoir m at stage t:

$$Cov_{m,t}^{historical} = \sum_{i=1}^{K} \{(H_{i,m,t} - \mu_{m,t}^{historical}) \times (H_{i,m,t-1} - \mu_{m,t-1}^{historical})\}/(K-1). \tag{27}$$

Lag-one Co-Variance of the full tree for reservoir m at stage t:

$$Cov_{m,t}^{fulltree} = \sum_{i=1}^{P} \{(N_{i,m,t}^{fulltree} - \mu_{m,t}^{fulltree}) \times (N_{i,m,t-1}^{fulltree} - \mu_{m,t-1}^{fulltree})\} \cdot \beta_i^{fulltree}. \tag{28}$$

Equations (26) and (28) are valid for $t \geqslant 2$. For historical data, we assume $Cov_{m,1}^{historical} = 0$.

Total Lag-one Co-Variance Deviation (TLCVD) of the reduced tree k:

The term TLCVD represents the similarity of the serial correlation between the reduced tree and the historical data series. Generally, the smaller the TLCVD the better, since it means the reduced tree has a similar serial correlation as the historical data series. This criterion can be defined as

$$TLCVD_k = \sum_{m=1}^{M} \gamma_m \cdot \sum_{t=1}^{T} |Cov_{m,t}^{historical} - Cov_{m,t}^{k}|. \tag{29}$$

2.4.4. Total Cross-site Co-Variance deviation (TCCVD)

The spatial correlation between two reservoirs in the historical streamflow series can be represented by the cross-site covariance.

Cross-reservoir Co-Variance of the reduced tree k between reservoir m and n ($m \neq n$) at stage t:

$$Cov_{m,n,t}^{k} = \sum_{i=1}^{R} \{(N_{i,m,t}^{k} - \mu_{m,t}^{k}) \times (N_{i,n,t}^{k} - \mu_{n,t}^{k})\} \cdot \beta_i^{k}. \tag{30}$$

Cross-reservoir Co-Variance of the historical data series between reservoir m and n ($m \neq n$) at stage t:

$$Cov_{m,n,t}^{historical} = \sum_{i=1}^{K} \{(H_{i,m,t} - \mu_{m,t}^{historical}) \times (H_{i,n,t} - \mu_{n,t}^{historical})\}/(K-1). \tag{31}$$

Cross-reservoir Co-Variance of the full tree between reservoir m and n ($m \neq n$) at stage t:

$$Cov_{m,n,t}^{fulltree} = \sum_{i=1}^{P} \{(N_{i,m,t}^{fulltree} - \mu_{m,t}^{fulltree}) \times (N_{i,n,t}^{fulltree} - \mu_{n,t}^{fulltree})\} \cdot \beta_i^{fulltree}. \tag{32}$$

Total Cross-reservoir Co-Variance Deviation (TCCVD) of the reduced tree k:

The term TCCVD represents the similarity of the cross-site correlation between the reduced tree and the historical data series. Like TLCVD, the lower the TCCVD the better, since it indicates the reduced tree has a similar cross-site correlation as the historical data series. This criterion can be defined as

$$TCCVD_k = \sum_{(m,n)}^{\binom{M}{2}} \sum_{t=1}^{T} |Cov_{m,n,t}^{historical} - Cov_{m,n,t}^{k}|. \tag{33}$$

2.4.5. Mean Squared error (MSE)

MSE is a commonly used index to evaluate the Euclidian distance between two data series. Here it is used to describe the distance between the reduced tree and the historical data series. The MSE of the reduced tree k is

$$MSE_k = \sum_{j=1}^{K} \left(\sum_{m=1}^{M} \gamma_m \cdot \sum_{t=1}^{T} \left(H_{j,m,t} - \sum_{i=1}^{R} \beta_i^{k} \cdot N_{i,m,t}^{k} \right)^2 \right)/K. \tag{34}$$

2.5. Regularized optimization for determining posterior scenario probability

As discussed in Section 2.3, it is desirable to keep the smallest deviation of statistical moments between the reduced tree and full tree, while making each scenario probability stable. In this section, we apply the technique of regularized optimization, which combines the modified ridge regression and moment matching.

2.5.1. Ridge regression and the modified assumption

Ridge regression is a widely used biased estimating technique in statistics for multiple linear regression that suffers from multicollinearity (Hoerl and Kennard, 1970). It is similar in form to the ordinary least square (OLS) estimator but adds another regularization term. Thus, a traditional ridge linear regression can be generalized as

the following optimization problem:

$$\underset{\omega_j}{Min}: \sum_{i=1}^{\Omega} \left(y_i - \sum_{j=1}^{\Psi} x_{ij}\omega_j \right)^2 + \lambda \cdot \sum_{j=1}^{\Psi} \omega_j^2, \tag{35}$$

where y_i are dependent variables; x_{ij} are independent variables; ω_j are regression coefficients; λ is the ridge parameter; and Ω and Ψ are the total number of observations and regressors, respectively.

From the Bayesian point of view, the OLS estimator is identical to the maximum likelihood estimator (MLE) under the normality assumption for the error terms. In contrast, the Ridge Linear Regression (Ridge LR) is identical to the maximum a posteriori (MAP) under the normality assumption for the error terms and coefficients terms. This condition can be expressed as

$$y = \sum_{j=1}^{\Psi} \omega_j x_j + \varepsilon = X\omega^T + \varepsilon, \qquad \varepsilon_i \sim N(0, \sigma^2), \ \omega_j \sim N(0, \tau^2), \tag{36}$$

where ε_i is the error term and ω_j is the coefficient term, both of which have a Gaussian distribution with zero mean. Compared with the OLS estimator, the Ridge LR utilizes the prior information that the coefficient term ω has a Gaussian distribution with zero mean. Based on these assumptions, the optimized coefficient ω will be close to zero if a very large value is assumed for the ridge parameter λ.

However, as illustrated in Section 2.3, it is desirable to make the scenario probability as close to β^{hist} as possible. Therefore, instead of assuming all the coefficient terms in Gaussian distributions with zero mean, we assume that the coefficient terms have Gaussian distributions with mean values of β^{hist}; that is, $\varepsilon_i \sim N(0, \sigma^2)$, $\omega_j \sim N(\beta_j^{hist}, \tau^2)$, where β^{hist} is obtained as in Section 2.3.

Based on the assumption and the MAP principle, the modified Ridge LR can be expressed as

$$\underset{\omega_j}{Minimize}: \sum_{i=1}^{\Omega} \left(y_i - \sum_{j=1}^{\Psi} x_{ij}\omega_j \right)^2 + \lambda \cdot \sum_{j=1}^{\Psi} (\omega_j - \beta_j^{hist})^2. \tag{37}$$

2.5.2. Regularized optimization for scenario tree reduction

Since matching the statistical moments between observed data (i.e. historical data series) and the reduced tree is very important, we conduct regularized optimization that combines the modified ridge regression and moment matching. In this study, we consider matching the first order moment (mean), the second order moment (variance), lag-one covariance, and cross-site covariance. Thus, the general form of the optimization model can be expressed as

$$\underset{\beta_i}{Minimize}:$$

$$[w_1 \cdot TMDS + w_2 \cdot TVD + w_3 \cdot TLCVD + w_4 \cdot TCCVD] + \lambda \cdot \sum_{i=1}^{R} (\beta_i - \beta_i^{hist})^2, \tag{38}$$

where w_1, w_2, w_3, and w_4 are the weighting parameters for TMDS, TVD, TLCVD, and TCCVD, respectively; β_i^{hist} is the historical probability of scenario i in a reduced tree candidate; and β_i, the decision variable, is the probability of scenario i in the final reduced tree. The weighting parameter values sum to one and should be specified by decision makers, depending on their preference.

After substitution, the optimization model can be written as

$$
\begin{aligned}
\underset{\beta_i^k}{Minimize} : [& w_1 \cdot \sum_{m=1}^{M} \gamma_m \cdot \sum_{t=1}^{T} (\mu_{m,t}^{historical} - \mu_{m,t}^{k})^2 \\
& + w_2 \cdot \sum_{m=1}^{M} \gamma_m \cdot \sum_{t=1}^{T} \sqrt{(\sigma_{m,t}^{2\ historical} - \sigma_{m,t}^{2\ k})^2} \\
& + w_3 \cdot \sum_{m=1}^{M} \gamma_m \cdot \sum_{t=1}^{T} \sqrt{(Cov_{m,t}^{historical} - Cov_{m,t}^{k})^2} \\
& + w_4 \cdot \binom{M}{2} \sum_{t=1}^{T} \sqrt{(Cov_{m,n,t}^{historical} - Cov_{m,n,t}^{k})^2}\ ] \\
& + \lambda \cdot \sum_{i=1}^{R} (\beta_i^k - \beta_i^{k\ hist})^2,
\end{aligned}
\tag{39}
$$

subject to:

$$
\sum_{i=1}^{R} \beta_i^k = 1 \text{ and } \beta_i^k \geqslant \frac{1}{K}, \quad \forall\ i \in [1, P], \quad k \in [1, C],
\tag{40}
$$

where K is the total number of historical streamflow series and λ is the ridge parameter. The decision variable in the optimization model is the reduced tree probability β_i^k for the reduced tree k. Note that Eq. (39), subject to constraint (40), is a nonlinear and constrained optimization problem. We solve it with the interior point method, via the MATLAB optimization toolbox (Byrd et al., 1999, 2000).

2.5.3. Determine the optimal value of ridge parameter λ

Ridge parameter λ controls the magnitude of preservation of β^{hist}. A larger λ value encourages the solution of β_i to be closer to β_i^{hist}. The value of λ determines the tradeoff between matching the statistical moments and matching the historical probabilities. We use the Ridge Trace method suggested by Hoerl and Kennard (1970) to determine the optimal λ value ($\lambda_{optimal}$):

1) Plot a figure that shows the optimal β_i values as a function of λ. These functions in the plot are considered ridge traces for each β_i;
2) View the ridge traces and pick values of λ for which all β_i values have stabilized. It is common that β_i will fluctuate very widely for small values of λ and then become stable;
3) Choose the smallest λ value after all β_i have approached constants. The smallest value is the optimal λ value, namely, $\lambda_{optimal}$. We provide an illustrative figure (Fig. 6) of this method in the case study section.

2.5.4. Measure total probability deviation by total probability error (TPE)

TPE measures the total probability deviation from the historical scenario probability β^{hist}. It is apparent that TPE decreases as the ridge parameter λ increases. The acceptable tolerance of TPE is subject to the decision maker. The TPE of the reduced tree k is
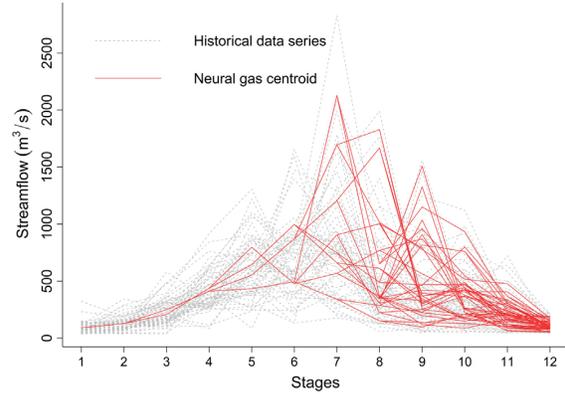
Fig. 5. Historical series value and full tree scenario (i.e. centroids) nodal values of the Shuibuya reservoir.

$$
TPE_k = \sum_{j=1}^{R} |\beta_j^k - \beta_j^{k\ hist}|.
\tag{41}
$$

2.6. Selection of the final reduced tree

Since we have C different reduced tree candidates, we select the reduced tree with the smallest objective function value in Eq. (39) as our final reduced tree, under the specified value of ridge parameter λ.

3. Case study

Our study area is the Qingjiang cascade reservoir system, located in the middle part of the Yangtze River, China. The system produces hydropower to supplement the energy demand in Hubei province. It consists of three reservoirs: Shuibuya (the most upstream reservoir), Geheyan (the middle reservoir), and Gaobazhou (the most downstream reservoir), respectively, as shown in Fig. 3.

In Fig. 3, Q_1 is the streamflow into Shuibuya reservoir; Q_2 is the lateral streamflow between Shuibuya and Geheyan; and Q_3 is the lateral streamflow between Geheyan and Gaobazhou. For this case study, we assume equal weights for all three reservoirs, i.e., $\gamma_1 = \gamma_2 = \gamma_3 = 1$. We have 59 years of monthly historical streamflow data series available, from Jan. 1951 to Dec. 2009. Each set of historical streamflow series consists of 12 stages (i.e. 12 months) and each stage has three
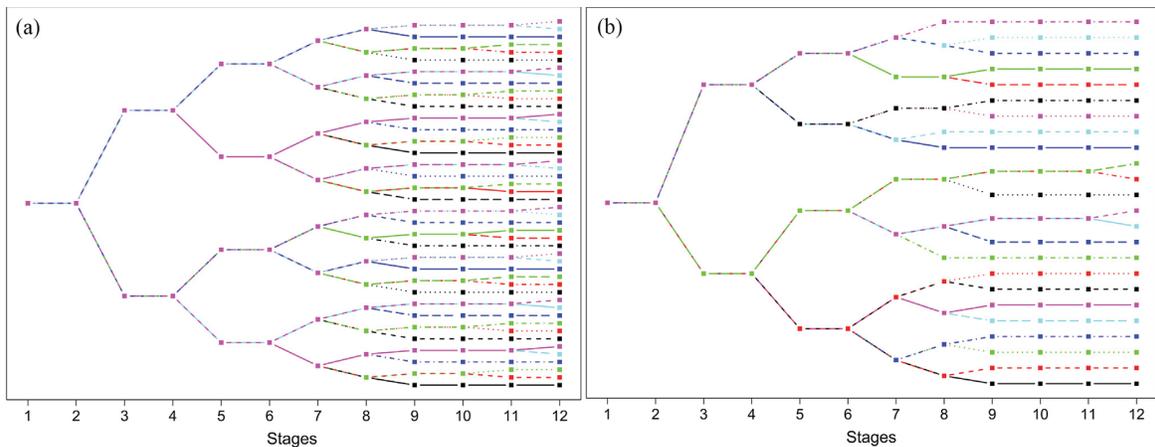
Fig. 4. Tree structures of the full tree (a) and 50% reduced tree structure (b).
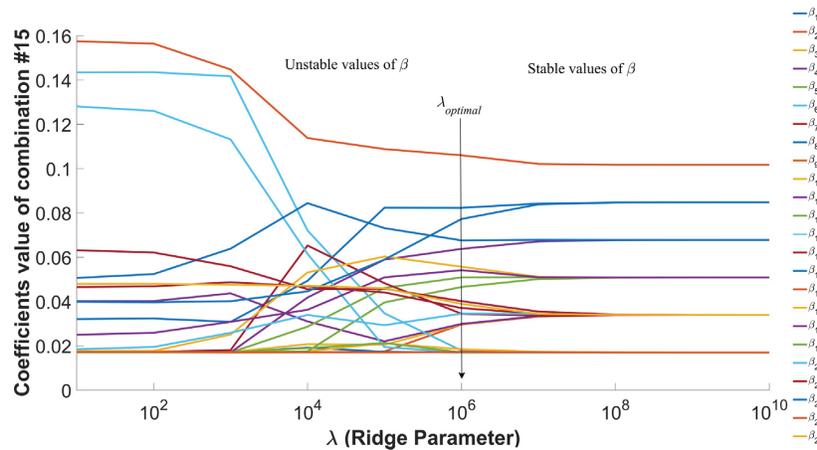
Fig. 6. Ridge Trace of optimization coefficients of reduced tree #15.

streamflow values, which can be vectorized as $[Q_1, Q_2, Q_3]^T$.

3.1. Generation of the full streamflow scenario tree

We use historical data and the neural gas algorithm to generate the full streamflow scenario tree. The step size and adaptation parameter values in Section 2.1.3 are used. The pre-specified structure for the full tree is shown in Fig. 4 (a) and contains 48 scenarios. We also provide Fig. 4(b) to show an example of the reduced scenario tree structure with a 50% reduction level.

The results from the neural gas algorithm are the 48 representative scenarios, including their nodal values and probabilities. Also, these nodal values at different stages can be interpreted as the "centroids" of the historical data series. These results are shown in Fig. 5, using the Shuibuya reservoir as an example.

As computed by the neural gas method, 11 scenarios of the full scenario tree have a zero probability. Because of the limited historical series sets (59) and relatively large number of scenarios in the full tree (48), it is highly possible that for a specific scenario S_i, there is no historical series whose closest scenario is S_i, which leads to zero probability of that scenario. In our analysis, the zero probability scenarios are discarded. Hence, our full tree is a scenario tree with 37 (i.e. $48−11 = 37$) non-zero probability scenarios.

Note that our study focuses on scenario tree reduction for a given a full tree. Therefore, any method can be used to generate the full tree. We use the neural gas algorithm because it has been shown that the generated scenario tree performs best for hydro inflow applications (Latorre et al., 2007).

3.2. Test ridge parameter λ

Ridge parameter λ will impact the optimization problem. Its optimum value should be determined prior to optimization. In this section, we use a 35% reduction level as an example, (i.e. the 37-scenario full tree is reduced to 24 scenarios), and the weighting parameters are assumed to be as follows: $w_1 = 0.98$, $w_2 = 0.02$, and $w_3 = w_4 = 0$ (i.e. mean–variance optimization), with the total combination number $C = 100$ (i.e., the number of reduced tree candidates generated). We first employ the Ridge Trace method to determine the optimal value of ridge parameter λ, then conduct a tradeoff analysis to validate the value of $\lambda_{optimal}$.

3.2.1. Ridge Trace method for determining $\lambda_{optimal}$

Using the specified parameter values, we solve the optimization model. Then we randomly select reduced tree #15 and its associated

scenario probabilities (i.e., optimized coefficients) and plot the ridge trace diagram shown in Fig. 6. The results show that all the coefficient values remain almost constant when $\lambda \geqslant 10^6$. Therefore, we adopt $\lambda_{optimal} = 10^6$.

3.2.2. Validate $\lambda_{optimal}$ value based on a tradeoff between weighted moment deviation and TPE

In Eq. (38), the two competing objectives are the weighted moment deviation ($w_1 \cdot$TMDS + $w_2 \cdot$TVD + $w_3 \cdot$TLCVD + $w_4 \cdot$TCCVD) and TPE. By varying the λ value, a tradeoff between these two objectives can be determined, as shown in Fig. 7. This makes it possible to select a compromise solution. Fig. 7 shows that $\lambda = 10^6$ yields the best compromise solution in that both optimized values are low. This validates the Ridge Trace method as suggested by Hoerl and Kennard (1970).

3.3. The mean–variance preservation of the reduced trees

In this section we demonstrate the mean–variance preservation of the reduced tree using the weighting parameters ($w_1 = 0.98$, $w_2 = 0.02$, and $w_3 = w_4 = 0$), total combination number $C = 100$, optimal ridge parameter $\lambda_{optimal} = 10^6$, and 35% reduction level. Additionally, we test four different reduction levels (10%, 35%, 50% and 70%) to evaluate the effect of the reduction level on the reduced tree's moment matching.

Before the optimization process, the scenario probabilities for each scenario combination (reduced tree candidate) are β^{hist} (i.e., historical probability). As the scenario combinations are generated by conditional Monte Carlo sampling, some scenario combinations may be able to
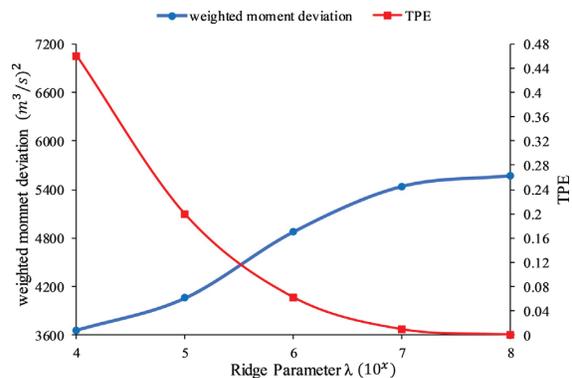

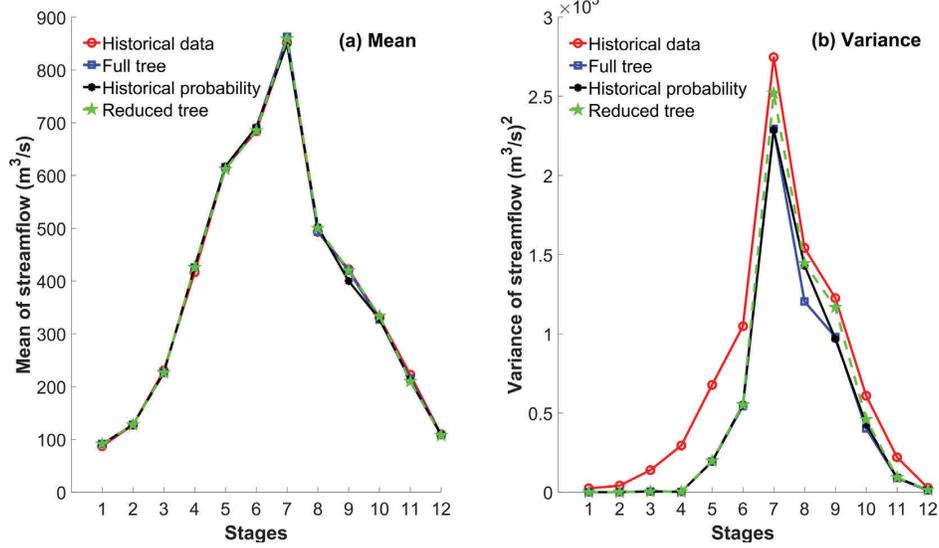**Fig. 7.** Weighted moment deviation & TPE vs. ridge parameter λ

Fig. 8. The mean (a) and variance (b) of different trees for Shuibuya (35% reduction level); ("Historical probability" represents the historical probability tree)

capture the extreme scenarios in the full tree. Extreme scenarios are those scenarios containing extreme nodal values, thus typically with low probabilities. If extreme scenarios are selected in some reduced tree candidates, then those candidates may be good candidates for the final reduced tree.

After the optimization process, the posterior scenario probabilities β_i are calculated. Based on the reduced tree selection rule, the best reduced tree is selected. This best reduced tree is defined as the "reduced tree" in short. The "historical probability tree" is then defined as the reduced tree candidate that has the same scenarios as the "reduced tree". The difference between the historical probability tree and the reduced tree is their scenario probabilities, which are β_i^{hist} and β_i, respectively.

Fig. 8 shows the mean and variance for the historical data series and different scenario trees, using Shuibuya reservoir as the example. Also, the total moment deviations and MSE for each tree are calculated and summarized in Table 1. The results reveal that the mean value of the full tree is preserved because of the nature of the neural gas algorithm, and the mean value of the reduced tree is also very close to the historical data mean due to the optimization. However, the historical probability tree has almost more than two times TMDS than both the full tree and the reduced tree, since it is not optimized. As for the variance, although the variances of both the full tree and reduced tree are not preserved when compared with the historical variance, the variance of the reduced tree deviates less than the full tree, which implies that the reduced tree matches the variance better than the full tree when using historical variance as the reference. This is because the full tree is generated by the neural gas method, which does not preserve higher order moments, while the reduced tree is based on the optimization, where the higher moments are optimized. In addition, the MSE value between the reduced tree and the historical series is very close to that between the full tree and the historical series, which indicates that

the absolute nodal value deviation from the historical series also is maintained after the reduction.

We then test four reduction levels: 10%, 35%, 50% and 70%, based on the full tree with 37 non-zero probability scenarios (i.e., $P = 37$), also using $\lambda_{optimal} = 10^6$ and $C = 100$. Their corresponding combination sizes are: $R = 33$, $R = 24$, $R = 18$, and $R = 11$, and these numbers are also the number of scenarios left in the reduced trees. Generally, the higher the reduction level the larger the information loss. Fig. 9 shows the mean and variance values for trees with different reduction levels, using the Shuibuya reservoir as an example. Note that results obtained for other reservoirs are similar. Also, the total moment deviations and TPE of those trees are summarized in Table 2.

We arrive at several results. First, under $\lambda_{optimal} = 10^6$, the TPE of all reduced trees is acceptable (less than 0.1). Second, the variance deviations of all reduced trees are smaller than the full tree, as indicated by TVD in Table 2.

Third, the trees with 10% and 35% reduction levels perform better than the full tree regarding variance, and their mean deviations are both quite close to the full tree. However, for the 50% and 70% reduction level cases, the means deviate significantly from historical series because of the large information loss, and the variance deviations also increase as the reduction level goes up. Among the four different reduction levels, the 35% reduction level is the most ideal. At this level of reduction, a considerable number of scenarios are eliminated, and the mean and variance are well preserved.

3.4. Preservation of covariances and sensitivity analysis of weighting parameters

In addition to mean–variance optimization, in this section we also account for covariances. Accordingly, we first use weighting coefficients of $w_1 = 0.96$, $w_2 = 0.02$, $w_3 = 0.02$, and $w_4 = 0$, total combination number $C = 100$, optimal ridge parameter $\lambda_{optimal} = 10^6$, and a 35% reduction level to include lag-one covariance in the analysis. Fig. 10 demonstrates the comparisons of mean (a), variance (b) and lag-one covariance (c) for the two test cases.

Compared with the "No covariance optimized" case ($w_1 = 0.98$, $w_2 = 0.02$, $w_3 = w_4 = 0$), the "Lag-one covariance optimized" case ($w_1 = 0.96$, $w_2 = 0.02$, $w_3 = 0.02$, $w_4 = 0$) produces a better match of the lag-one covariance but worsens the match of

Table 1
The total moment deviations and MSE for different trees.

	full tree	35% reduced tree	historical probability tree
TMDS $(m^3/s)^2$	386	452	979
TVD $(m^3/s)^2$	3.08×10^5	2.30×10^5	2.81×10^5
MSE $(m^3/s)^2$	5.33×10^7	5.33×10^7	5.34×10^7
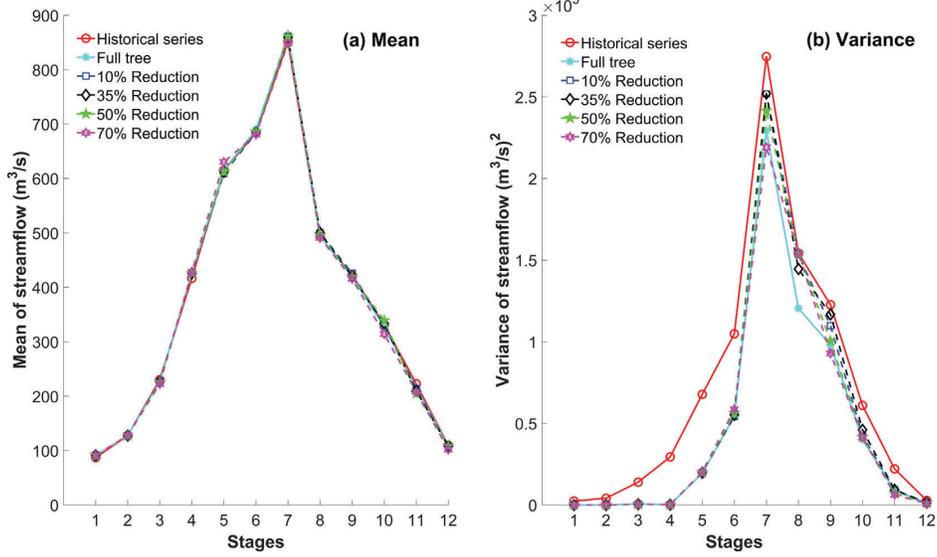
Fig. 9. The mean (a) and variance (b) of trees with different reduction levels

Table 2
The total moment deviations and TPE for different reduced trees.

Reduction level	full tree	10%	35%	50%	70%
TMDS $(m^3/s)^2$	386	418	452	706	1163
TVD $(m^3/s)^2$	3.08×10^5	2.33×10^5	2.30×10^5	2.55×10^5	2.85×10^5
TPE	0.00	0.08	0.08	0.07	0.10

Table 3
Two different test cases, both of 50 experiments.

Type of the tree	Full tree	No covariance optimized	Lag-one covariance optimized
TMDS $(m^3/s)^2$	386	453	402
TVD $(m^3/s)^2$	307,649	229,166	244,788
$w_1 \cdot TMDS + w_2 \cdot TVD (m^3/s)^2$	6523	5027	5281
TLCVD $(m^3/s)^2$	72,497	70,136	53,980
MSE $(m^3/s)^2$	5.3362×10^7	5.3365×10^7	5.3362×10^7
TPE	0	0.076	0.067

variance, as shown in Fig. 10. To avoid the random error, 50 experiments are conducted for both cases and their means are computed. Table 3 and Fig. 10 show the similar results. Table 3 also indicates that the "weighted TMDS and TVD" (defined as $w_1 \cdot$TMDS $+ w_2 \cdot$TVD) of the

Fig. 10. The mean (a), variance (b) and lag-one covariance (c) of trees with and without lag-one covariance optimization
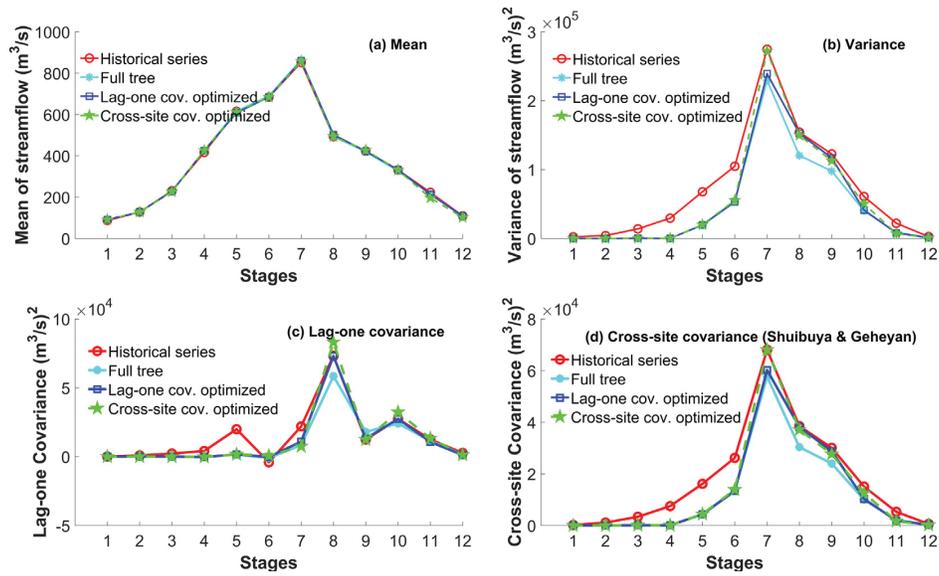
Fig. 11. The mean (a), variance (b), lag-one covariance (c), and cross-site covariance (d) of trees with and without cross-site covariance optimization

"Lag-one covariance optimized" case is greater than that of the "No covariance optimized" case.

We conduct another experiment, named "Cross-site covariance optimized," to include the cross-site covariance in the optimization. We use the following weighting coefficients: $w_1 = 0.94$, $w_2 = 0.02$, $w_3 = 0.02$, and $w_4 = 0.02$ in Eq. (38). A comparison of moment preservations is shown in Fig. 11, for both the "Lag-one covariance optimized" case and the "Cross-site covariance optimized" case. The results show that the former case can better preserve the mean and lag-one covariance. However, the variance and cross-site covariance are better matched in the latter case.

Also, we can see that unlike the mean value, the reduced tree's variance and covariance are not very close to the historical variance and covariance in the first three stages. This is caused by the pre-determined tree structure that has very few nodes in the first few stages. This problem can be corrected by pre-specifying a different tree structure that has more nodes at the first few stages. However, the historical mean value can be preserved very well regardless of the t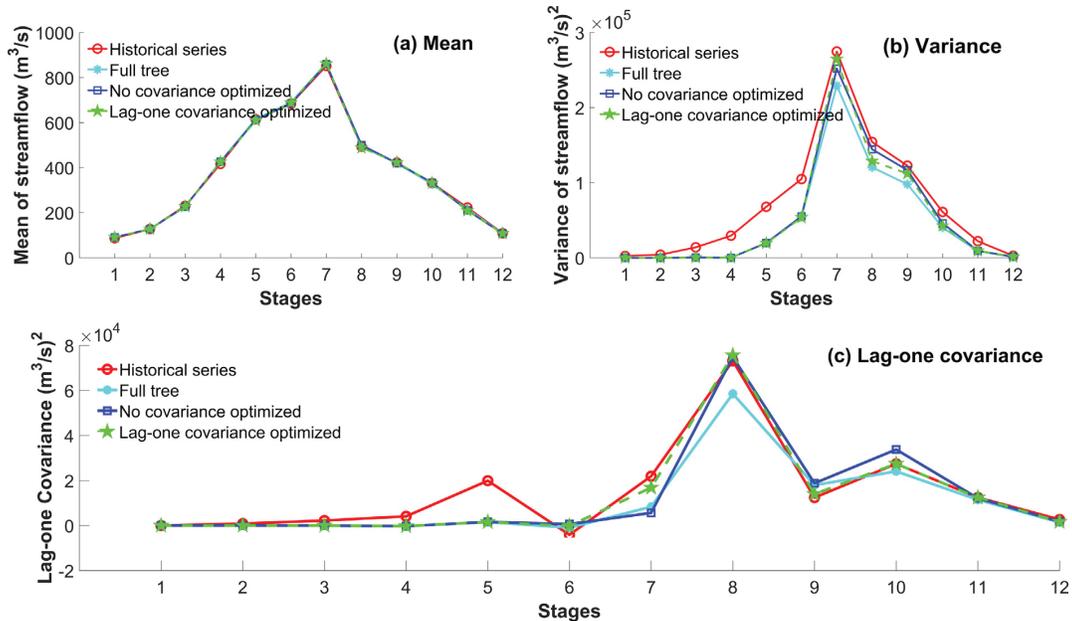ree structure, since the nodal values obtained from the neural gas algorithm are the centroids of the historical series at each stage.

We conduct sensitivity analysis on the reduced tree's weighting coefficients (w_1, w_2, w_3, w_4), under a 35% reduction level. The results of moments matching, based on the reduced trees with seven different sets of weighting coefficients, are listed in Table 4.

As Table 4 shows, the full tree does not preserve the historical second-order moments very well (i.e. variance, lag-one covariance, and cross-site covariance), since it is generated by a clustering method that considers only matching the mean value. With the assigned weighting parameter values for Case 1, the mean and variance are well preserved when compared with the full tree. The lag-one covariance is included in Case 2. Case 3 and Case 4 are designed to find the best weighting parameter values for cross-site covariance. Case 4 has a worse cross-site preservation than Case 3, although Case 4's w_4 value is larger. This is because the w_2 value of Case 4 is smaller than that of Case 3, which indicates the w_2 value may have a higher influence on the optimization of cross-site covariance than the w_4 value. To test this hypothesis, Case 5 and Case 6 are included. Comparing Case 3 with Case 5, we see that under the same w_2 values, cross-site covariance preservation can be improved by simply increasing the w_4 value. Comparing Case 4 with Case 5, the positive effect of w_2 on cross-site covariance preservation can be detected. Case 6 shows that w_1 does not have much effect on covariances. Case 7 tests the range of the w_1 value for obtaining an acceptable TMDS value and indicates that w_1 should be at least 0.9.

The weights represent the preference of the decision maker. However, sensitivity analysis can be used to determine the influence of the weights on the objective function and help select the appropriate values. It is also important to know the difference in magnitude of each moment in choosing weights. In Table 4, the weights are varied systematically, and the corresponding objective function values calculated. The results should help the decision maker to select the appropriate

Table 4
Different test cases, all of 50 experiments.

| Weighting coefficients	Full tree	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7
$w1$	/	0.98	0.96	0.94	0.94	0.94	0.95	0.9
$w2$	/	0.02	0.02	0.02	0.01	0.02	0.01	0.05
$w3$	/	0	0.02	0.02	0.02	0.01	0.02	0.02
$w4$	/	0	0	0.02	0.03	0.03	0.02	0.03
TMDS $(m^3/s)^2$	386	468	365	476	390	642	337	1012
TVD $(m^3/s)^2$	3.08×10^5	2.30×10^5	2.47×10^5	2.33×10^5	2.47×10^5	2.16×10^5	2.56×10^5	1.92×10^5
TLCVD $(m^3/s)^2$	7.25×10^4	6.93×10^4	5.37×10^4	5.77×10^4	5.38×10^4	6.77×10^4	5.09×10^4	7.40×10^4
TCCVD $(m^3/s)^2$	9.98×10^4	7.47×10^4	8.01×10^4	7.55×10^4	8.02×10^4	7.02×10^4	8.31×10^4	6.31×10^4
MSE $(m^3/s)^2$	5.336×10^7	5.337×10^7	5.336×10^7	5.337×10^7	5.336×10^7	5.338×10^7	5.336×10^7	5.340×10^7
TPE	0.000	0.079	0.066	0.083	0.066	0.093	0.060	0.130
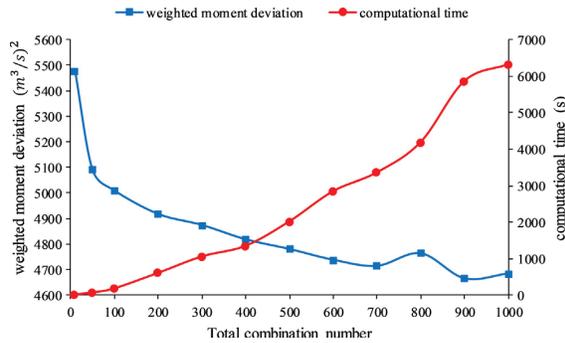
Fig. 12. Stability test of different total combination number

values for the weights.

For the Qingjiang reservoir case study, we suggest the following: for mean–variance preservation, use Case 1 parameter values; for including lag-one covariance, adopt Case 2 parameter values; for also including cross-site covariance preservation, select Case 3 or Case 5 parameter values. Admittedly, for a different case study, the best parameter values may be different.

3.5. Stability test of the total combination number C and computation time report

In the above tests, we conducted all the experiments under total combination number $C = 100$. In this section, we test the stability of the reduction method under various total combination number C and find a number of C at which the solutions of this method converge.

We conduct a numerical experiment to test the stability of the conditional Monte Carlo sampling method by systematically increasing the C value ($C = 10$, 50, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000). We use the following parameter values: $w_1 = 0.98$, $w_2 = 0.02$, and $w_3 = w_4 = 0$; optimal ridge parameter $\lambda_{optimal} = 10^6$; and a 35% reduction level. Fifty numerical experiments are conducted for each C number. The mean of the results for the weighted moment deviation ($w_1 \cdot$TMDS + $w_2 \cdot$TVD + $w_3 \cdot$TLCVD + $w_4 \cdot$TCCVD) is plotted in Fig. 12. Additionally, we report the computational time for the reduction method under each C number.

Fig. 12 reveals that, generally, the weighted moment deviation stops decreasing at $C = 700$. We conduct the significance test showing that after $C = 700$, the weighted moment deviation has no significant linear relationship with the total combination number under the 95% confidence interval. Therefore, at $C = 700$, the proposed method converges, and the small fluctuation after 700 is due to random errors. On the other hand, the computational time increases almost linearly as the total combination number rises. Therefore, if we trade-off between the weighted moment deviation and the computational time, a total combination number $C = 400$ would be considered ideal. This is because at $C = 400$ both weighted moment deviation and computational times are low.

4. Conclusion

In this paper, we presented a new scenario tree reduction method, based on stepwise conditional Monte Carlo sampling and regularized optimization. The proposed method is particularly suited for reducing a streamflow scenario tree that consists of many stages for three reasons. (1) This method does not require updating nodal values at each stage. It is consistent with the definition of scenario tree reduction, i.e., "to determine a subset of the initial scenario tree and assign new probability to the reduced scenarios" (Growe-Kuska et al., 2003). (2) This method stabilizes the reduced tree scenario's probability, and the

physical meaning of the streamflow scenario probability is interpreted easily. (3) This method does not rely on the probability metric but takes advantage of the basic moment matching technique to provide a direct moment matching between the historical data series and the reduced tree.

In the Qingjiang reservoir case study, we determined the optimal ridge parameter $\lambda_{optimal}$ using the ridge trace method and confirmed this value through the tradeoff between the weighted moment deviation and TPE. We tested the moment preservation of the reduced tree with different reduction levels and under "mean–variance" optimization, finding that even with a high level of reduction (such as 35%), the reduced tree still can moment-match well with historical series. We also examined the covariance, showing that it can be preserved well while other moments are not significantly impacted. Also, the stability test showed that the proposed method is stable and approximately converges when the total combination number $C = 700$, which is a small sample size compared with all possible combinations of sampling.

References

Birge, J.R., Louveaux, F., 2011. Introduction to Stochastic Programming. Springer, N. Y., pp. 485.

Boender, G.C.E., 1997. A hybrid simulation/optimisation scenario model for asset/liability management. Eur. J. Oper. Res. 99 (1), 126–135. https://doi.org/10.1016/S0377-2217(96)00387-6.

Boyd, S., Vandenberghe, L., 2004. Convex optimization. Cambridge University Press.

Byrd, R.H., Hribar, M.E., Nocedal, J., 1999. An interior point algorithm for large-scale nonlinear programming. SIAM J. Optim. 9 (4), 877–900.

Byrd, R.H., Gilbert, J.C., Nocedal, J., 2000. A trust region method based on interior point techniques for nonlinear programming. Math. Program. 89 (1), 149–185.

Casey, M.S., Sen, S., 2005. The scenario generation algorithm for multistage stochastic linear programming. Math. Oper. Res. 30 (3), 615–631. https://doi.org/10.1287/moor.1050.0146.

Chen, L., Singh, V.P., Guo, S., Zhou, J., Zhang, J., 2015. Copula-based method for multisite monthly and daily streamflow simulation. J. Hydrol. 528, 369–384.

Da Costa, J. P., De Oliveira, G. C., Legey, L. F. L., 2006. Reduced scenario tree generation for mid-term hydrothermal operation planning. In: Paper presented at 9th International Conference on Probabilistic Methods Applied to Power Systems, PMAPS, 1–7, Stockholm, Sweden. https://doi.org/10.1109/PMAPS.2006.360199.

De Oliveira, W.L., Sagastizábal, C., Penna, D.D.J., MacEira, M.E.P., Damázio, J.M.H., 2010. Optimal scenario tree reduction for stochastic streamflows in power generation planning problems. Optimization Methods Software 25 (6), 917–936. https://doi.org/10.1080/10556780903420135.

Dupacová, J., Consigli, G., Wallace, S.W., 2000. Scenarios for Multistage Stochastic Programs. Ann. Oper. Res. 100 (1), 25–53. https://doi.org/10.1023/A:1019206915174.

Dupacová, J., Growe-Kuska, N., Römisch, W., 2003. Scenario reduction in stochastic programming - an approach using probability metrics. Math. Program. 95 (3), 493–511. https://doi.org/10.1007/s10107-002-0331-0.

Gröwe-Kuska, N., Heitsch, H., Römisch, W., 2003. Scenario Reduction and Scenario Tree Construction for Power Management Problems. Paper presented at Power tech conference, 2003 IEEE Bologna, Bologna, Italy.

Gülpinar, N., Rustem, B., Settergren, R., 2004a. Multistage Stochastic Mean-Variance Portfolio Analysis with Transaction Costs. Innovative Fin. Econ. Networks 3, 46–63.

Gülpinar, N., Rustem, B., Settergren, R., 2004b. Simulation and optimization approaches to scenario tree generation. J. Econ. Dyn. Control 28 (7), 1291–1315. https://doi.org/10.1016/S0165-1889(03)00113-1.

Hansen, P., Jaumard, B., 1997. Cluster analysis and mathematical programming. Math. Program. 79 (1–3), 191–215. https://doi.org/10.1007/BF02614317.

Hao, Z., Singh, V.P., 2013. Modeling multisite streamflow dependence with maximum entropy copula. Water Resour. Res. 49, 7139–7143.

Heitsch, H., Römisch, W., 2005. Generation of multivariate scenario trees to model stochasticity in power management. In: Paper presented at 2005 IEEE Russia Power Tech, Petersburg, Russia. https://doi.org/10.1109/PTC.2005.4524696.

Heitsch, H., Römisch, W., 2003. Scenario Reduction Algorithms in Stochastic Programming. Comput. Optim. Appl. 95 (3), 493–511. https://doi.org/10.1023/A:102180592.

Heitsch, H., Römisch, W., 2007. A note on scenario reduction for two-stage stochastic

programs. Oper. Res. Lett. 35 (6), 731–738. https://doi.org/10.1016/j.orl.2006.12.008.

Heitsch, H., Römisch, W., 2009a. Scenario tree modeling for multistage stochastic programs. Math. Program. 118 (2), 371–406. https://doi.org/10.1007/s10107-007-0197-2.

Heitsch, H., Römisch, W., 2009b. Scenario tree reduction for multistage stochastic programs. CMS 6 (2), 117–133. https://doi.org/10.1007/s10287-008-0087-y.

Hochreiter, R., Pflug, G.C., 2007. Financial scenario generation for stochastic multi-stage decision processes as facility location problems. Ann. Oper. Res. 152 (1), 257–272. https://doi.org/10.1007/s10479-006-0140-6.

Hoerl, A.E., Kennard, R.W., 1970. Ridge Regression: Biased Estimation for Nonorthogonal Problems. Technometrics 12 (1), 55–67. https://doi.org/10.1080/00401706.1970.10488634.

Høyland, K., Kaut, M., Wallace, S.W., 2003. A Heuristic for Moment-Matching Scenario Generation. Comput. Optim. Appl. 24 (2/3), 169–185. https://doi.org/10.1023/A:1021853807313.

Høyland, K., Wallace, S.W., 2001. Generating Scenario Trees for Multistage Decision Problems. Manage. Sci. 47 (2), 295–307. https://doi.org/10.1287/mnsc.47.2.295.9834.

Kaut, M., Wallace, S.W., 2007. Evaluation of scenario-generation methods for stochastic programming. Pacific J. Optim. 3 (2), 257–271. https://doi.org/10.18452/2928.

King, A.J., Wallace, S.W., 2012. Modeling with Stochastic Programming. Springer, N. Y.

Latorre, J.M., Cerisola, S., Ramos, A., 2007. Clustering algorithms for scenario tree generation: application to natural hydro inflows. Eur. J. Oper. Res. 181 (3), 1339–1353. https://doi.org/10.1016/j.ejor.2005.11.045.

Li, H., Luo, L., Wood, E.F., Schaake, J., 2009. The role of initial conditions and forcing uncertainties in seasonal hydrologic forecasting. J. Geophys. Res. Atmos. 114 (4), 1–10. https://doi.org/10.1029/2008JD010969.

Martinetz, T.M., Berkovich, S.G., Schulten, K.J., 1993. Neural-Gas" Network for Vector Quantization and its Application to Time-Series Prediction. IEEE Trans. Neural Networks 4 (4), 558–569. https://doi.org/10.1109/72.238311.

Martinetz, T., Schulten, K., 1991. A "Neural-Gas" Network Learns Topologies. Artificial Neural Networks 397–402.

Melato, M., Hammer, B., Hormann, K., 2007. Neural gas for surface reconstruction. http://www.inf.usi.ch/hormann/papers/Melato.2007.NGF.pdf.

Pan, L., Housh, M., Liu, P., Cai, X., X. C, 2015. Robust stochastic optimization for reservoir operation Limeng. Water Resour. Res. 409–429. https://doi.org/10.1002/2014WR015380.

Pflug, G.C., 2001. Scenario tree generation for multiperiod financial optimization by optimal discretization. Math. Program. 89 (2), 251–271. https://doi.org/10.1007/PL00011398.

Pflug, G. C., Pichler, A., 2015a. Convergence of the Smoothed Empirical Process in Nested Distance, 1–27. http://dx.doi.org/10.18452/8448.

Pflug, G.C., Pichler, A., 2015b. Dynamic generation of scenario trees. Comput. Optim. Appl. 62 (3), 641–668. https://doi.org/10.1007/s10589-015-9758-0.

Rubasheuski, U., Oppen, J., Woodruff, D.L., 2014. Multi-stage scenario generation by the combined moment matching and scenario reduction method. Oper. Res. Lett. 42 (5), 374–377. https://doi.org/10.1016/j.orl.2014.06.006.

Séguin, S., Fleten, S.E., Côté, P., Pichler, A., Audet, C., 2017. Stochastic short-term hydropower planning with inflow scenario trees. Eur. J. Oper. Res. 259 (3), 1156–1168. https://doi.org/10.1016/j.ejor.2016.11.028.

Šutiene, K., Makackas, D., Pranevicius, H., 2010. Multistage K -Means Clustering for Scenario Tree Construction. Informatica, Lith. Acad. Sci. 21 (1), 123–138.

Vitoriano, B., Cerisol, S., Ramos, A., 2000. Generating scenario trees for hydro inflows. In: Paper presented at 6th International Conference on Probabilistic Methods Applied to Power Systems PMAPS (Vol. 2), Madeira Island, Portugal.

Xu, B., Zhong, P.A., Zambon, R.C., Zhao, Y., Yeh, W.W.G., 2015. Scenario tree reduction in stochastic programming with recourse for hydropower operations. Water Resour. Res. 51 (8), 6359–6380. https://doi.org/10.1002/2014WR016828.

Yeh, W.W., 1985. Reservoir Management and Operations Models: A State-of-the-Art Review. Water Resour. Res. 21 (12), 1797–1818. https://doi.org/10.1029/WR021i012p01797.

Zhao, T., Cai, X., Yang, D., 2011. Effect of streamflow forecast uncertainty on real-time reservoir operation. Adv. Water Resour. 34 (4), 495–504. https://doi.org/10.1016/j.advwatres.2011.01.004.

Zhu, F., Zhong, P.-A., Sun, Y., Yeh, W.W.-G., 2017. Real-time optimal flood control decision making and risk propagation under multiple uncertainties. Water Resour. Res. 53, 10635–10654. https://doi.org/10.1002/2017WR021480.

Zhu, F., Zhong, P.A., Sun, Y., 2018. Multi-criteria group decision making under uncertainty: application in reservoir flood control operation. Environ. Modell. Software 100, 236–251.

Chapter 4

A proposed multi-objective, multi-stage stochastic programming with recourse model for reservoir management and operation

Jinshu Li [1]; Wei Zhang [2,3]; William W.-G. Yeh [1*]

[1] Department of Civil and Environmental Engineering, University of California, Los Angeles, California, USA

[2] China Three Gorges Corporation, Beijing, China

[3] Wuhan University, Wuhan, China

Corresponding author: William W.-G. Yeh (williamy@seas.ucla.edu)

Key Points:

- Proposes a general multi-stage stochastic multi-objective optimization framework for reservoir operation

- Develops a linear spline utility function for selecting the most preferred solution on the Pareto front

- Evaluates the performance of the proposed optimization framework with a real-world case study

Abstract

We propose a multi-objective, multi-stage stochastic programming with recourse model for reservoir management and operation, where we use utility theory to select the best compromise solution from the Pareto front. A multi-stage streamflow scenario tree is generated first by the neural gas method. Then the Pareto front at each stage is produced by a modified constrained NSGA-II. A single best compromise solution on the Pareto front must be selected for the

immediate stage and the model moves forward one stage and is re-optimized over a moving planning horizon of fixed duration. The selection is achieved by a proposed linear spline utility function allied with regression. Our proposed utility function has the following advantages: 1) it satisfies the law of diminishing marginal rate of substitution, 2) it does not rely on the pre-specified weight or goal, and 3) it selects the best compromise solution that is likely to fall in the "knee regions" of the Pareto front. We apply the proposed optimization model to the Three Gorges Reservoir (TGR) in China. The two conflicting objectives are 1) maximizing the total expected energy output in the planning horizon, and 2) maximizing the average expected ecological benefits in the planning horizon. The results show that the proposed model produces the optimal water release policy successfully under different hydrological scenarios, when considering both the inflow uncertainty and the tradeoff between the two conflicting objectives.

1. Introduction

In a survey paper, Gutjahr and Pichler (2016) pointed out that stochastic optimization and multi-objective optimization are well established in the field of operations research, but their interaction is less developed. Many real-life decision-making problems frequently involve multiple objectives and stochastically represented uncertainty simultaneously. For example, in reservoir management and operation, the multiple objectives include hydropower generation, water supply and flood control, etc. The uncertainty of reservoir inflows is represented by a stochastic process. When

optimizing the operation of a reservoir system, we simultaneously must consider both multiple objectives as well as the stochastically represented uncertainty of inflows.

Stochastic programming (SP) has been developed to resolve the uncertainty of random variables. The simplest SP problem can be expressed as follows (Shapiro and Philpott, 2007):

$$\underset{x}{\text{maximize}} \ \boldsymbol{E}[f(x,w)]$$
$$s.t. \quad x \in X; w \in \Omega,$$

(1)

where x is the decision variable (vector); w is the uncertain parameter; X is the set of feasible solutions (i.e., solution space); Ω is the set of $\{w_i\}$, $i = 1, 2, ..., I$ (i.e., sample space); $\boldsymbol{E}[\cdot]$ is the expectation operator; and $f(x,w)$ is a general function associated with the uncertain parameter w. There are different ways to represent the uncertain parameter w, and the scenario tree method is one of the popular ways in the literature (Xu et al., 2015a; Séguin et al., 2017; Li et al., 2019). For reservoir management and operation, w is the discretized inflow scenario and Ω is the inflow scenario tree. The methods of generating an inflow scenario tree are detailed in previous studies (Li et al., 2019); most of them require available historical inflow data.

In practice, people are more likely to make sequential decisions, between which some of the uncertain parameters gradually become known (Pflug and Pichler, 2014; Gutjahr and Pichler, 2016). Thus, another practical version of stochastic programming is two-stage stochastic programming. That is, at a given time point (i.e., the immediate stage or the 1st stage) before the realization of the uncertain parameter w, a first decision (i.e., here-and-now decision) must be made. Then, after the realization of uncertain parameter w [i.e., the future stage or the 2nd stage, and $w = (q, J, L, h)$], we make a second decision (i.e., wait-and-see decision) by solving a deterministic optimization problem. The generic form of a two-stage stochastic optimization problem is

$$\underset{x}{\text{maximize}} \quad f(x) + \boldsymbol{E}[g(x,w)]$$

$$\text{s.t.} \qquad x \in X; w \in \Omega, \tag{2}$$

where g, the recourse function, is the solution of the following second stage problem

$$\underset{y}{\text{maximize}} \quad q(w)^T y$$

$$\text{s.t.} \qquad J(w)x + L(w)y \le h(w), \tag{3}$$

where x is the 1st stage decision (here-and-now decision) and y is the 2nd stage decision (wait-and-see decision). Recent studies based on this model include a two-stage stochastic model for supply chain design (Schütz et al., 2009) and a similar model for reservoir operation (Ortiz-Partida, 2019). Koppa et al. (2019) developed a stochastic programming with recourse model based on ensemble forecasts for hydropower optimization.

In addition to the inflow uncertainty, decision makers frequently encounter multiple objectives that conflict. For example, minimizing flood risk and maximizing hydropower output are typically two conflicting objectives. A general form of multi-objective optimization is

$$\underset{x}{\text{maximize}} \quad (f_1(x), f_2(x), ..., f_m(x))$$

$$\text{s.t.} \qquad x \in X, \tag{4}$$

where m is the number of objective functions $(m \ge 2)$, and $(f_1, ..., f_m)$ is the vector of objective functions.

Pareto optimal solutions (Moore, 1907) typically are pursued in multi-objective optimization problems. In general, solution methods can be classified into the following two categories: 1) priori methods aimed at producing one single Pareto optimal solution, and 2) posterior methods aimed at producing the Pareto front, or a subset of all Pareto optimal solutions (Wang et al., 2017). The three most widely used methods in the first category are the scalarizing method (also known as the weighting method), the ε-constraint method, and the goal programming

method. The methods in the second category focus on Pareto front generation. Mathematical programming techniques (such as linear programming and nonlinear programming) and evolutionary algorithms (such as the genetic algorithms) are used to generate the Pareto front. However, when the optimization problem is nonconvex and non-differentiable, mathematical programming techniques may encounter difficulties in finding the true Pareto front.

Evolutionary algorithms (EAs) are used widely for Pareto front generation, because they often can generate a more complete Pareto front (Deb, 2011). Most EAs apply Pareto-based ranking schemes, and the most widely used algorithm is Non-dominated Sorting Genetic Algorithm-II (i.e., NSGA-II), developed by Deb et al. (2002). Other popular multi-objective EAs include SPEA2 (Kim et al., 2004), ε-NSGA-II (Kollat and Reed, 2005), ε -MOEA, AMALGAM (Vrugt and Robinson, 2007), and BORG (Hadka and Reed, 2013). Maier et al. (2014) reviewed the development and application of EAs and other meta-heuristics for the optimization of water resource systems. Reed et al. (2013) summarized popular EAs and concluded that BORG performed best in complex water resources problems. Recent studies on multi-objective models include models for portfolio optimization (Roman et al., 2007) and for water system design and operation (Foued and Sameh, 2001; Reddy and Nagesh Kumar, 2007; Dittmann, et al., 2009; Liu et al., 2011; Wang et al., 2012; Ramos et al., 2014; Zhao and Zhao, 2014; Giuliani et al., 2016a; Giuliani et al., 2016b; Giuliani et al., 2017; Yang et al., 2017; Yu et al., 2017, Tarebari et al., 2018; Wang et al., 2018; Yang et al., 2019).

Stochastic multi-objective programming (SMOP) should consider both uncertainty and multiple objectives. Common ways to incorporate the uncertainty include 1) stochastic constrained methods that assume a distribution for the uncertain parameter, and 2) scenarios-related methods, which use discretized scenarios for the uncertain parameter (Abdelaziz, 2012). Multi-criteria

decision-making (MCDM) methods can be used to select the preferred non-inferior solution from the Pareto front in which the decision makers choose criteria and calculate the priority score or rank for each alternative based on the assumed weight for each criterion (Dhiman and Deb, 2020). Common methods under MCDM include the Simple Additive Weighting (SAW) method (Hwang and Yoon., 1981), which is essentially an application of the scalarization; Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) (Yang et al., 2018), whose principle is similar to goal programming; and Stochastic Multicriteria Acceptability Analysis-2 (SMAA2), which considers the uncertainties in the criteria but requires a good choice of utility function. (Lahdelma and Salminen, 2001; Zhu et al., 2017a).

Recent studies on SMOP have emerged in different fields. Abdelaziz et al. (2007) developed a SMOP for portfolio selection, using the stochastic constrained method and goal programming. Azaron et al. (2008) developed a similar model based on scenarios and goal programming for supply chain design. Fonseca et al. (2010) proposed a stochastic bi-objective for reverse logistics planning, based on discretized scenarios and scalarization. Bath et al. (2010) applied a similar approach in a SMOP for a thermal power generation schedule. Gutjahr and Reiter (2010) set up a bi-objective project portfolio selection model under uncertainty by sampling random scenarios via an adaptive Pareto sampling technique. Hnaien et al. (2010) developed a SMOP for inventory control, using a stochastic constrained method and a scalarization algorithm. Tricoire et al. (2012) developed a bi-objective stochastic model for covering a tour problem based on discretized scenarios and the ε-constraint method. Rath et al. (2016) proposed a similar SMOP model for disaster relief operations. Despite these existing studies across many fields, however, the literature on SMOP in the field of water resources planning and management is sparse. One of the few examples is Zhu et al (2017b), who developed a multi-objective stochastic programming

with recourse model for optimal flood control operation with an emphasis on risk assessments and propagation. They used SMAA-2 to select the best compromise solution.

Distinct from all existing studies in SMOP, this study develops a multi-objective, multi-stage stochastic programming with recourse model for reservoir management and operation. Within this proposed framework, we develop a modified constrained NSGA-II to produce the Pareto front, and propose a linear spline utility function with regression method for selecting the "best compromise" Pareto optimal point.

Since the desired model is not only multi-objective, but also multi-stage stochastic, a unique optimal release policy must be identified for each stage, so that the initial state of the reservoir at the following stage can be determined, and then the rolling horizon technique can be applied. In other words, at each stage, how to select the best compromise Pareto optimal solution among all noninferior alternatives is regarded as the most important task in the proposed model. As discussed, the scalarization method, which essentially evaluates different alternatives by their weighted sum objectives, may provide the simplest solution to this task. However, in the scalarization method, the weight for each objective is typically not easy to determine. Other methods, such as goal programming and ε-constraint, also require pre-specifying parameter values. In addition to this problem, most existing MCDM methods suffer from an even more serious disadvantage. In these methods, the preference coefficient for each objective is assumed to be fixed over the entire domain, which is usually not the case in the real world. For example, a high reservoir water level is usually preferable for power generation but unfavorable for flood control. So, suppose we have two conflicting objectives of maximizing hydropower output (economic objective, F1) and minimizing the reservoir water level (safety objective, F2). When the reservoir water level is low, we would like to increase the water level by 1m, in exchange of 100 kWh power

output. In this case, the preference coefficient of objective F2 over F1 is $100/1=100$. However, when the reservoir water level is high, then we would require a high exchange rate (e.g., a 1m water level increase in exchange for 10000 kWh hydropower output). The preference coefficient of objective F2 over F1 is now $10000/1=10000$. Therefore, we can see that the weight of each objective should change with respect to the objective values. In economics, this phenomenon is called the law of diminishing marginal rate of substitution (Hicks, 1939; Besada and Vázquez, 1999; Dittmer, 2005, White, 2015), which states that consumers are willing to part with less and less quantity of one good in order to get one more additional unit of another good.

The marginal rate of substitution (MRS) is a feature of utility theory. In this theory, utility is the total satisfaction received from consuming a good or service. A utility function is a numerical representation of agent preferences over different objectives (Debreu, 1954; Hanemann, 2006; Zhao et al., 2013). Indifference curves are the level sets of a utility function (Marshall, 2009). The slope of the indifference curves measures the rate at which the agent is willing to substitute one good for another, which is equal to MRS (Baldwin, 1948). MRS is different from Marginal Utility (MU), as MRS is equal to the ratio of the marginal utilities. In general, MU describes the change in utility (happiness) when the amount of one good changes a small amount while the amount of other goods remains constant, which can be utilized to derive optimal operating rules for decision making in hydropower scheduling (Zhao et al., 2015). However, MRS measures the exchange rate of two goods, which can be used to assist in reaching a compromise resolution between two or more conflicting objectives. According to the law of diminishing MRS, the slope of the indifference curve should vary among different regions for an objective. However, it can be shown that scalarization-related methods (including SWE) are equivalent to using a linear utility function, whose slope (i.e., objective preference) is fixed in the entire domain of any objective. The goal

programming related methods (including TOPSIS) limit the representation of the utility function due to the fact that the implied dissatisfaction increases linearly with the deviation away from the goal (Tamiz et al., 1995). In economics, an extensively used utility function is the quadratic utility function (Levy and Markowitz, 1979; Johnstone and Lindley, 2011), which satisfies the law of diminishing MRS. But since the slope of its indifference curve (MRS) changes in every position, the underlying preference for each objective from this utility function is not clear.

A literature review indicates little attention has been paid to developing an appropriate utility function that can be used in tradeoff analysis among different objectives in reservoir operation, even though many decision-making methods, such as SMAA-2, require decision makers to choose a good utility function (Lahdelma and Salminen., 2001). Thus, in our framework, we propose a linear spline utility function method based on utility theory. There are advantages to this method. First, it satisfies the law of diminishing MRS, and the preference coefficients are fixed in each segment with clear interpretations. Second, it does not rely on the specified weight of each objective or any pre-specified parameters, if used along with the proposed regression method. Instead, it finds such information from the Pareto front. Third, the tradeoff solution from this method is likely to be the "knee point" of the Pareto front, which is usually the preferred solution for a decision maker (Branke et al., 2004; Deb and Gupta, 2011).

In sum, in this study we propose a multi-stage stochastic, multi-objective optimization framework that is suitable for reservoir management and operation. Within this proposed framework, we develop a modified constrained NSGA-II to produce the Pareto front, and propose a linear spline utility function with regression method for selecting the "best compromise" Pareto optimal point. We outline this paper as follows: Section 2.1 introduces the streamflow scenario tree and the neural gas method. Section 2.2 presents the general framework of the multi-objective,

46

multi-stage stochastic programming with recourse model, based on the generated scenario tree and the rolling horizon technique. This is followed by an application of the model to reservoir operation. Section 2.3 develops the linear spline utility function method to find the best compromise Pareto optimal point on the Pareto front for each stage, based on the utility theory. Section 3 applies the proposed SMOP model to the Three Gorges Reservoir in China. We then provide final remarks and conclusions in Section 4.

2. Methodology

Fig. 1 shows a flowchart of the proposed methodology.
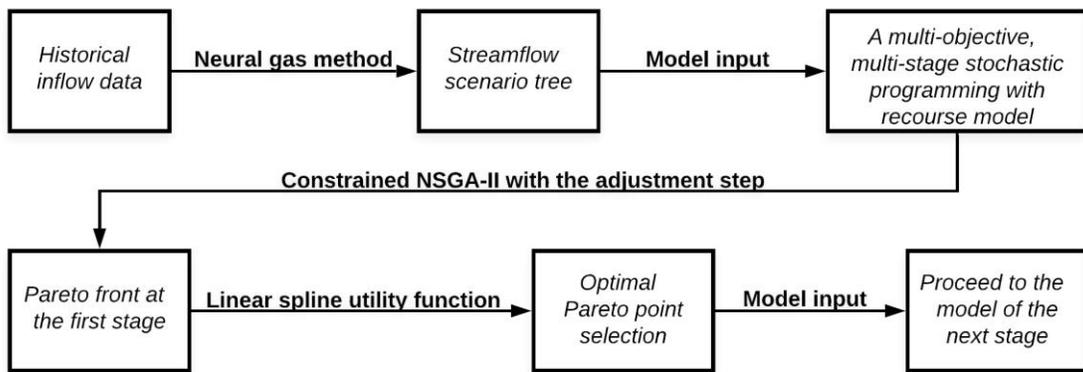
Figure 1. The flowchart of the proposed method

2.1 Streamflow scenario tree and the neural gas algorithm

A scenario tree is used to represent the stochastic process in stochastic programming. The tree is composed of a finite number of outcomes and their corresponding probabilities at each stage. Each scenario is a path from the root outcome to the leaves outcome, and its probability is the product of the outcomes' probabilities on that path. (Li et al., 2019).

In this study, we generate a multi-stage streamflow scenario tree from the historical observed streamflow series, and it is defined as $\{w_i\}$, $i = 1, 2, ..., I$, in which w_i is the scenario i

and *I* is the total number of scenarios. In addition, $P(w_i)$ is defined as the scenario probability of scenario *i*. We designate *t* as the stage and *T* as the total number of stages. Note that $w_{i,t} = w_{i*,t}$ if scenario w_i and scenario w_{i*} have a common node $w_{i,t}$ (or $w_{i*,t}$) at stage *t*. Fig. 2 is an example of a three-stage streamflow scenario tree with four scenarios, in which $w_{1,2} = w_{2,2}$ and $w_{3,2} = w_{4,2}$ (even though $w_{2,2}$ and $w_{3,2}$ are not shown in Fig. 2).
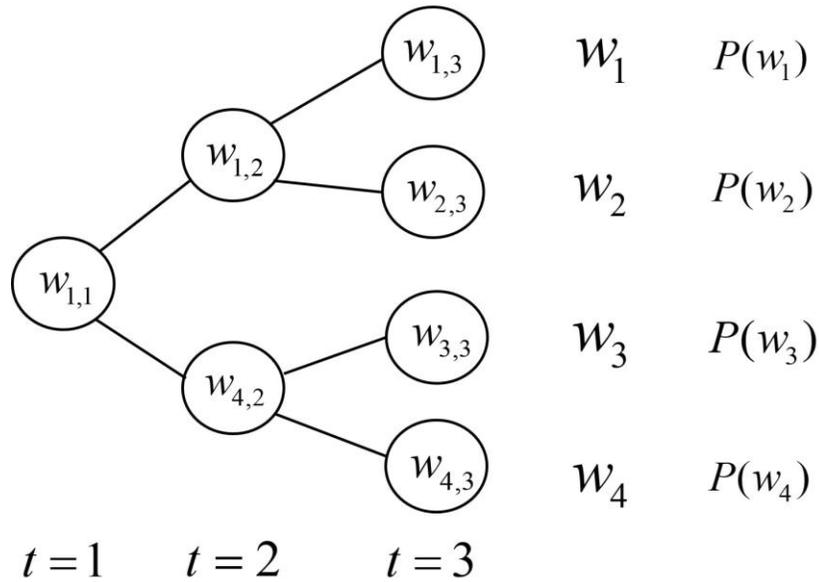
Figure 2. An example of the three-stage streamflow scenario tree structure

The neural gas algorithm is one of the clustering methods that used originally for vector quantization in topology (Martinetz and Schulten, 1991; Martinetz et al., 1993). It requires a fixed pre-specified tree structure and uses a distance-based iterative method to update nodal values to centroids. Latorre et al. (2007) showed that the neural gas algorithm outperforms other clustering methods when generating a streamflow scenario tree for hydro inflows data. Thus, in this study, we employ the neural gas algorithm for streamflow scenario tree generation. We adopt five algorithm parameter values suggested by Xu et al. (2015) and Li et al. (2019) as follows: maximum iteration time $j_m = 20,000$; step size parameters $\varepsilon_0 = 0.5$ and $\varepsilon_f = 0.05$; and adaptation

parameters $\lambda_0 = 10$ and $\lambda_f = 0.01$. The steps for performing the neural gas algorithm are detailed in previous studies (Melato et al., 2007; Li et al., 2019).

2.2 The proposed SMOP model for reservoir operation

2.2.1 The general form

Our proposed optimization model is a multi-objective stochastic optimization model based on a multi-stage scenario tree. It has a general form as follows:

$$
\begin{aligned}
\text{maximize} \quad & \{f_1(x,w),...,f_m(x,w)\} \\
s.t. \quad & x \in X \\
& w \in \Omega,
\end{aligned}
\tag{5}
$$

where x is the decision variable and X is the solution space; w is the scenario and Ω is the set of $\{w_i\}, i = 1,2,...,I$ (i.e., scenario tree); and f_i is the ith objective function.

We further simplify the general form into a two-stage stochastic multi-objective optimization form:

$$
\begin{aligned}
\underset{x}{\text{maximize}} \quad & \{(f_1(x)+\boldsymbol{E}[g_1(x,w)]),...,(f_m(x)+\boldsymbol{E}[g_m(x,w)])\} \\
s.t. \quad & x \in X; w \in \Omega,
\end{aligned}
\tag{6}
$$

where g_i is the ith recourse function. Note that all objectives in Eqn. (6) depend on the decisions in both the 1st stage and 2nd stage. Gutjahr and Pichler (2016) pointed out in their review paper of stochastic multi-objective optimization that most existing studies of stochastic multi-objective optimization assume a bi-objective situation, where one of the two objectives only depends on the

49

1st stage decision (i.e., deterministic objective). Under this assumption, the multi-objective two-stage stochastic optimization can be simplified into a single-objective two-stage stochastic problem. The ε-constraint method, which imposes additional ε-constraints on the deterministic objective in the 1st stage problem, often is used for this simplification (Laumanns et al., 2005). However, the parameter ε is difficult to choose since it represents the tradeoff between two objectives.

The general form above can be transformed into a series of two-stage multi-objective stochastic programming with recourse models. This can be achieved via the idea of rolling horizon. There are two different versions of rolling horizon. The first is to move the entire planning horizon as a moving window (Zhao et al., 2012), while the second is to move only the starting time point and keep the ending time point (boundary) fixed (Yeh, 1985; Xu et al., 2015a; Zhu et al., 2017b). In this study, we adopt the second version. For example, we first construct a two-stage stochastic multi-objective optimization for the planning horizon of [1, T] and obtain its 1st stage decision. Then we move the planning horizon to [2, T]. Based on this new planning horizon and the previously obtained 1st stage decision, we construct another two-stage stochastic multi-objective optimization model and obtain its 1st stage (immediate stage) decision as the recourse action. This procedure is continued until we reach the fixed boundary condition (i.e., the planning horizon is [T-1, T]). Note that, for reservoir operation, the initial storage and final storage conditions typically are fixed.

2.2.2 An application to reservoir operation

We apply the proposed optimization framework to the operation of a multi-purpose reservoir operation system. The two conflicting objectives are 1) maximizing the total expected energy

output in the planning horizon, and 2) maximizing the average expected ecological benefits (represented by the ecological assurance rate) in the planning horizon (Becker and Yeh, 1974; Liu et al., 2011; Zhang et al., 2020).

Assume we have a T-stage scenario tree with a total number of I scenarios. As we mentioned earlier, we want to transfer the original multi-stage multi-objective optimization problem into a series of two-stage stochastic multi-objective models with a rolling horizon. Thus, at each stage t, we construct a two-stage stochastic multi-objective model for the planning horizon $[t, T]$, and obtain its immediate stage decision. We define the model-stage for the planning horizon as stg ($stg = 1,...,t,...,T$). For example, the $stg = 1$ model is built for horizon $[1, T]$ to obtain the decision for stage 1. In this model, the immediate stage is stage 1, and the future stages are $t = 2,...,T$. We then proceed with the $stg = 2$ model, whose planning horizon is $[2, T]$. For the entire planning horizon, we construct T-1 two-stage stochastic multi-objective models (i.e., $stg = 1,...,T-1$ models) and one deterministic multi-objective model (i.e., $stg = T$ model). We illustrate this procedure as follows.

For the $stg = 1$ model, the two objectives f_1 and f_2 can be shown as

$$
\begin{aligned}
\underset{R_t^i, SP_t^i}{\text{maximize}} \quad & f_1 = \mathrm{E}_{t=1} + \sum_{i=1}^{I} P(w_i) \cdot \sum_{t=2}^{T} \mathrm{E}_t^i \\
\underset{R_t^i, SP_t^i}{\text{maximize}} \quad & f_2 = \frac{1}{T}[\Lambda_{t=1} + \sum_{i=1}^{I} P(w_i) \cdot \sum_{t=2}^{T} \Lambda_t^i],
\end{aligned}
\tag{7}
$$

where R_t^i is the power release in the period $[t_{\text{beg}}, t_{\text{end}}]$ for scenario i; SP_t^i is the non-power release in the period $[t_{\text{beg}}, t_{\text{end}}]$ for scenario i; E_t^i is the energy output during the period $[t_{\text{beg}}, t_{\text{end}}]$ for scenario i; t_{beg} and t_{end} are the beginning and the end of stage t, respectively; I is the total number

51

of scenarios and T is total number of stages; $P(w_i)$ is the probability of scenario i; and Λ_t^i is the ecological assurance rate of stage t for scenario i.

The term f_1 is the total energy production objective and f_2 represents the ecological benefit from the reservoir. As ecological benefit has gained more attention in reservoir management and operation, this objective requires a minimum water release to downstream that is essential for natural ecosystems. It conflicts with the energy production objective that encourages maintaining a higher reservoir water level. The ecological assurance rate can be approximated by a piecewise function as follows (Zhang et al., 2020):

$$\Lambda_t = \begin{cases} 0 & \text{if } Q_t < Q_t^{eco,\min} \\ \dfrac{Q_t - Q_t^{eco,\min}}{Q_t^{eco,pro} - Q_t^{eco,\min}} & \text{if } Q_t^{eco,\min} \leq Q_t < Q_t^{eco,pro} \\ 1 & \text{if } Q_t \geq Q_t^{eco,pro} \end{cases}, \tag{8}$$

where $Q_t^{eco,\min}$ is the minimum ecological streamflow for the period $[t_{\text{beg}}, t_{\text{end}}]$; and $Q_t^{eco,pro}$ is the appropriate ecological streamflow for the period $[t_{\text{beg}}, t_{\text{end}}]$. They are determined using the month-by-month frequency calculation method (Chen, 2005): 90% for the dry period (Dec.-Mar.), 70% for the normal period (Apr. Oct. and Nov.), and 50% for the wet period (other months). The term Q_t is the total reservoir discharge in the period $[t_{\text{beg}}, t_{\text{end}}]$.

Other constraints for the $stg = 1$ model include the following:

1. Water balance equations:

$$V_t^i = V_{t-1}^i + (W_t^i - R_t^i - SP_t^i) \cdot \Delta t;$$
$$Q_t^i = R_t^i + SP_t^i$$
$$W_t^i = w_{i,1} = D_1 \qquad \text{if } t = 1;$$
$$W_t^i = w_{i,t} \qquad \text{if } t > 1; \qquad\qquad (9)$$
$$V_0^i = V_{Initial};$$
$$V_T^i \geq V_{Initial};$$
$$\forall t = 1, ..., T;$$
$$\forall i = 1, ..., I,$$

where R_t^i and SP_t^i are defined as before; Δt is the time duration between each stage; V_t^i is the ending reservoir storage of stage t (i.e., the reservoir storage at t_{end}) for scenario i (for example, $V_{October}$ is the reservoir storage at the end of October). $V_{Initial}$ is the initial reservoir storage (i.e., the reservoir storage at the beginning of the first stage); and V_T^i is the ending reservoir storage of the entire planning horizon (i.e., the reservoir storage at T_{end}) for scenario i, which is set to be equal or greater than the initial reservoir storage; W_t^i is the inflow in the time period of $[t_{beg}, t_{end}]$ for scenario i; and D_t is the deterministic forecasted inflow during the period of $[t_{beg}, t_{end}]$. Note that D_t is available only when we have reached the model-stage $stg = t$. For example, for the $stg = 1$ model, D_1 is available. And in this model, we first use D_1 to replace $w_{i,1}$ (i.e. the scenario tree node value at stage 1 for all scenarios $i = 1, ..., I$), then W_1^i equates with D_1. For other stages (i.e., $t > 1$), W_t^i is equal to the original scenario tree node value $w_{i,t}$ (Xu et al., 2015a).

2. Power and energy output equations:

$$
\begin{aligned}
&\mathrm{E}_t^i = N_t^i \cdot \Delta t; \\
&N_t^i = K \cdot R_t^i \cdot H_t^i; \\
&H_t^i = FN_1(\frac{V_{t-1}^i + V_t^i}{2}) - FN_2(R_t^i + SP_t^i); \\
&\forall t = 1,...,T; \\
&\forall i = 1,...,I,
\end{aligned}
\tag{10}
$$

where N_i^t is the power output during the period $[t_{\mathrm{beg}}, t_{\mathrm{end}}]$ for scenario i; H_t^i is the gross average

water head during period $[t_{\mathrm{beg}}, t_{\mathrm{end}}]$ under scenario i, which is the difference between average

forebay water level and tailrace water level; FN_1 and FN_2 are the functions of forebay water level

and tailrace water level (Xu et al., 2015a); and K is the comprehensive output coefficient for the

reservoir (Liu et al., 2011; Yang et al., 2017; Yang et al., 2018).

3. Storages and power output limits:

$$
V_{t,\min} \le V_t^i \le V_{t,\max},
\tag{11}
$$

$$
N_{\min} \le N_t^i \le N_{\max},
\tag{12}
$$

where $V_{t,\min}$ and $V_{t,\max}$ are the lower and upper bounds of the reservoir storage at each stage t, which

include flood control reservation. The terms $N_{\min}$ and $N_{\max}$ are the minimum and maximum limits

of power output. Those are all constants, $\forall t = 1,...,T$ and $\forall i = 1,...,I$.

4. Release limits:

$$
R_{\min} \le R_t^i \le R_{\max},
\tag{13}
$$

$$
SP_{\min} \le SP_t^i \le SP_{\max},
\tag{14}
$$

where $R_{\min}$ and $R_{\max}$ are the lower and upper bounds of the power release; and $SP_{\min}$ and $SP_{\max}$ are the lower and upper bounds of the non-power release, $\forall t = 1,...,T$ and $\forall i = 1,...,I$.

5. Uniqueness of decision variables:

$$R_t^{i_1} = R_t^{i_2}; SP_t^{i_1} = SP_t^{i_2} \quad \text{if } w_{i_1,t} = w_{i_2,t}, \quad \forall i_1, i_2 \in [1, I]. \tag{15}$$

These constraints basically state that decision variables (i.e., R_t^i and SP_t^i) shared by more than one scenario should be the same.

We employ NSGA-II to solve the above optimization problem. In terms of constraint handling, Constraints (9), (10) and (15) are hard equality constraints that can be encoded in the algorithm. Constraints (13) and (14) are the initial ranges of decision variables that can be specified at the beginning. Thus, only the storage and output limit constraints [i.e., Constraints (11) and (12)] will be given specific consideration. The original constrained NSGA-II's approach is to ensure that any feasible solution has a better nondomination rank than any infeasible solution. It works well for problems with a small number of constraints. However, in practice, we find it is not efficient for our proposed model, since our model has many more constraints than an ordinary multi-objective model. For example, if the scenario tree has 50 scenarios with 10 stages, then we will have $50 \times 10 \times 2 \times 2 = 2000$ constraints for Constraints (11) and (12). With such a very large number of constraints, it is very difficult for the original method to generate sufficient feasible offspring, and most of the initialized individuals will be stuck in the infeasible zone.

To overcome this problem, we developed a modified constrained NSGA-II. Specifically, we slightly modify the approach of constraint handling in the original NSGA-II by proposing an additional "adjustment" step to every offspring solution. This "adjustment" step is added before

55

each round of the non-domination sorting step in the NSGA-II. Unlike the original constrained NSGA-II that makes infeasible solutions inferior in non-domination sorting and removes them by tournament selection, our "adjustment" step eliminates the infeasible solutions before sorting by adjusting the decision variable values to satisfy the constraints. To implement this, we need to set up an adjustment strategy for each type of constraint. Fig. 3 shows the adjustment strategies for Constraints (11) and (12) in this model. These strategies also are detailed in Appendix B.

(1)

Chart #1 Starts: An original offspring comes in

$for \ \forall (t, i)$

$V_t^i < V_{t,\min}$?

No → Go to Chart #2

Yes → $SP_t^i (new) = W_t^i - \dfrac{(V_{t,\min} - V_{t-1}^i)}{\Delta t} - R_t^i$

$SP_t^i (new) \geq SP_{\min}$?

No → $SP_t^i = SP_{\min}$
$R_t^i = W_t^i - \dfrac{(V_{t,\min} - V_{t-1}^i)}{\Delta t} - SP_{\min}$

Yes → $SP_t^i = SP_t^i (new)$
$R_t^i = R_t^i$

(2)

Chart #2 Starts

$for \ \forall (t, i)$

$V_t^i > V_{t,\max}$?

No → Go to Chart #3

Yes → $R_t^i (new) = W_t^i - \dfrac{(V_{t,\max} - V_{t-1}^i)}{\Delta t} - SP_t^i$

$R_t^i (new) \leq R_{\max}$?

No → $R_t^i = R_{\max}$
$SP_t^i = W_t^i - \dfrac{(V_{t,\max} - V_{t-1}^i)}{\Delta t} - R_{\max}$

Yes → $R_t^i = R_t^i (new)$
$SP_t^i = SP_t^i$

(3)

Chart #3 Starts

$for \ \forall (t, i)$

$Q_t^i = R_t^i + SP_t^i$

$N_t^i < N_{\min}$?

No → Go to Chart #4

Yes → $R_t^i (new) = \dfrac{N_{\min}}{(K \cdot H_t^i)}$

$R_t^i = R_t^i (new)$
$SP_t^i = Q_t^i - R_t^i (new)$

(4)

Chart #4 Starts

$for \ \forall (t, i)$

$N_t^i > N_{\max}$?

No → Check feasiblity R_t^i, SP_t^i

Yes → $R_t^i (new) = \dfrac{N_{\max}}{(K \cdot H_t^i)}$

$R_t^i = R_t^i (new)$
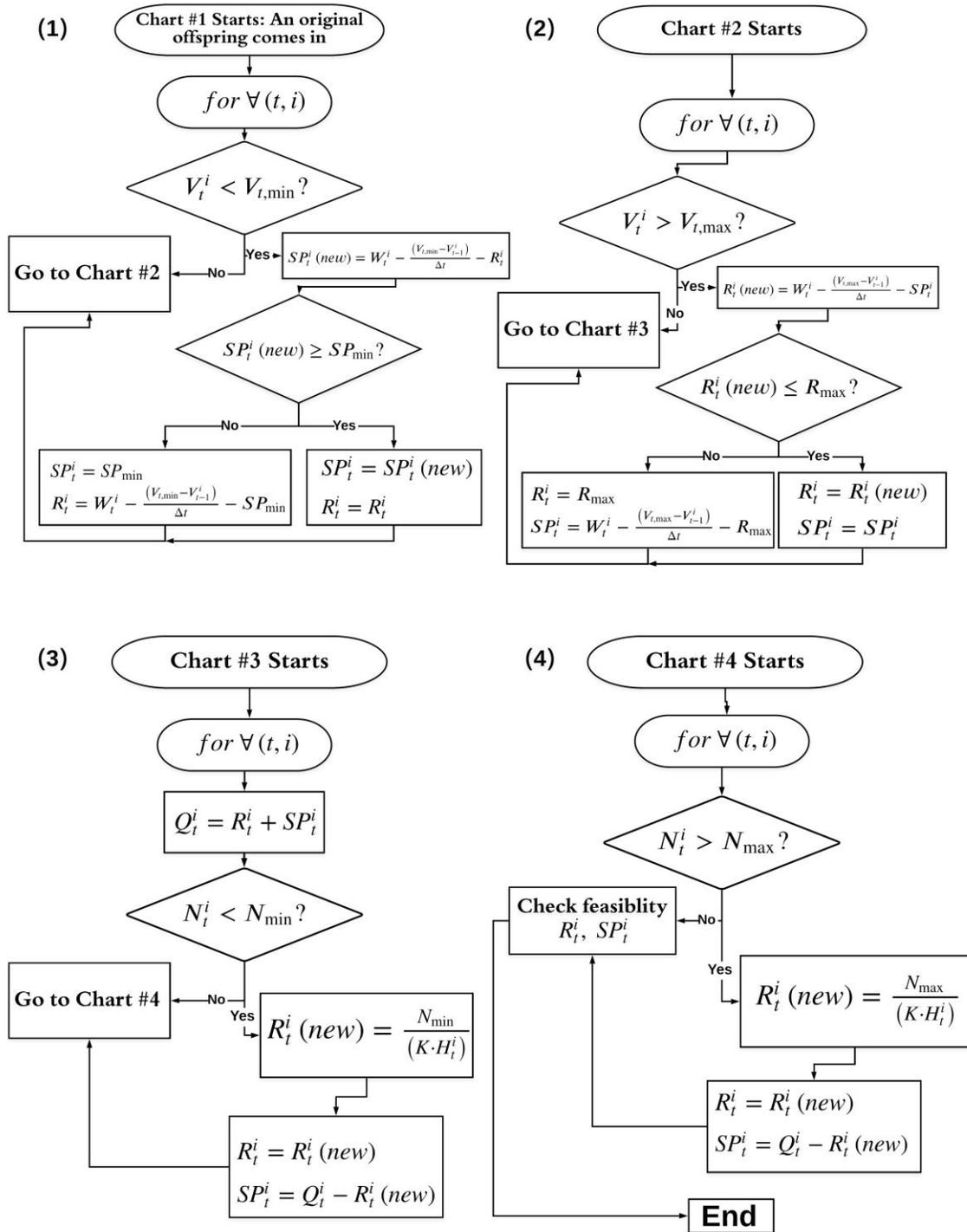$SP_t^i = Q_t^i - R_t^i (new)$

End

Figure 3. Flowchart of the adjustment strategies for Constraints (11) and (12)

Our modified constrained NSGA-II with the adjustment step aims at transforming an infeasible solution to a guaranteed feasible one before each round of non-domination sorting. This method is highly efficient for models that have a large number of upper and lower bounds. For example, Constraint (11) (e.g., $V_{t,\min} \le V_t^i$) incorporates 500 constraints but they are all in the same form. Furthermore, each constraint [e.g., for (t_1, i_1)] only contains two decision variables: $V_{t_1,\min} \le V_{t_1}^{i_1} = V_{t_1-1}^{i_1} + (W_{t_1}^{i_1} - R_{t_1}^{i_1} - SP_{t_1}^{i_1}) \cdot \Delta t$, where only $R_{t_1}^{i_1}$ and $SP_{t_1}^{i_1}$ are involved. This property makes the strategy possible: if ending storage is below the limit, then we first want to reduce non-power release SP_t^i. If SP_t^i is reduced to $SP_{\min}$, then we adjust power release R_t^i to satisfy the constraint. The ideas for other steps are similar. Thus, to formulate the adjustment step, it is necessary to build a strategy for each type of constraint. But once we set it up, it can be used for all constraints of the same form. In general, our proposed adjustment step offers another way to handle constraints in NSGA-II and is most efficient for models with simple constraints.

The modified constrained NSGA-II will provide a Pareto front for the immediate stage. To proceed with our rolling horizon procedure, we need to select one Pareto optimal point on the generated Pareto front for decision making for the immediate stage. How to select this "best compromise" solution will be discussed in section 2.3. For now, we assume that the "best compromise" point has been selected. This point is the solution (or the output) of the $stg = 1$ model (i.e., R_t^i and SP_t^i, $i = 1,...,I$ and $t = 1,...,T$). Since we will update $R_{t \ge 2}^i$ and $SP_{t \ge 2}^i$ when moving forward, we only keep $R_{t=1}^i$ and $SP_{t=1}^i$. Also, since the scenario tree node values at stage 1 are replaced by D_1 (i.e. all scenarios have the identical stage 1 value as D_1), then $R_1^1 = R_1^2 = ... R_1^I = R_1$

and $SP_1^1 = SP_1^2 = ...SP_1^I = SP_1$. These R_1 and SP_1 are the actual power and non-power releases we will adopt for stage 1.

Next, similarly, we proceed into horizon [2, T] and build the $stg = 2$ model. The objective functions are slightly modified as

$$\underset{R_t^i, SP_t^i}{\text{maximize}} \quad f_1 = E_{t=2} + \sum_{i=1}^{I} P(w_i) \cdot \sum_{t=3}^{T} E_t^i$$

$$\underset{R_t^i, SP_t^i}{\text{maximize}} \quad f_2 = \frac{1}{T-1}[\Lambda_{t=2} + \sum_{i=1}^{I} P(w_i) \cdot \sum_{t=3}^{T} \Lambda_t^i],$$

(16)

where the definition of each variable is the same as the $stg = 1$ model. The constraints are identical except that $t = 2,...,T$. The outputs of the $stg = 2$ model are R_2 and SP_2, which are the power and non-power releases to be adopted.

We then construct and solve models for $stg = 3$ to $stg = T$. And we can obtain R_3 and SP_3,..., R_T and SP_T. The last model (i.e., $stg = T$ model) is a deterministic model, since all scenario shares only one node of deterministic forecast value (i.e., D_T). Therefore, to obtain all the release solutions (i.e., $R_1,...,R_T$ and $SP_1,...,SP_T$), we construct T-1 stochastic multi-objective models and one deterministic multi-objective model in a series, with a rolling horizon. Fig. 4 illustrates the recourse procedure, exemplified through a three-stage problem (i.e., T=3). Starting with the $stg = 1$ model, we first update the generated streamflow scenario tree by replacing the t=1 node value $w_{1,1}$ with D_1 (i.e., the deterministic forecasted inflow in the first period). Then, combining the updated scenario tree with the initial state information of the $stg = 1$ model, we solve the proposed SMOP, yielding R_1 and SP_1. With R_1 and SP_1, the initial state information of

59

$stg = 2$ can be identified. We then proceed to the $stg = 2$ model and follow the same procedure
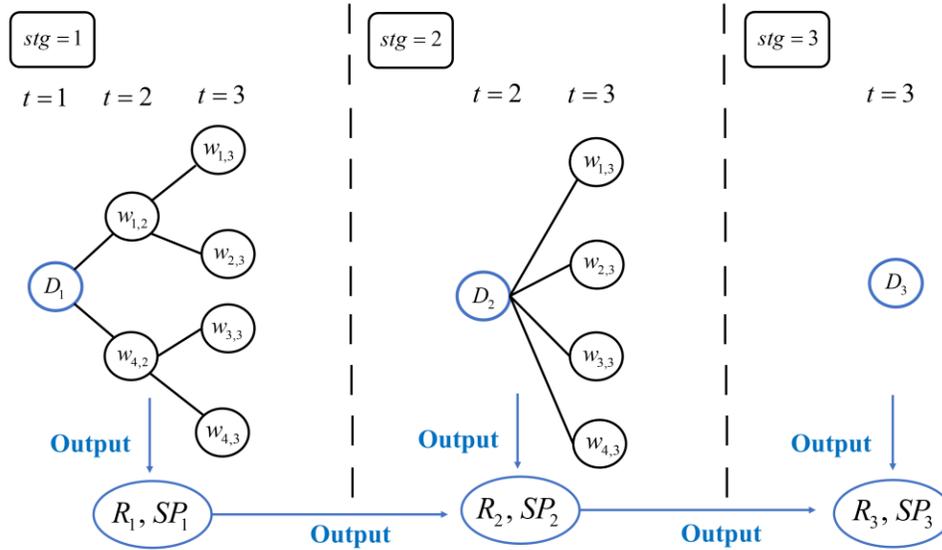
until the end.

Figure 4. An illustrative diagram of an example problem (T=3)

2.3 Linear spline utility function method

We propose a new method based on utility theory to select the "best compromise" solution at each

stage.

Utility is a widely accepted concept in economics that describes the satisfaction (or

happiness) of decisionmakers (Levy and Markowitz, 1979). Thus, how to allocate resources in

order to maximize utility is the central question in the utility maximization problem. The utility

function expresses the utility with each option φ, and is denoted by $U(\varphi)$ (Debreu, 1954;

Hanemann, 2006). A utility function $U(\varphi)$ represents a decisionmaker's preferences if

$$U(\varphi) \geq U(\gamma) \text{ if and only if } \varphi \geq \gamma. \tag{17}$$

Furthermore, a decisionmaker's indifference curve is defined as the set of products that yield a constant level of utility (Marshall, 2009).

$$\text{Indifference Curve} = \{\varphi \mid U(\varphi) = Const.\}. \tag{18}$$

Each indifference curve corresponds to a different level of utility. So, if we want to maximize utility, we need to find the corresponding indifference curve. This idea is the basis of many Pareto optimal point selection methods. For example, the scalarization method selects the "best compromise" point by assigning a weight for each objective. Consider there are two objectives f_1 and f_2. This bi-objective optimization problem can be transformed into a scalar optimization problem as follows:

$$\text{maximize} \quad U(f_1, f_2) = 0.6f_1 + 0.4f_2 \tag{19}$$

where 0.6 and 0.4 are the pre-specified weights; $U(f_1, f_2)$ is the utility function. From the utility theory perspective, the indifference curve can be derived by assuming that utility is a constant, such as

$$U = Const. = 0.6f_1 + 0.4f_2. \tag{20}$$

And it is equivalent to

$$f_1 = -\frac{0.4}{0.6}f_2 + \frac{U}{0.6}. \tag{21}$$

The (f_2, f_1) points on the same indifference curve will provide an identical utility value, and this utility value can be represented by the intercept term of the indifference curve (i.e., $U/0.6$). Since

61

our goal is to maximize utility, the intercept term is to be maximized. Based on the Pareto front

obtained, the intercept term is maximized when Eqn. (21) is tangent with the Pareto front. This is

illustrated in Fig. 5(a). Fig. 5(b) shows another example of maximizing objective f_1 while

minimizing objective f_2. The idea and procedure are identical. That is, maximum utility (i.e.,

maximum intercept) occurs when the indifference curve is moved tangent to the Pareto front.
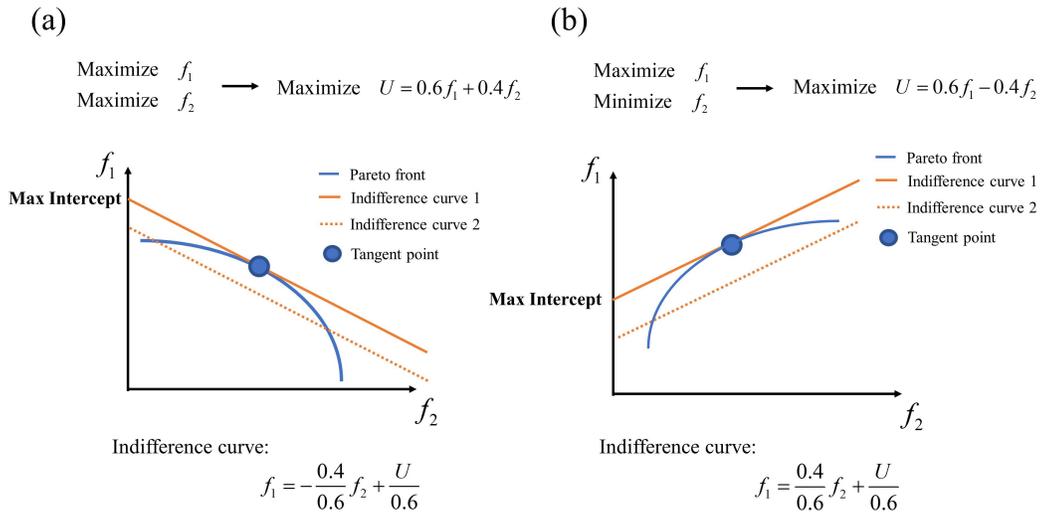

Figure 5. Utility maximization examples: (a) Maximize both f_1 and f_2; (b) Maximize f_1 and minimize

f_2

Although the scalarization method is easy to implement, we see that the underlying

assumption for this method is to use a simple linear utility function for the entire domain. This

assumption is not always valid. To confirm that, we introduce the concept of marginal utility as

the incremental increase in utility that results from consumption of one additional unit, and the

ratio of two marginal utilities of two different objectives is defined as the marginal rate of

substitution (MRS). In the bi-objectives case, MRS is the slope of the indifference curve.

There is an important law in economics, the law of diminishing MRS, which states that as more of one good is consumed, people prefer to give up fewer units of a second good to get additional units of the first good (Hicks, 1939; Besada and Vázquez, 1999). For example, say we want to maximize power generation (i.e., f_1) and minimize flood risk (i.e., f_2). When the flood risk is very low, then we are willing to increase a certain amount of flood risk, in exchange for even a small increase of power generation. This preference can be described by a small MRS of f_2 for f_1, or a mild slope of the indifference curve. In contrast, when flood risk is already very high, then each amount of further increase of flood risk must be able to exchange for a significantly high increase of power generation. This preference can be described by a large MRS, or a steep slope of the indifference curve. Therefore, we see that the gradient (or slope) of the indifference curve should vary within the range of selected objectives.

To consider gradient variation of the indifference curve, we propose a linear spline utility function. Our proposed method uses a linear spline as the utility function. The gradients of linear spline vary in the different regions (i.e., segments) of the selected objectives. The procedure of this method is explained in the following steps:

Step 1: Select one objective (e.g., f_1) as the reference objective (f_{ref}). This objective is preferably a non-safety-concerned objective (e.g., an economic objective), such as maximizing power generation. It acts as a compromise indicator. The tradeoff relationship among other non-reference objectives can be evaluated by comparing the magnitude of their preference coefficients with the reference objective. This reference objective will serve as the y-axis (in a bi-objective problem), or the z-axis (in a three-objective problem) of the Pareto front. We name other objectives as non-reference objectives.

63

Step 2: Segment the domain of non-reference objectives with knots. The knot positions can be determined by expert preference or simply using percentiles of the domain. Then construct the linear spline utility function by assigning each segment a different preference coefficient (or gradient). The general form of the linear spline utility function is as follows:

$$f_{ref} = \beta_0 + \sum_{k_1=1}^{d_1} \beta_{k_1} (f_1 - \alpha_{k_1})_+ + \sum_{k_2=1}^{d_2} \beta_{k_2} (f_2 - \alpha_{k_2})_+ + ... + \sum_{k_n=1}^{d_n} \beta_{k_n} (f_n - \alpha_{k_n})_+, \qquad (22)$$

where $(f_1 - \alpha_{k_1})_+ = \max(0, f_1 - \alpha_{k_1})$; there are $m = n+1$ objectives with f_{ref} as the reference objective; $k_1, ..., k_n$ are the segments for different non-reference objectives $(f_1, ..., f_n)$; β_0 is equivalent to the utility, so maximizing the utility is equivalent to maximizing β_0; $d_1, ..., d_n$ are the total number of knots for each non-reference objective in the Pareto front plot; $\alpha_{k_1}, ..., \alpha_{k_n}$ are the knots for each non-reference objective; and $\beta_{k_1}, ..., \beta_{k_n}$ are the increments of preference in each segment of each non-reference objective. Fig. 6 shows a bi-objective ($m = 2$) example.
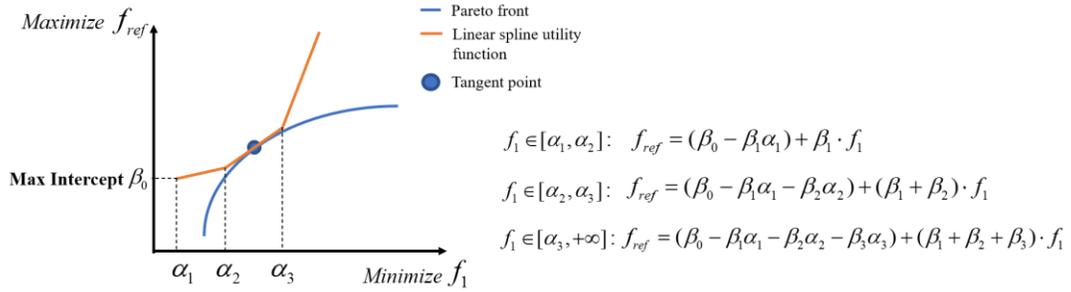
Figure 6. A bi-objective example of the linear spline utility function method

The value of the increments of preference $\beta_{k_1}, ..., \beta_{k_n}$ (i.e., equivalent to the preference coefficients) can be determined by expert preference and experience. However, we further develop a "regression method" to automatically determine $\beta_{k_1}, ..., \beta_{k_n}$ without consulting expert experience. In a bi-objective example, for each segment k_1, we conduct a linear regression by using Pareto optimal points {e.g. $(f_1^{k_1}[i], f_{ref}^{k_1}[i])$} and obtain the linear regression coefficient $\hat{b}_{k_1}$ for each segment k_1:

$$\hat{f}_{ref}^{k_1} = \hat{a}_{k_1} + \hat{b}_{k_1} f_1^{k_1} \quad \forall\, k_1 \in [1, d_1]. \tag{23}$$

If $\hat{b}_{k_1}^{1} > \hat{b}_{k_1}^{d_1}$, then sort the linear regression coefficients from low to high:

$$\hat{b}_{k_1}^{d_1} = \hat{b}_{(1)} \leq \hat{b}_{(2)} \leq ... \leq \hat{b}_{(d_1)} = \hat{b}_{k_1}^{1}. \tag{24}$$

If $\hat{b}_{k_1}^{1} < \hat{b}_{k_1}^{d_1}$, then sort the linear regression coefficients from high to low. The idea is to ensure that the sorted regression coefficients (i.e., $\hat{b}_{(1)}, \hat{b}_{(2)}, ..., \hat{b}_{(d_1)}$) are always in the reverse magnitude order of all the original linear regression coefficients $\hat{b}_{k_1}$ (for the convex Pareto front case), or at least $\hat{b}_{k_1}^{1}$ and $\hat{b}_{k_1}^{d_1}$ (for the non-convex Pareto front case). These sorted regression coefficients (i.e., $\hat{b}_{(1)}, \hat{b}_{(2)}, ..., \hat{b}_{(d_1)}$) will be assigned as the preference coefficient in each segment, which ensures satisfaction of the law of diminishing MRS for the proposed utility function. Fig. 7 provides an illustrative diagram using a bi-objective problem as an example.
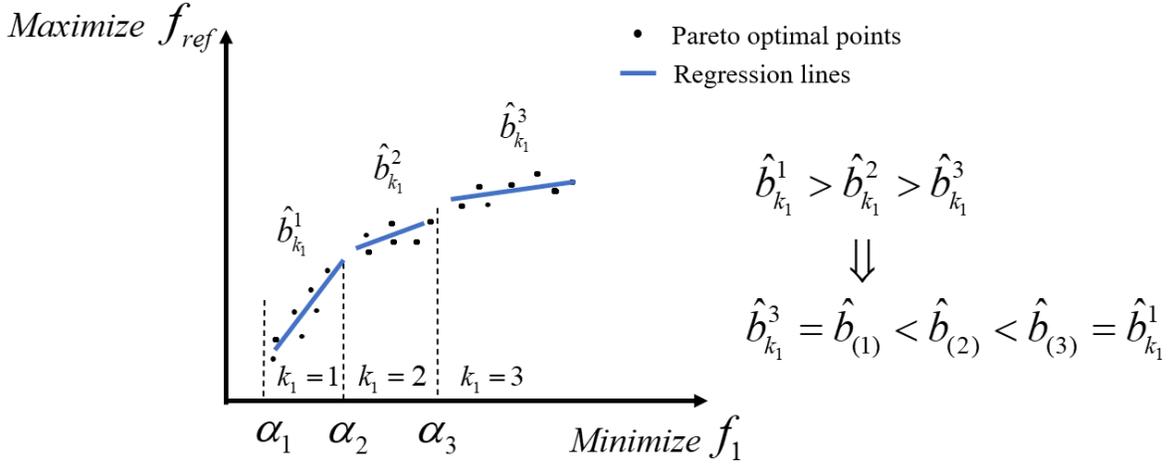
Figure 7. Diagram of the regression method

Lastly, assign those sorted regression coefficients as the preference coefficients for the segments in order, and calculate the increments of preference β_{k_1}, using the bi-objective example in Fig. 6 as an example:

$$\beta_1 = \hat{b}_{(1)}; \quad (\beta_1 + \beta_2) = \hat{b}_{(2)}; \quad (\beta_1 + \beta_2 + \beta_3) = \hat{b}_{(3)}. \tag{25}$$

For a three-objective problem, the procedure is the same. But we need to perform a multiple linear regression:

$$\hat{f}_{ref}^{k_1,k_2} = \hat{a} + \hat{b}_{k_1} f_1^{k_1} + \hat{b}_{k_2} f_2^{k_2} \quad \forall\, k_1 \in [1,d_1], k_2 \in [1,d_2]. \tag{26}$$

Then sort each partial regression coefficient $\hat{b}_{k_1}$ and $\hat{b}_{k_2}$. The preference coefficient increments β_{k_1} and β_{k_2} then can be deduced accordingly.

This regression method does not require any information from expert experience. Instead, it assumes that the preference is related to a linear relationship of the two objectives in the Pareto

front. It also satisfies the law of diminishing MRS. In addition, under the regression method, the tangent point (i.e., the "best" point) is most likely to occur in the "knee region" of the Pareto front. A knee point is characterized as the farthest solution from the extreme line/plane defined by the extreme solutions, as Fig. 8 shows (Bechikh et al., 2010; Bechikh et al., 2011). It is usually regarded as one of the "best compromise" points on a Pareto front (Branke et al., 2004; Deb and Gupta, 2011). For example, in Fig. 8, the knee point is at the edge of a significant f_{ref} decrease while keeping the minimum f_1 value, which is the most preferable compromise. The "knee region" consists of multiple points that are close to the knee point. We can assume these points have similar advantages as the knee point.
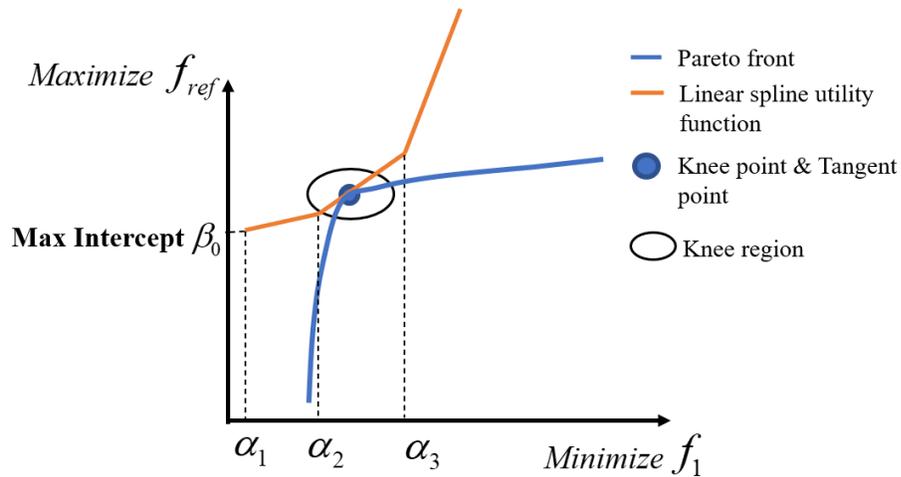
Figure 8. The knee point on a Pareto front

If the Pareto front can be assumed as convex, the knee point is typically unique. Then under the regression method, the linear spline utility function is constructed in such a way that it has large slope differences with the Pareto front on both sides of the knee region. So, the tangent point, a point required to have the same first derivative on both curves, only can appear in the knee region. From another perspective, the knee point also is defined as the point with the largest curvature K_f

(Satopaa, et al., 2011). It can be simplified further to be the point with a very large second derivative f'' [$K_f = |f''| / (1 + f'^2)^{1.5}$]. A large second derivative indicates that the slope in the knee region is changing rapidly. So, the regression slope of this region is far from any extreme slopes that appear on both sides of this region. This enables some points in the knee region to be tangent with the linear spline utility function. In the case where the Pareto front is non-convex, the knee points and knee regions are not unique. The tangent point may appear in any knee region, depending on the different knot pre-settings. But the advantage of the method remains.

Of course, if expert experience is available, then using the empirical preference coefficients is easier and more preferred. We provide both a three-objective example and a bi-objective example with their spline utility functions in Fig. 9.
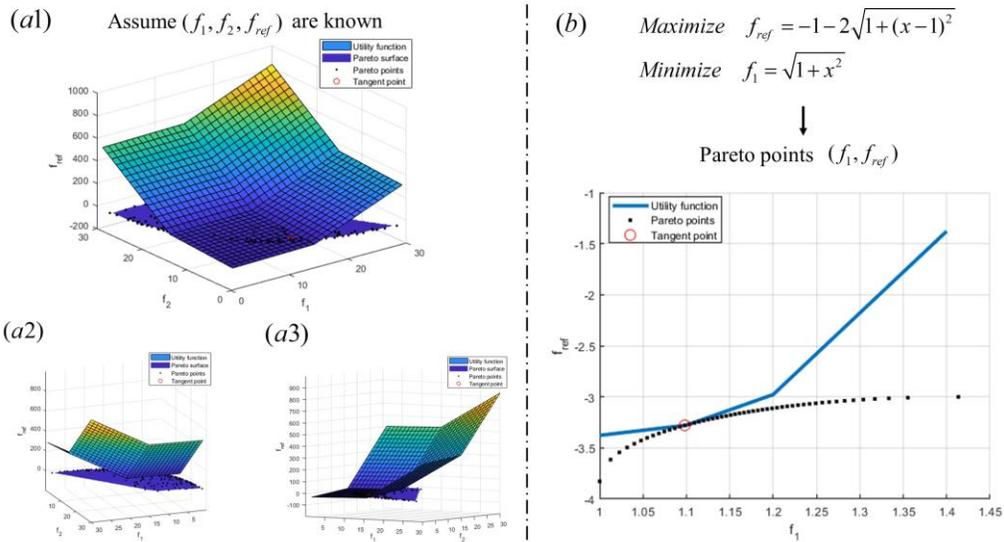

Figure 9. (a) A three-objective example with different views (*a*1, *a*2, and *a*3); (b) A bi-objective example

Step 3: Maximize the utility β_0 by finding the tangent point between the linear spline utility function and the Pareto front. The generated Pareto front is composed of many Pareto optimal points. Denote the Pareto optimal points as $(f_1[i], f_{ref}[i])$. We develop a linear programming problem to find the tangent point, demonstrated by using Fig. 6 as an example:

$$
\begin{aligned}
&\text{minimize} \quad \beta_0 \\
&s.t. \\
&(\beta_0 - \beta_1\alpha_1) + \beta_1 f_1[i] \geq f_{ref}[i] & \forall f_1[i] \in [\alpha_1, \alpha_2] \\
&(\beta_0 - \beta_1\alpha_1 - \beta_2\alpha_2) + (\beta_1 + \beta_2) f_1[i] \geq f_{ref}[i] & \forall f_1[i] \in [\alpha_2, \alpha_3] \\
&(\beta_0 - \beta_1\alpha_1 - \beta_2\alpha_2 - \beta_3\alpha_3) + (\beta_1 + \beta_2 + \beta_3) f_1[i] \geq f_{ref}[i] & \forall f_1[i] \in [\alpha_3, +\infty] \\
&(f_1[i], f_{ref}[i]) \in \text{Pareto front,}
\end{aligned}
\tag{27}
$$

where β_0 is the decision variable (also the utility), and $\beta_1, \beta_2, \beta_3$ are known. Once the optimal β_0 is found, the tangent indifference curve is determined, and the tangent point is identified.

Compared with the scalarization method, our proposed linear spline utility function method offers key advantages. First, it allows decision makers to specify different preference coefficients among the range of objectives. Second, it satisfies the law of diminishing MRS, which is more likely to be the case in reality. Also, unlike other popular utility functions such as the quadratic utility function, whose slope keeps changing, our proposed linear spline utility function keeps the preference coefficient (i.e., the slope of utility function) fixed in each segment. This is often preferable because the compromise relationship between each objective is much clearer. Third, it does not require expert input on preference coefficients if the regression method is used.

3. Case study

Our case study is the Three Gorges Reservoir (TGR), located in the Yichang City of Hubei Province in China. The TGR project has produced many benefits, including flood control, hydropower generation, navigation, and tourism. In general, flood control and hydropower generation are the two main purposes of the TGR. For power generation, the TGR is the largest hydropower station in the world, with an installed capacity of 22,500 MW (32 sets of 700 MW and 2 sets of 50 MW generators).

The storage capacity of the TGR is $3.93 \times 10^{10} \ m^3$. The normal pool level and flood limited water level of the TGR are 175 m and 145 m, respectively. For each year, starting from October (the end of flood season), the conventional TGR operating rule is first to gradually refill the reservoir to the normal water level (175m). Then the reservoir begins gradually to empty its water storage until reaching the flood limited water level (145m) by the beginning of the flood season (around the end of June). During the flood season (June-Sept.), the reservoir water level is required to be maintained at the flood limited water level (145m) (Liu et al., 2015; Zhang et al., 2020).

In this study, we are interested in finding the optimal release of the TGR under the proposed stochastic multi-objective reservoir operation model. The two objectives are maximizing the total expected hydropower output (f_{ref}) and maximizing the average expected ecological benefits (f_1). The planning horizon of the monthly model is one year and has 12 stages, where each month is a stage. We first generate an inflow scenario tree using the historical inflow data.

3.1 Streamflow scenario tree generation

In this section, we generate a streamflow scenario tree by the neural gas method. Specifically, we first collect the historical TGR monthly streamflow data from 1956-2009 (54 years). We then pre-specify a scenario tree structure that consists of 24 scenarios, as Fig. 10(a)

shows. Note our model starts from October, which is also stage 1. The generated scenario tree consists of 24 representative scenarios, including their nodal values and probabilities. Fig. 10(b) and Table 1 show these results. We see from Fig. 10(b) that the highest inflows to TGR usually occur during the flood season (Jun.-Sep., or stage 9-12).
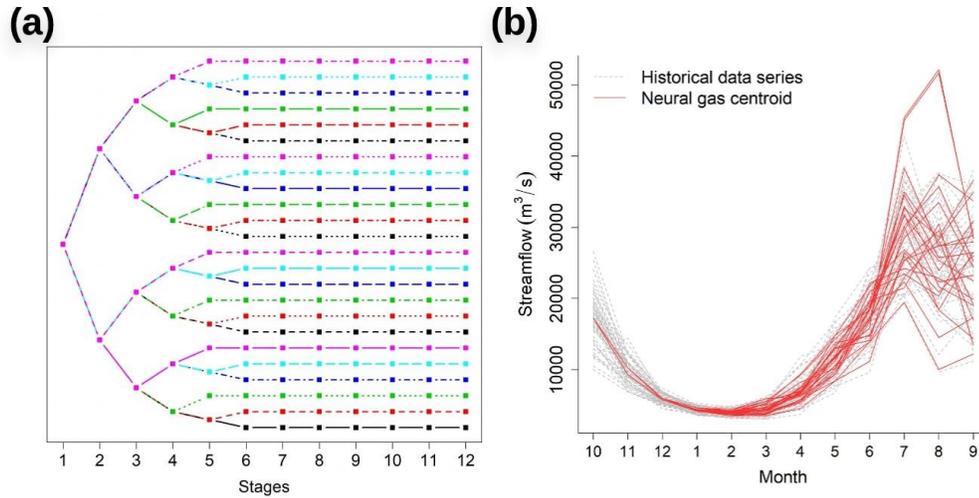
Figure 10. (a) The tree structure of the 24-scenario inflow scenario tree; (b) Historical series value and scenario (i.e., centroids) nodal values

Table 1. The probability of each scenario in the scenario tree

Scenario Probability							
#1	0.02	#7	0.02	#13	0.02	#19	0.04
#2	0.00	#8	0.02	#14	0.04	#20	0.06
#3	0.07	#9	0.04	#15	0.06	#21	0.06
#4	0.04	#10	0.06	#16	0.06	#22	0.04
#5	0.04	#11	0.04	#17	0.04	#23	0.06
#6	0.06	#12	0.09	#18	0.02	#24	0.06

3.2 Produced Pareto front and the optimal solution

We apply the proposed SMOP reservoir operation model to the TGR. Some constraint parameter values in the model are selected as follows: $V_{initial} = 2.9 \times 10^{10} \ m^3$; $N_{min} = 0$, $N_{max} = 22500 \text{MW}$;

71

$R_{\min} = 0$, $R_{\max} = 2.5 \times 10^4 \ m^3 / s$; and $SP_{\min} = 0$, $SP_{\max} = 9.8 \times 10^4 \ m^3 / s$. To comply with the reservoir operating rule for the non-flood season (Oct.-May), the reservoir storage limits are $V_{t,\min} = 1.71 \times 10^{10} \ m^3$ and $V_{t,\max} = 3.93 \times 10^{10} \ m^3$, corresponding to reservoir water levels of 145m and 175m, respectively. For the flood season (Jun.-Sep.), we set $V_{t,\min} = 1.71 \times 10^{10} \ m^3$ and $V_{t,\max} = 1.76 \times 10^{10} \ m^3$, except that the ending storage of Sept. is set to be greater than or equal to the initial storage $V_{initial}$. The comprehensive output coefficient (i.e., *K*) for the TGR has a value of 8.8 (Liu et al., 2011; Yang et al., 2018). Fig. 10(a) shows the inflow scenario tree used in the model. To run the stochastic programming with recourse model with a rolling horizon, a deterministic inflow forecast is needed for the immediate stage (1st stage), and beyond the immediate stage the inflow stochasticity is modeled by the scenario tree.

To test our proposed model, we obtain synthetic deterministic inflow forecast series of a dry year, a normal year, and a wet year from inflow data of years 2010, 1986, and 1973, with small perturbations. We then test our proposed method against all three different hydrological year inflow series. Note that in real-time operation, a deterministic inflow forecast should be made available for the immediate stage.

Starting from stage 1 (Oct.) with the deterministic inflow forecast, we produce the Pareto front using the modified constrained NSGA-II. The front shows the tradeoff between the two conflicting objectives: maximizing the total expected hydropower generation E and maximizing the average expected ecological assurance rate Λ. Next, we employ the proposed linear spline utility function with regression to select the best compromise Pareto optimal point. Once we have selected the best compromise Pareto optimal point for stage 1, the optimal water release for stage 1 can be determined accordingly. Then the initial state of stage 2 (Nov.) can be determined,

and the planning window moves to [2, *T*], given the deterministic stage 2 inflow forecast. This

procedure continues until the water release of the final stage *T* is obtained. Fig. 11 shows the

results (i.e., the Pareto front and the best compromise solution) of selected stages [stage 1 (Oct.),

stage 2 (Nov.), stage 3 (Dec.), and stage 6 (Mar.)], using the dry year test series as an example.

Other stages and test series yield similar results.

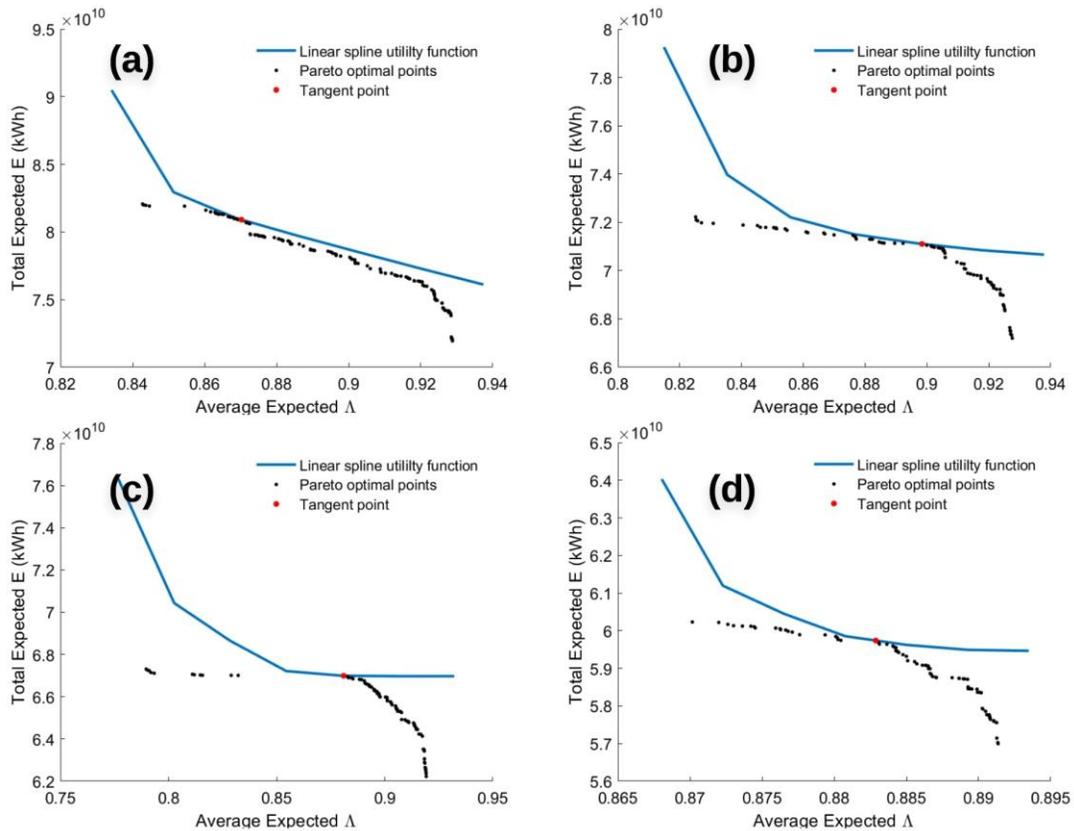**Figure 11.** Pareto optimal points and linear spline utility function for different stages under the dry year

test series: (a) Stage 1 in Oct.; (b) Stage 2 in Nov.; (c) Stage 3 in Dec.; and (d) Stage 6 in Mar.

From Fig. 11, we see that the proposed modified NSGA-II with the adjustment step works

as expected. It produces a relatively complete Pareto front at each stage by searching with a wide

range of Pareto optimal points. Second, for each stage, the "best compromise" Pareto optimal point is selected (in red) based on the proposed linear spline utility function with regression. Without relying on expert experience on preference coefficients β_{k_1}, we gain information about the preference coefficients directly from the Pareto front itself. The Pareto fronts in Fig.11(b) and (c) approximately can be assumed to be convex, and both of their "knee points" are captured under the proposed utility function. While the Pareto fronts in Fig.11(a) and (d) are more non-convex, our tangent points still fall into their "knee regions," even though the "knee regions" may not be unique. This clearly shows the advantage of our proposed utility function. That is, the selected tangent point is likely to occur in the preferable "knee region" of the generated Pareto front, regardless of its convexity, and it satisfies the law of diminishing MRS.

Also, Fig. 11(a) shows the "best compromise" solution for stage 1, so it reflects the compromise between the total expected hydropower output and the average assurance rate for the entire planning horizon. According to its selection, we say the total expected hydropower output is approximately 8.2×10^{10} kWh, with the average expected assurance rate for each month as 0.87. Note these are the expected values at the beginning of the entire one-year planning horizon, not the actual values. The actual values will depend on the actual forecasted inflow as well as the compromise solutions selected in the ensuing stages.

3.3 Comparisons among utility functions on the optimal Pareto point selection

Using different utility functions will generally yield different Pareto optimal point selections. As demonstrated in section 2.3, adopting the traditional linear utility function is equivalent to using the traditional scalarization method to handle conflicting objectives, which assigns a constant weight for each objective. In this section, we compare the proposed linear spline utility with the

traditional linear and quadratic utility, where we can see the difference among these methods on the optimal Pareto point selection. The Pareto front at stage 6 in the dry year is used to demonstrate the results, shown in Fig. 12.
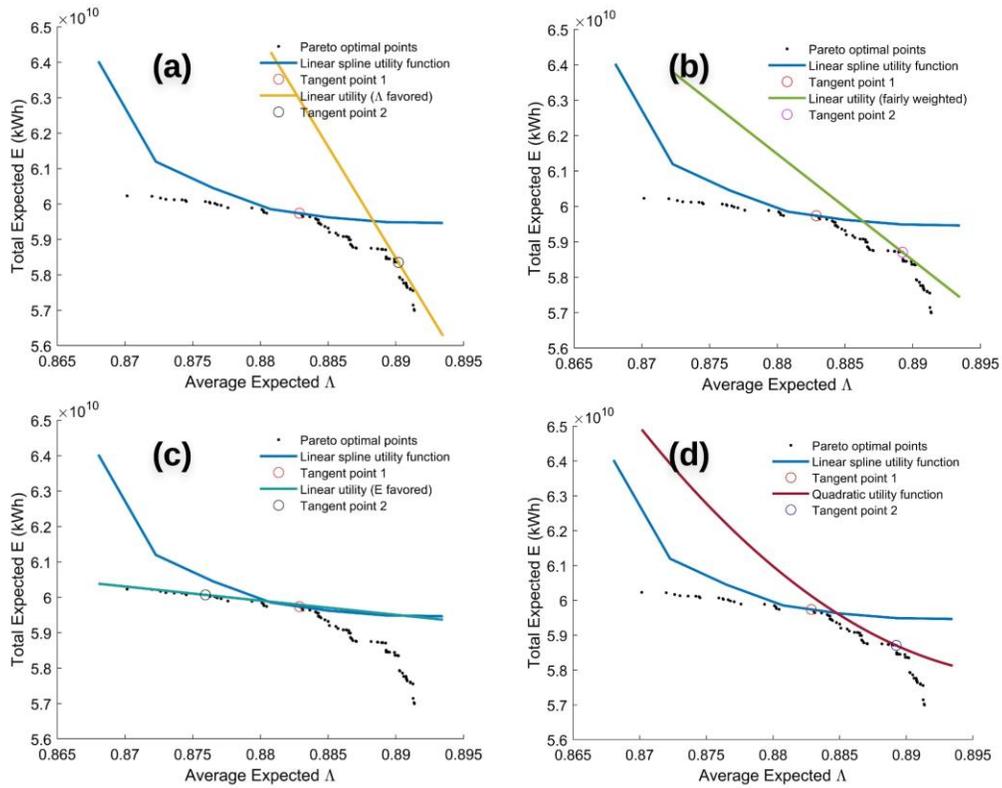
Figure 12. Comparison among different utility functions: (a) Linear utility favored on Λ; (b) Linear utility fairly weighted; (c) Linear utility favored on E; (d) Quadratic utility.

First, it is difficult to determine the weight for each objective in the traditional linear and quadratic utility function. People usually need to set the preference coefficients (i.e., weights) based on experience. Fig. 12(a) – (c) show three traditional linear utility functions with various preference coefficients on the assurance rate, from the highest value to the lowest value. We see that the optimal Pareto point selections based on these linear utility functions highly depend on the weights, thus the quality of the selection relies on these weight assignment accuracies. In contrast,

our proposed linear spline utility with the regression method does not require any information from expert experience to set up the preference coefficients. Instead, it assumes that the preference is related to a linear relationship of the two objectives in the Pareto front, as discussed in section 2.3.

Second, the traditional linear utility function has the inherent drawback that it does not satisfy the law of diminishing MRS, since the preference coefficient is fixed in the entire domain of each objective. Therefore, the selection based on this approach is not consistent with standard economic theory. On the contrary, our linear spline utility and the quadratic utility do not have this problem, since their preference coefficients can change in each segment of the non-reference objective's domain. However, a disadvantage of using a quadratic utility function is that its preference coefficient changes frequently, which leads to an unclear interpretation of the tradeoff relationship between the two objectives. Third, although the "knee point" may not necessarily always be the right choice, it is usually regarded as one of the "best" compromise points on a Pareto front (Branke et al., 2004; Deb and Gupta, 2011). It is usually difficult to select the point in the "knee region", for both traditional linear and quadratic utility functions. However, with our proposed regression method, the linear spline utility function is constructed in such a way that it is likely to select the optimal point in the "knee region" of a Pareto point, as shown in Fig. 12.

3.4 Optimal water release and reservoir storage for different inflow test series

The optimal water release corresponds to the selected optimal Pareto point. At each stage, once the optimal Pareto point is selected, the values of the corresponding decision variables [i.e., water release for all stages in the planning horizon] are determined. Then the water release for the stage under consideration can be obtained. In this section, the optimal Pareto point of each stage is selected by our proposed linear spline utility function. Fig. 13 shows the obtained releases and the corresponding reservoir storages for all three different hydrological year inflow test series.
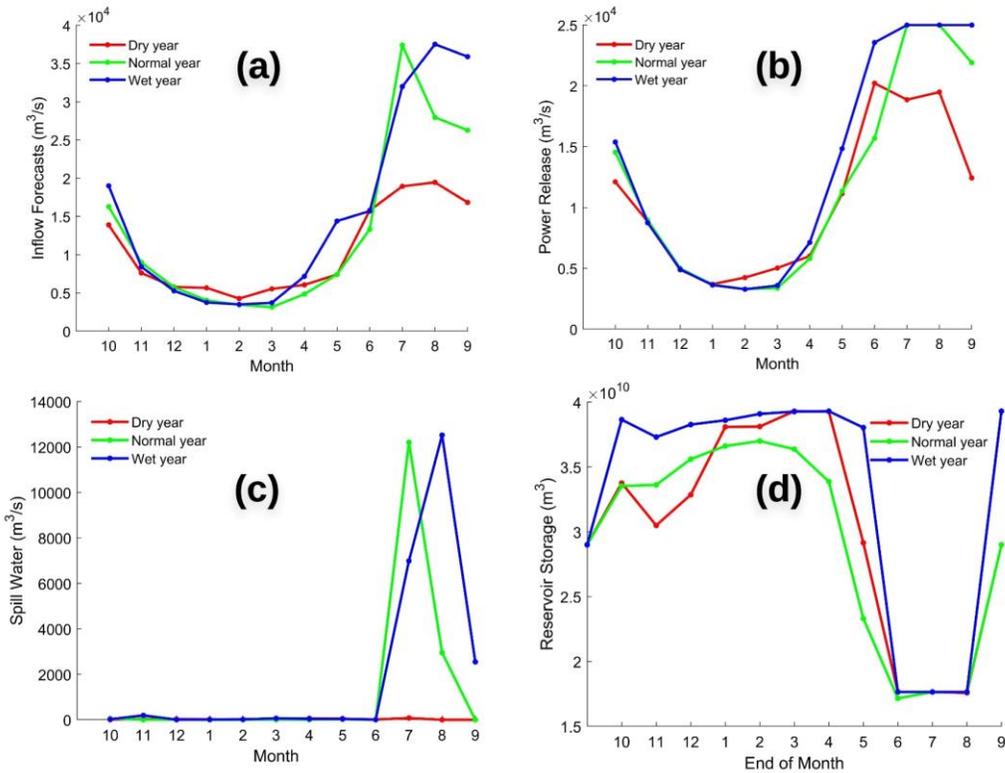
Figure 13. Different hydrological years comparison: (a) Inflow forecasts; (b) Power release; (c) Spill water; (d) Reservoir ending storage

Fig.13(a) shows the test inflow forecast data used for the dry year, normal year, and wet year. The most significant inflow difference among these three hydrological years occurs around the flood season (i.e., May-Sept.). The dry year inflow is around half of the inflow of the normal year and the wet year in the flood season, but it has slightly higher inflow than the normal year and wet year in the non-flood season (Dec.-Mar.). The wet year has more inflow than the normal year, mainly in pre-flood (i.e., May) and post-flood (i.e., Sep. and Oct.) seasons. These inflow series serve as the deterministic inflow forecasts for the immediate stage of the model. Important model results are then shown in Fig. 13(b), (c), and (d).

Fig. 13(b) shows the obtained power releases. The power releases depend on the compromise at each month, the inflow forecast, and the reservoir storage limit. This can be further

analyzed through the reservoir ending storage at each month, as shown in Fig. 13(d). In general, we see that the power release reveals a similar pattern with the inflow forecasts for different hydrological years. That is, the higher the inflow forecast for a given month, the more power release. For example, the normal year inflow forecast is slightly lower than the dry year during January to April, which leads to a lower power release of the normal year than the dry year in this period.

Fig. 13(c) shows that there are no spills (non-power releases) during the entire planning horizon for the dry year. This is because all water released in the dry year is used to produce hydropower, which aligns with one of the model objectives, maximizing hydropower output. For the normal and wet year, spills occur in the flood season, as the inflow forecasts exceed the upper power release limit.

The reservoir ending storage results are shown in Fig. 13(d). For the dry year and the wet year, we see that the reservoir slowly refills to its normal pool level by the end of March, but with a storage decrease between October and November. This reflects the multi-objective compromise: A single power output maximization objective would prefer a sharp water level increase to the normal pool level and maintain this water level until the flood season (demonstrated below in Fig. 15). However, the ecological objective requires a certain amount of water release during each month, which prevents the reservoir from quickly refilling to the maximum storage. For example, the decrease in reservoir storage during October indicates the need for water release for ecological purposes during that month. For the normal year, the inflow forecasts are too low in the non-flood period, especially from January to April, which prevents the reservoir from refilling to the normal pool level.

For all three hydrological years, starting from April, the reservoir begins to gradually empty the storage by the beginning of the flood season. During the flood season (June-Aug.), the reservoir water level is maintained at the flood control level. The ending storage (Sept.) is greater than the initial storage. In sum, reservoir storage variation depends on water release choices at each stage, which further depend on the compromise made between the two objectives at each stage. But in general, we see that the proposed model can produce reasonable optimal solutions when dealing with two conflicting objectives for all three different hydrological scenarios.

Also, according to the obtained water release and reservoir storage from our method, we calculate the power output and ecological assurance rate in each month for the three different hydrological years. Fig. 14 shows the results.
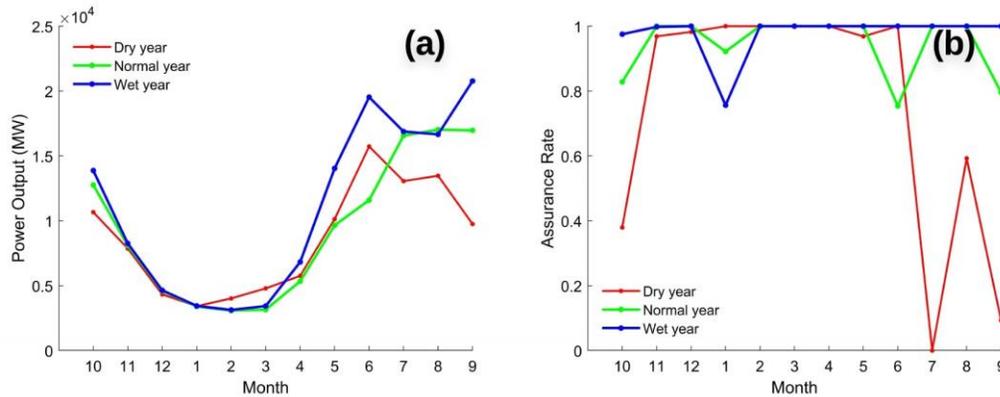
Figure 14. (a) The power output result for each month; (b) The assurance rate result for each month

As Fig. 14 shows, power output and the assurance rate are in general negatively correlated, which indicates the conflicting goals. The minimum power output, which is around 4,900MW, occurs in the non-flood season for all three testing series. In general, the power output is positively correlated with the power release and the gross average water head. Thus, the higher the reservoir water level (reservoir storage), the more power can be generated. For example, the dry year and

the normal year have almost identical power releases in April [as shown in Fig. 13(b)]. However, the average reservoir storage in April for the dry year is higher than the normal year, which leads to a slightly higher power output in April for the dry year than the normal year.

For the assurance rate, ecological needs can be satisfied for most months for all three testing years, as these values are close to one. The wet year has the highest annual assurance rate, due to its abundant inflow forecasts. For the dry year, a zero assurance rate in July occurs as an outlier. This is because 1) the inflow forecast in July of the dry year was lower than the minimum ecological streamflow $Q_t^{eco,\min}$, and 2) during the flood season the reservoir water level was required to be maintained at the flood-limited water level (145m). Similarly, a low assurance rate in September was due to the fact that the reservoir was required to be refilled at least to $V_{initial}$, thus not enough water could be provided to ensure a high ending ecological assurance rate.

3.5 Comparisons with traditional single-objective stochastic models

To demonstrate the utility of our proposed stochastic approach for tradeoff analysis between two conflicting objectives, we compare our proposed model with traditional single-objective stochastic programming. Specifically, we solve two single-objective stochastic programming with recourse models, with the objective of 1) only maximizing the total expected energy, and 2) only maximizing the average expected ecological benefits. Note that our proposed multi-objective stochastic model maximizes utility and also can be considered a single-objective utility-maximizing stochastic model, in which utility is defined by the proposed linear spline utility function. Fig. 15 compares the models and shows the results, exemplified by the dry year inflow test series.
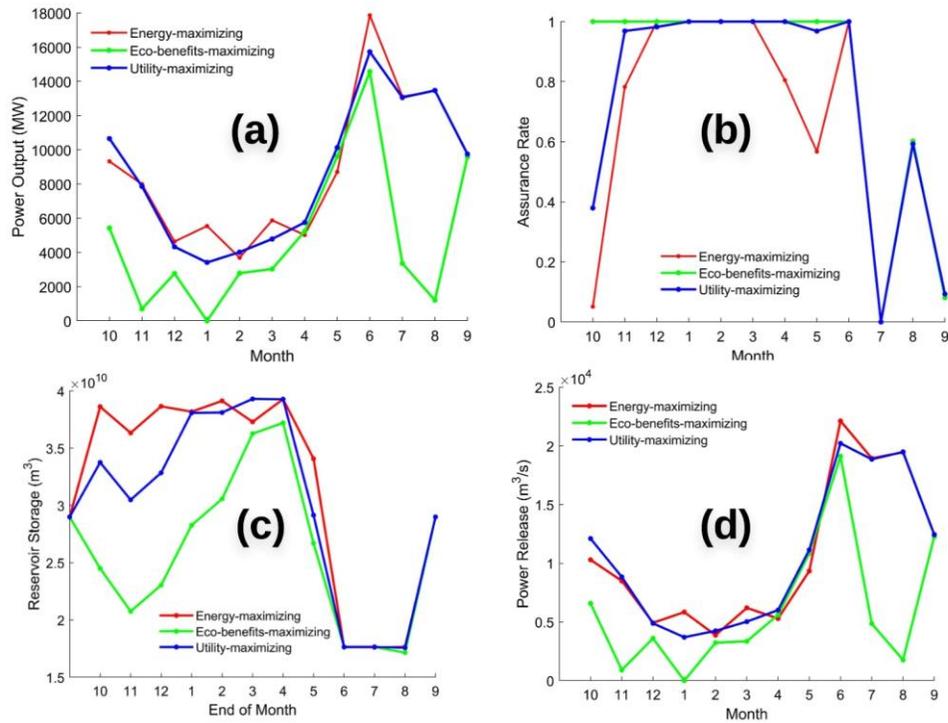
Figure 15. Model result comparisons: (a) Power output; (b) Assurance rate; (c) Reservoir ending storage;

(d) Power release

Fig. 15(a) shows the power output comparison. It clearly can be seen that the energy-maximizing model generates the highest power output for most stages, followed by the utility-maximizing model, and finally the eco-benefits-maximizing model. This result is consistent with the reservoir storage result shown in Fig. 15(c): In the energy-maximizing model, water level is kept at a higher level for most stages, since a high water level is beneficial to hydropower production. In contrast, the water level for eco-benefits maximizing is generally kept at a lower level, since this model only aims at satisfying the downstream assurance rate requirement so that it tends to release water to downstream instead of retaining in the reservoir. The utility-maximizing model water level is in the middle, due to the compromise made between two conflicting objectives.

The assurance rate comparison is shown in Fig. 15(b). In general, the results follow the inverse pattern of Fig. 15(a). That is, the eco-benefits-maximizing model provides the highest assurance rate for most stages, followed by the utility-maximizing model, and finally the energy-maximizing model. The power release for the eco-benefits maximizing model is shown in Fig. 15(d). It is lower than the other two models as the eco-model does not have any energy goal, and part of its release water is in the form of the spill water rather than the power release. We also calculate the total annual energy output and assurance rate results by summing up the results over all stages for the three models, as shown in Table 2. We observe that results from the utility-maximizing model are between the results in terms of both energy output and ecological benefits. The energy-maximizing model generates 1.7×10^9 kWh more annual energy than the utility model, and the eco-benefits-maximizing model has 0.7 more annual assurance rate over the utility model.

Table 2. Total annual energy output and assurance rate results for three different models

	Total annual energy output (10^{10} kWh)	Total annual assurance rate
Energy-maximizing	7.69	7.89
Eco-benefits-maximizing	4.25	9.68
Utility-maximizing	7.52	8.98

4. Conclusion

In this paper, we proposed a multi-objective, multi-stage stochastic programming with recourse model for reservoir management and operation. The model simultaneously considers both multiple objectives and stochastically represented uncertainty. The stochasticity of the reservoir inflow is represented by the generated scenario tree. Using the concept of a rolling horizon for real-time reservoir operation, we converted the original model into a process of solving a sequence of two-stage stochastic programming with recourse problems. Typical of a stochastic programming with recourse model, the inflow forecast for the first stage (immediate stage) is deterministic, and, from

the second stage onward, inflow branches out into a scenario tree. The Pareto front for the immediate stage is generated by a modified constrained NSGA-II. To handle a large number of constraints and to ensure feasibility, we made the modification by adding an "adjustment" step that consists of several strategies to ensure feasibility.

After obtaining the Pareto front, we selected a single tradeoff point (the best compromise solution) using the proposed linear spline utility with regression method. Our proposed utility function has the following key advantages. First, it satisfies the law of diminishing MRS, and the weights are fixed in each segment with clear interpretations. Second, it does not necessarily rely on the specified weight of each objective or any pre-specified parameters if used along with the proposed regression method. Instead, it finds such information from the Pareto front. Third, it selects the best compromise solution that is likely to fall in the "knee regions" of the Pareto front.

In the TGR case study, we applied the proposed framework considering two conflicting objectives: 1) maximizing the total expected energy output in the planning horizon, and 2) maximizing the average expected ecological assurance rate in the planning horizon. The results show that the proposed model successfully produces the optimal water release, considering both inflow uncertainty and the tradeoff between the two conflicting objectives.

summary of reviewer comments provided by the Editor and Associate Editor also are greatly appreciated.

Appendix A

Table 3. Symbol descriptions

Symbol	Description
$E\,[*]$	Expectation operator
x	Decision vector
X	Solution space
$f(x,w)$	A general function associated with the uncertain parameter w
w	Uncertain parameter; Scenario
g	Recourse function
q, J, L, h	Realization of uncertain data at the second stage
f_1, f_2	The first and second objective
m	The number of objective functions
I	Total number of scenarios in the scenario tree
t	The scenario stage (i.e., stage);
t_{beg}	The beginning of stage t
t_{end}	The end of stage t
$[t_{beg}, t_{end}]$	The time period during stage t
Δt	The time duration between each stage
T	Total number of stages in the scenario tree
$w_{i,t}$	The node of scenario i at scenario tree stage t
E_t^i	The energy output in the time period $[t_{beg}, t_{end}]$ (i.e., during the stage t) for scenario i
R_t^i	The power release in the time period $[t_{beg}, t_{end}]$ for scenario i
SP_t^i	The non-power release (spill water) in the time period $[t_{beg}, t_{end}]$ for scenario i
$P(w_i)$	The probability of scenario i
Λ_t^i	The ecological assurance rate of stage t for scenario i
$Q_t^{eco,\min}$	The minimum ecological streamflow for the time period $[t_{beg}, t_{end}]$
$Q_t^{eco,pro}$	The appropriate ecological streamflow for the time period $[t_{beg}, t_{end}]$
Q_t	The total reservoir discharge in the time period $[t_{beg}, t_{end}]$
stg	Model-stage; The start of the rolling horizon;
V_t^i	The ending reservoir storage of stage t for scenario i

$V_{Initial}$	The initial reservoir storage
V_T	The ending reservoir storage of the entire planning horizon
W_t^i	The inflow in the time period of $[t_{beg}, t_{end}]$ for scenario i
D_t	The deterministic inflow forecast during period of $[t_{beg}, t_{end}]$
N_i^t	The power output during the period $[t_{beg}, t_{end}]$ for scenario i
H_t^i	The gross average water head during period $[t_{beg}, t_{end}]$ under scenario i
FN_1, FN_2	The functions of forebay water level and tailrace water level
K	The comprehensive output coefficient for the reservoir
$V_{t,min}, V_{t,max}$	The lower and upper bounds of the reservoir storage for each stage t
N_{min}, N_{max}	The minimum and maximum limits of power output
R_{min}, R_{max}	The lower and upper bounds of the power release
SP_{min}, SP_{max}	The lower and upper bounds of the non-power release
U	Utility function; utility
f_{ref}	The reference objective
n	The number of non-reference objectives
$k_1, ..., k_n$	The segments for each non-reference objective
β_0	Equivalent to the utility
$d_1, ..., d_n$	The number of knots for each non-reference objective
$\alpha_{k_1}, ..., \alpha_{k_n}$	The knots for each non-reference objective
$\beta_{k_1}, ..., \beta_{k_n}$	The increments of preference in each segment of each non-reference objective
$\hat{b}_{k_1}$	The linear regression coefficient of Pareto points for each segment k_1
$\hat{b}_{(1)}, \hat{b}_{(2)}, \hat{b}_{(3)}$	Sorted linear regression coefficient
$(f_1^{k_1}[i], f_{ref}^{k_1}[i])$	Pareto points in segment k_1

Appendix B

The adjustment strategies for Constraints (11) and (12) are shown in Fig. 3 and outlined in the following:

Phase I: Check Constraint (11) ($V_{t,min} \leq V_t^i \leq V_{t,max}$):

Step 1. Upon obtaining an original offspring solution, check if this solution satisfies Constraint (11). If yes, directly go to Phase II. If not, proceed to either Step 2 (fix the left constraint violation) or Step 3 (fix the right constraint violation).

Step 2. Check for $\forall i,t$, if $V_t^i < V_{t,\min}$. If it is true for some i and t, then let $SP_t^i\_new = W_t^i - (V_{t,\min} - V_{t-1}^i)/\Delta t - R_t^i$, where R_t^i is the power release in the original offspring solution. If $SP_i^t\_new \geq SP_{\min}$, then only update SP_i^t with $SP_i^t\_new$, with no update on R_t^i . By doing so, we ensure $V_t^i = V_{t,\min}$ for this particular i and t. If $SP_t^i\_new < SP_{\min}$, then we further let $SP_t^i\_new = SP_{\min}$ and also let $R_t^i\_new = W_t^i - (V_{t,\min} - V_{t-1}^i)/\Delta t - SP_{\min}$. Update SP_i^t with $SP_t^i\_new$, and R_t^i with $R_t^i\_new$, respectively. By doing so, we also ensure $V_t^i = V_{t,\min}$.

Step 3. Check for $\forall i,t$, if $V_t^i > V_{t,\max}$. If this is true for some i and t, then we let $R_t^i\_new = W_t^i - (V_{t,\max} - V_{t-1}^i)/\Delta t - SP_t^i$, where SP_t^i is the non-power release in the original offspring solution. If $R_t^i\_new \leq R_{\max}$, only update R_t^i with $R_t^i\_new$, with no update on SP_t^i . By doing so, we ensure $V_t^i = V_{t,\max}$ for the particular i and t. If $R_t^i\_new > R_{\max}$, then we further let $R_t^i\_new = R_{\max}$, and also let $SP_t^i\_new = W_t^i - (V_{t,\max} - V_{t-1}^i)/\Delta t - R_{\max}$. Update SP_t^i with $SP_t^i\_new$, and R_t^i with $R_t^i\_new$, respectively. By doing so, we also ensure $V_t^i = V_{t,\max}$.

Step 4. Once updating R_t^i and SP_t^i is complete, for $\forall i,t$, calculate $Q_t^i = R_t^i + SP_t^i$, which is the total outflow for each i and t that satisfies Constraint (11); Q_t^i will be fixed in the next Phase. Note that once Q_t^i is fixed, V_t^i is determined given W_t^i and V_{t-1}^i .

Phase II: Check Constraints (12) ($N_{\min} \leq N_t^i \leq N_{\max}$):

Step 5. Use the updated solution after Step 1-4. Check if this solution satisfies Constraint (12). If it does, jump to Step 8. If not, proceed to either Step 6 or Step 7.

Step 6. Check for $\forall i, t$, if $N_t^i < N_{\min}$. If this is true for some i and t, then we let $R_t^i \_ new = N_{\min} / (K \cdot H_t^i)$. Then $SP_t^i \_ new = Q_t^i - R_t^i \_ new$, where Q_t^i is calculated at Step 4. By doing so, we make $N_t^i = N_{\min}$ for the specific i and t. Update SP_t^i with $SP_t^i \_ new$, and R_t^i with $R_t^i \_ new$, respectively.

Step 7. Check for $\forall i, t$, if $N_t^i > N_{\max}$. If this is true for some i and t, then we let $R_t^i \_ new = N_{\max} / (K \cdot H_t^i)$. Then $SP_t^i \_ new = Q_t^i - R_t^i \_ new$. By doing so, we will have $N_t^i = N_{\max}$ for the particular i and t. Update SP_t^i with $SP_t^i \_ new$, and R_t^i with $R_t^i \_ new$, respectively.

Step 8. If, after Step 1-7 updates for $\forall i, t$, R_t^i and SP_t^i are in the feasible range of R_t^i and SP_t^i (e.g. between $R_{\min}$ and $R_{\max}$); the original infeasible solution is guaranteed to have mutated to a new feasible solution that satisfies Constraints (11) and (12). Now it can be used for non-domination sorting, tournament selection, and cross-over. If for some i and t, R_t^i or SP_t^i are not in the feasible range anymore, then this infeasible solution is discarded, which only happens in very rare cases. The entire procedure also is illustrated in Fig. 3.

References

Abdelaziz, F.B., Aouni, B. and El Fayedh, R., 2007. Multi-objective stochastic programming for portfolio selection. European Journal of Operational Research, 177(3), pp.1811-1823.

Azaron, A., Brown, K.N., Tarim, S.A. and Modarres, M., 2008. A multi-objective stochastic programming approach for supply chain design considering risk. International Journal of Production Economics, 116(1), pp.129-138.

Abdelaziz, F.B., 2012. Solution approaches for the multiobjective stochastic programming. European Journal of Operational Research, 216(1), pp.1-16.

Baldwin, R.E., 1948. Equilibrium in international trade: A diagrammatic analysis. The Quarterly Journal of Economics, 62(5), pp.748-762.

Becker, L. and Yeh, W.W.G., 1974. Optimization of real time operation of a multiple-reservoir system. Water Resources Research, 10(6), pp.1107-1112.

Besada, M. and Vázquez, C., 1999. The generalized marginal rate of substitution. Journal of Mathematical Economics, 31(4), pp.553-560.

Binh, T.T., 1999, January. A multiobjective evolutionary algorithm: The study cases. In Proceedings of the 1999 Genetic and Evolutionary Computation Conference. Workshop Program (pp. 127-128).

Boyd, S., Boyd, S.P. and Vandenberghe, L., 2004. Convex optimization. Cambridge university press.

Branke, J., Deb, K., Dierolf, H. and Osswald, M., 2004, September. Finding knees in multi-objective optimization. In International conference on parallel problem solving from nature (pp. 722-731). Springer, Berlin, Heidelberg.

Bath, S.K., Dhillon, J.S. and Kothari, D.P., 2010. Stochastic multi-objective generation dispatch. Electric Power Components and Systems, 32(11), pp.1083-1103.

Bechikh, S., Ben Said, L. and Ghédira, K., 2010, March. Searching for knee regions in multi-objective optimization using mobile reference points. In Proceedings of the 2010 ACM symposium on applied computing (pp. 1118-1125).

Bechikh, S., Said, L.B. and Ghédira, K., 2011. Searching for knee regions of the Pareto front using mobile reference points. Soft Computing, 15(9), pp.1807-1823.

Cass, D., 1965. Optimum growth in an aggregative model of capital accumulation. The Review of economic studies, 32(3), pp.233-240.

Chen, Z. Q. 2005. Ecological flow requirments for the middle and lower reaches of the Yangtze River. [In Chinese.] Nanjin, China: Hohai Univ.

Debreu, G., 1954. Representation of a preference ordering by a numerical function. Decision processes, 3, pp.159-165.

Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T.A.M.T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation, 6(2), pp.182-197.

Dittmer, T., 2005. Diminishing Marginal Utility in economics textbooks. The Journal of Economic Education, 36(4), pp.391-399.

Dittmann, R., Froehlich, F., Pohl, R. and Ostrowski, M., 2009. Optimum multi-objective reservoir operation with emphasis on flood control and ecology. Natural Hazards and Earth System Sciences, 9(6), p.1973.

Deb, K. and Gupta, S., 2011. Understanding knee points in bicriteria problems and their implications as preferred solution principles. Engineering optimization, 43(11), pp.1175-1204.

Deb, K., 2011. Multi-objective optimisation using evolutionary algorithms: an introduction. In Multi-objective evolutionary optimisation for product design and manufacturing (pp. 3-34). Springer, London.

Deb, K. and Jain, H., 2013. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. IEEE transactions on evolutionary computation, 18(4), pp.577-601.

Dhiman, H. and Deb D., 2020 Multi-criteria Decision-Making: An Overview. In: Decision and Control in Hybrid Wind Farms. Studies in Systems, Decision and Control, vol 253. Springer, Singapore

Foued, B.A. and Sameh, M., 2001. Application of goal programming in a multi-objective reservoir operation model in Tunisia. European Journal of Operational Research, 133(2), pp.352-361.

Fonseca, M.C., García-Sánchez, Á., Ortega-Mier, M. and Saldanha-da-Gama, F., 2010. A stochastic bi-objective location model for strategic reverse logistics. Top, 18(1), pp.158-184.

Gutjahr, W.J. and Reiter, P., 2010. Bi-objective project portfolio selection and staff assignment under uncertainty. Optimization, 59(3), pp.417-445.

Gutjahr, W.J. and Pichler, A., 2016. Stochastic multi-objective optimization: a survey on non-scalarizing methods. Annals of Operations Research, 236(2), pp.475-499.

Giuliani, M., Castelletti, A., Pianosi, F., Mason, E. and Reed, P.M., 2016a. Curses, tradeoffs, and scalable management: Advancing evolutionary multiobjective direct policy search to improve water reservoir operations. Journal of Water Resources Planning and Management, 142(2), p.04015050.

Giuliani, M., Li, Y., Cominola, A., Denaro, S., Mason, E. and Castelletti, A., 2016b. A Matlab toolbox for designing Multi-Objective Optimal Operations of water reservoir systems. Environmental Modelling & Software, 85, pp.293-298.

Giuliani, M., Quinn, J.D., Herman, J.D., Castelletti, A. and Reed, P.M., 2017. Scalable multiobjective control for large-scale water resources systems under uncertainty. IEEE Transactions on Control Systems Technology, 26(4), pp.1492-1499.

Hicks, J.R., 1939. Value and Capital, 2nd edn, 1946. Oxford: Clarendon Press. DURATION ANALYSIS AND ITS APPLICATIONS, 427, pp.449-470.

Hwang CL and Yoon K, 1981. Methods for multiple attribute decision making. Springer, Berlin, pp 58–191

Hanemann, W.M., 2006. The economic conception of water. Water Crisis: myth or reality, 61, pp.74-76.

Hnaien, F., Delorme, X. and Dolgui, A., 2010. Multi-objective optimization for inventory control in two-level assembly systems under uncertainty of lead times. Computers & operations research, 37(11), pp.1835-1843.

Hadka, D. and Reed, P., 2013. Borg: An auto-adaptive many-objective evolutionary computing framework. Evolutionary computation, 21(2), pp.231-259.

Johnstone, D.J. and Lindley, D.V., 2011. Elementary proof that mean–variance implies quadratic utility. Theory and Decision, 70(2), pp.149-155.

Jain, H. and Deb, K., 2013. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: handling constraints and extending to an adaptive approach. IEEE Transactions on evolutionary computation, 18(4), pp.602-622.

Jakob, W. and Blume, C., 2014. Pareto optimization or cascaded weighted sum: A comparison of concepts. Algorithms, 7(1), pp.166-185.

Kim, M., Hiroyasu, T., Miki, M. and Watanabe, S., 2004, September. SPEA2+: Improving the performance of the strength Pareto evolutionary algorithm 2. In International Conference on Parallel Problem Solving from Nature (pp. 742-751). Springer, Berlin, Heidelberg.

Kollat, J.B. and Reed, P.M., 2005, March. The value of online adaptive search: a performance comparison of NSGAII, ε-NSGAII and ε-MOEA. In International Conference on Evolutionary Multi-Criterion Optimization (pp. 386-398). Springer, Berlin, Heidelberg.

Koppa, A., Gebremichael, M., Zambon, R.C., Yeh, W.W.G. and Hopson, T.M., 2019. Seasonal Hydropower Planning for Data-Scarce Regions Using Multimodel Ensemble Forecasts, Remote Sensing Data, and Stochastic Programming. Water Resources Research, 55(11), pp.8583-8607.

Levy, H. and Markowitz, H.M., 1979. Approximating expected utility by a function of mean and variance. The American Economic Review, 69(3), pp.308-317.

Lahdelma, R. and Salminen, P., 2001. SMAA-2: Stochastic multicriteria acceptability analysis for group decision making. Operations research, 49(3), pp.444-454.

Laumanns, M., Thiele, L. and Zitzler, E., 2005. An adaptive scheme to generate the pareto front based on the epsilon-constraint method. In Dagstuhl Seminar Proceedings. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

Latorre, J.M., Cerisola, S. and Ramos, A., 2007. Clustering algorithms for scenario tree generation: Application to natural hydro inflows. European Journal of Operational Research, 181(3), pp.1339-1353.

Liu, P., Guo, S., Xu, X. and Chen, J., 2011. Derivation of aggregation-based joint operating rule curves for cascade hydropower reservoirs. Water resources management, 25(13), pp.3177-3200.

Liu, P., Li, L., Guo, S., Xiong, L., Zhang, W., Zhang, J. and Xu, C.Y., 2015. Optimal design of seasonal flood limited water levels and its application for the Three Gorges Reservoir. Journal of Hydrology, 527, pp.1045-1053.

Li, J., Zhu, F., Xu, B. and Yeh, W.W.G., 2019. Streamflow scenario tree reduction based on conditional Monte Carlo sampling and regularized optimization. Journal of Hydrology, 577, p.123943.

Moore, H.L., 1907. The Efficiency Theory of Wages. The Economic Journal, 17(68), pp.571-579.

Martinetz, T. and Schulten, K., 1991. A" neural-gas" network learns topologies.

Martinetz, T.M., Berkovich, S.G. and Schulten, K.J., 1993. 'Neural-gas' network for vector quantization and its application to time-series prediction. IEEE transactions on neural networks, 4(4), pp.558-569.

Melato, M., Hammer, B. and Hormann, K., 2007. Neural gas for surface reconstruction. Institut für Informatik-IfI Technical Report Series.

Marshall, A., 2009. Principles of economics: unabridged eighth edition. Cosimo, Inc..

Maier, H.R., Kapelan, Z., Kasprzyk, J., Kollat, J., Matott, L.S., Cunha, M.C., Dandy, G.C., Gibbs, M.S., Keedwell, E., Marchi, A. and Ostfeld, A., 2014. Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges and future directions. Environmental Modelling & Software, 62, pp.271-299.

Ortiz-Partida, J.P., Kahil, T., Ermolieva, T., Ermoliev, Y., Lane, B., Sandoval-Solis, S. and Wada, Y., 2019. A Two-Stage Stochastic Optimization for Robust Operation of Multipurpose Reservoirs. *Water Resources Management*, *33*(11), pp.3815-3830.

Pflug, G.C. and Pichler, A., 2014. Multistage stochastic optimization. Cham: Springer International Publishing.

Revankar, N.S., 1971. A class of variable elasticity of substitution production functions. Econometrica: Journal of the Econometric Society, pp.61-71.

Reddy, M.J. and Nagesh Kumar, D., 2007. Multi‑objective particle swarm optimization for generating optimal trade‑offs in reservoir operation. Hydrological Processes: An International Journal, 21(21), pp.2897-2909.

Roman, D., Darby-Dowman, K. and Mitra, G., 2007. Mean-risk models using two risk measures: a multi-objective approach. Quantitative Finance, 7(4), pp.443-458.

Reed, P.M., Hadka, D., Herman, J.D., Kasprzyk, J.R. and Kollat, J.B., 2013. Evolutionary multiobjective optimization in water resources: The past, present, and future. Advances in water resources, 51, pp.438-456.

Ramos, M.A., Boix, M., Montastruc, L. and Domenech, S., 2014. Multiobjective optimization using goal programming for industrial water network design. Industrial & engineering chemistry research, 53(45), pp.17722-17735.

Rath, S., Gendreau, M. and Gutjahr, W.J., 2016. Bi-objective stochastic programming models for determining depot locations in disaster relief operations. International Transactions in Operational Research, 23(6), pp.997-1023.

Shapiro, A. and Philpott, A., 2007. A tutorial on stochastic programming. Manuscript. Available at www2. isye. gatech. edu/ashapiro/publications. html, 17.

Schütz, P., Tomasgard, A. and Ahmed, S., 2009. Supply chain design under uncertainty using sample average approximation and dual decomposition. European journal of operational research, 199(2), pp.409-419.

Satopaa, V., Albrecht, J., Irwin, D. and Raghavan, B., 2011, June. Finding a" kneedle" in a haystack: Detecting knee points in system behavior. In 2011 31st international conference on distributed computing systems workshops (pp. 166-171). IEEE.

Séguin, S., Fleten, S.E., Côté, P., Pichler, A. and Audet, C., 2017. Stochastic short-term hydropower planning with inflow scenario trees. European Journal of Operational Research, 259(3), pp.1156-1168.

Tamiz, M., Jones, D.F. and El-Darzi, E., 1995. A review of goal programming and its applications. Annals of Operations Research, 58(1), pp.39-53.

Tricoire, F., Graf, A. and Gutjahr, W.J., 2012. The bi-objective stochastic covering tour problem. Computers & operations research, 39(7), pp.1582-1592.

Tarebari, H., Javid, A.H., Mirbagheri, S.A. and Fahmi, H., 2018. Multi-objective surface water resource management considering conflict resolution and utility function optimization. Water Resources Management, 32(14), pp.4487-4509.

Uzawa, H., 1962. Production functions with constant elasticities of substitution. The Review of Economic Studies, 29(4), pp.291-299.

Vrugt, J.A. and Robinson, B.A., 2007. Improved evolutionary optimization from genetically adaptive multimethod search. Proceedings of the National Academy of Sciences, 104(3), pp.708-711.

Wang, F., Wang, L., Zhou, H., Saavedra Valeriano, O.C., Koike, T. and Li, W., 2012. Ensemble hydrological prediction-based real-time optimization of a multiobjective reservoir during flood season in a semiarid basin with global numerical weather predictions. Water Resources Research, 48(7).

White, C., 2015, April. Understanding water markets: Public vs. private goods. In Global Water Forum (pp. 1-8).

Wang, H., Olhofer, M. and Jin, Y., 2017. A mini-review on preference modeling and articulation in multi-objective optimization: current status and challenges. Complex & Intelligent Systems, 3(4), pp.233-245.

Wang, J., Cheng, C., Shen, J., Cao, R. and Yeh, W.W.G., 2018. Optimization of Large-Scale Daily Hydrothermal System Operations With Multiple Objectives. Water Resources Research, 54(4), pp.2834-2850.

Xu, B., Zhong, P.A., Zambon, R.C., Zhao, Y. and Yeh, W.W.G., 2015a. Scenario tree reduction in stochastic programming with recourse for hydropower operations. Water Resources Research, 51(8), pp.6359-6380.

Xu, B., Zhong, P.A., Stanko, Z., Zhao, Y. and Yeh, W.W.G., 2015b. A multiobjective short-term optimal operation model for a cascade system of reservoirs considering the impact on long-term energy production. Water Resources Research, 51(5), pp.3353-3369.

Yeh, W.W.G. and Becker, L., 1982. Multiobjective analysis of multireservoir operations. Water resources research, 18(5), pp.1326-1336.

Yeh, W.W.G., 1985. Reservoir management and operations models: A state-of-the-art review. Water resources research, 21(12), pp.1797-1818.

Yang, G., Guo, S., Liu, P., Li, L. and Xu, C., 2017. Multiobjective reservoir operating rules based on cascade reservoir input variable selection method. Water Resources Research, 53(4), pp.3446-3463.

Yang W, Xu K, Lian J, Ma C, Bin L, 2018. Integrated flood vulnerability assessment approach based on TOPSIS and shannon entropy methods. Ecol Indic 89:269–280. https://doi.org/10.1016/j.ecolind.2018.02.015

Yu, Y., Wang, C., Wang, P., Hou, J. and Qian, J., 2017. Assessment of multi-objective reservoir operation in the middle and lower Yangtze River based on a flow regime influenced by the Three Gorges Project. Ecological Informatics, 38, pp.115-125.

Yang, Z., Yang, K., Hu, H. and Su, L., 2019. The cascade reservoirs multi-objective ecological operation optimization considering different ecological flow demand. Water resources management, 33(1), pp.207-228.

Zhao, T., Yang, D., Cai, X., Zhao, J. and Wang, H., 2012. Identifying effective forecast horizon for real-time reservoir operation under a limited inflow forecast. Water Resources Research, 48(1).

Zhao, J., Cai, X. and Wang, Z., 2013. Comparing administered and market-based water allocation systems through a consistent agent-based modeling framework. Journal of environmental management, 123, pp.120-130.

Zhao, T. and Zhao, J., 2014. Improved multiple-objective dynamic programming model for reservoir operation optimization. Journal of Hydroinformatics, 16(5), pp.1142-1157.

Zhao, T., Zhao, J., Liu, P. and Lei, X., 2015. Evaluating the marginal utility principle for long-term hydropower scheduling. Energy Conversion and Management, 106, pp.213-223.

Zhu, F., Zhong, P.A., Wu, Y.N., Sun, Y., Chen, J. and Jia, B., 2017a. SMAA-based stochastic multi-criteria decision making for reservoir flood control operation. Stochastic Environmental Research and Risk Assessment, 31(6), pp.1485-1497.

Zhu, F., Zhong, P.A., Sun, Y. and Yeh, W.W.G., 2017b. Real-time optimal flood control decision making and risk propagation under multiple uncertainties. Water Resources Research, 53(12), pp.10635-10654.

Zhang, W., Wang, X., Lei, X., Liu, P., Yan, X. and Feng, M., 2020. Multicriteria Decision-Making Model of Reservoir Operation Considering Balanced Applicability in Past and Future: Application to the Three Gorges Reservoir. Journal of Water Resources Planning and Management, 146(6), p.04020033.

Chapter 5

CONCLUSION, DISCUSSIONS, AND FUTURE RESEARCH

This dissertation aims at improving the accuracy, efficiency, and applicability of the traditional stochastic programming models in reservoir operation, so that a better optimized releasing strategy can be obtained under the uncertain inflow. In particular, three underlying drawbacks of the traditional stochastic programming model are studied and optimized: 1) increasing the parameter estimation accuracy when fitting the inflow distribution; 2) improving the efficiency of the multi-stage stochastic programming with recourse model by proposing a scenario tree reduction method; and 3) expanding the applicability of the stochastic programming model by integrating with multi-objective programming. This dissertation tests the proposed methods on different case studies, including eight watersheds across the U.S., Qingjiang cascade reservoir system in Yangtze River, China, and the Three Gorges Reservoir in China.

Specifically, in Chapter 2, a Bayesian hierarchical model (BHM) is proposed for estimating the statistical parameters for monthly average streamflows. It is assumed that the monthly average streamflow follows a three parameter, log-normal distribution (LN3). When estimating a distribution parameter of a given month, the proposed BHM utilizes historical observations not only from the month under consideration but also from all other months. This is different from traditional statistical parameter estimation methods that only use historical observations for the month under consideration. Cross-Validation (CV) method and the log-likelihood are selected to evaluate the performance of BHM against selected traditional parameter estimation method methods, including LMLE, L-MOM, MME, MMME, etc.

In the case study, this dissertation selects eight watersheds across the United States, where historical unimpaired streamflows are collected. A Shapiro-Wilk normality test is employed to validate the LN3 assumption. The results show that the proposed BHM produced better parameter estimates than the selected traditional parameter estimation methods for all data sizes (long, medium, and short). The fewer the observation data, the more the proposed method improves compared with the traditional methods. The two reasons that BHM produced better parameter estimates are: 1) BHM utilizes historical observations not only from the month under consideration but also from all other months, and 2) Due to the shrinkage, the Bayesian estimator from BHM is guaranteed to be no worse than the MLE based estimator. The proposed BHM is especially suited for parameter estimation where historical observations are limited. In addition, when compared to an auto-regression model, the proposed BHM shows its advantage as a data-driven model with fewer assumptions.

The proposed method in Chapter 2 increases the accuracy of stochastic programming models, especially for the chance-constrained model. This method can also be used to better estimate parameters of any random variables that follow LN3 distribution. Some potential future research opportunities for this method include: 1) reduce the potential parameter uncertainty loss by switching the currently applied sequential estimation approach to a full Bayesian analysis based on the joint posterior. 2) investigate on the choices of the informative hyperprior and evaluate their impacts.

In Chapter 3, a novel scenario tree reduction method is developed based on a stepwise conditional Monte Carlo sampling and regularized optimization. A neural gas algorithm is first employed for scenario tree generation, then a stepwise conditional Monte Carlo sampling method is established for systemically reducing the number of scenarios from the full tree. A regularized optimization model based on ridge regression and moment matching is further developed to determine the posterior scenario probability. The developed method is aimed at improving the efficiency of the multi-stage stochastic programming with recourse model, and it is particularly suited for reducing a streamflow scenario tree for reservoir operations. This is because: 1) this method does not require updating nodal values at each stage. Therefore, it is consistent with the definition of scenario tree reduction, which is known as determining a subset of the initial scenario tree and assigning new probability to the reduced scenarios (Growe-Kuska et al., 2003); 2) this method stabilizes the reduced tree scenario's probability, and the physical meaning of the streamflow scenario probability can be interpreted easily; 3) this method does not rely on the probability metric but takes advantage of the basic moment matching technique to provide a direct moment matching between the historical data series and the reduced tree.

Qingjiang cascade reservoir system is used as the case study. The results show that the reduced tree with 35% reduction level can still maintain robust moment preservations, including the mean, variance, lag-one covariance, cross-site covariance, and scenario probability. The stability test indicates that the proposed conditional Monte Carlo sampling method is stable and converges within a reasonable number of scenario combinations. Potential future studies may include evaluating the quality of the reduced scenario tree obtained by the proposed method from the optimization solution perspective (Felten et al., 2002).

In Chapter 4, a multi-objective, multi-stage stochastic programming with recourse model is proposed for reservoir management and operation, where the utility theory is adopted to select the best compromise solution from the Pareto front. This proposed model is aimed at expanding the applicability of stochastic programming model by integrating with multi-objective programming. Specifically, a multi-stage streamflow scenario tree is first generated by the neural gas method. Then the Pareto front at each stage is produced by a modified constrained NSGA-II. A single best compromise solution on the Pareto front must be selected for the immediate stage and the model moves forward one stage and is re-optimized over a moving planning horizon of fixed duration. The selection is achieved by a proposed linear spline utility function allied with regression. Our proposed utility function has the following advantages: 1) it satisfies the law of diminishing marginal rate of substitution, 2) it does not rely on the pre-specified weight or goal if used along with the regression method, and 3) it selects the best compromise solution that is likely to fall in the "knee regions" of the Pareto front.

The proposed optimization model is applied to the Three Gorges Reservoir (TGR) in China. The two conflicting objectives are 1) maximizing the total expected energy output in the planning horizon, and 2) maximizing the average expected ecological benefits in the planning horizon. The results show that the proposed model successfully produces the optimal water release policy under different hydrological test inflow scenarios, when considering both the inflow uncertainty and the tradeoff between the two conflicting objectives. As discussed in Chapter 4, one of the most important tasks for constructing a stochastic multi-objective model is to find a way for selecting the "best compromise" solution on the Pareto front. Thus, besides the proposed linear spline utility function method, other utility functions that satisfy the economic theory can be explored for future research.

Overall, this dissertation improves the accuracy, efficiency, and applicability of the traditional stochastic programming model. With such an improved stochastic programming model, a better optimized reservoir operation strategy can be obtained when considering the uncertain inflow. The topics discussed in this dissertation are also expected to provide insights for the future research in other fields of water resource management and operations.

REFERENCE

Besada, M. and Vázquez, C., 1999. The generalized marginal rate of substitution. Journal of Mathematical Economics, 31(4), pp.553-560.

Casey, M.S., Sen, S., 2005. The scenario generation algorithm for multistage stochastic linear programming. Math. Oper. Res. 30 (3), 615–631. https://doi.org/10.1287/moor.1050.0146.

Dupacová, J., Growe-Kuska, N., Römisch, W., 2003. Scenario reduction in stochasticprogramming - an approach using probability metrics. Math. Program. 95 (3), 493–511. https://doi.org/10.1007/s10107-002-0331-0.

Dittmer, T., 2005. Diminishing Marginal Utility in economics textbooks. The Journal of Economic Education, 36(4), pp.391-399.

Fleten, S.E., Høyland, K. and Wallace, S.W., 2002. The performance of stochastic dynamic and fixed mix portfolio models. European Journal of Operational Research, 140(1), pp.37-49.

Growe-Kuska, N., Heitsch, H. and Romisch, W., 2003, June. Scenario reduction and scenario tree construction for power management problems. In 2003 IEEE Bologna Power Tech Conference Proceedings, (Vol. 3, pp. 7-pp). IEEE.

Gutjahr, W.J. and Pichler, A., 2016. Stochastic multi-objective optimization: a survey on non-scalarizing methods. Annals of Operations Research, 236(2), pp.475-499.

Hicks, J.R., 1939. Value and Capital, 2nd edn, 1946. Oxford: Clarendon Press. DURATION ANALYSIS AND ITS APPLICATIONS, 427, pp.449-470.

Houck, M.H., 1979. A chance constrained optimization model for reservoir design and operation. Water Resources Research, 15(5), pp.1011-1016.

Li, J., Zhu, F., Xu, B. and Yeh, W.W.G., 2019. Streamflow scenario tree reduction based on conditional Monte Carlo sampling and regularized optimization. Journal of Hydrology, 577, p.123943.

Li, J., Zhou, Q. and Yeh, W.W.G., 2020. A Bayesian hierarchical model for estimating the statistical parameters in a three-parameter log-normal distribution for monthly average streamflows. Journal of Hydrology, 591, p.125265.

Mao, Q., Mueller, S.F. and Juang, H.M.H., 2000. Quantitative precipitation forecasting for the tennessee and cumberland river watersheds using the NCEP regional spectral model, Weather and forecasting, 15(1), pp.29–45, doi:10.1175/1520-0434(2000)015<0029:QPFFTT>2.0.CO;2.

Revelle, C., Joeres, E. and Kirby, W., 1969. The linear decision rule in reservoir management and design: 1, Development of the stochastic model. Water resources research, 5(4), pp.767-777.

Sreenivasan, K.R. and Vedula, S., 1996. Reservoir operation for hydropower optimization: a chance-constrained approach. Sadhana, 21(4), pp.503-510.

Séguin, S., Fleten, S. E., Côté, P., Pichler, A., and Audet, C., 2017. Stochastic short-term hydropower planning with inflow scenario trees. European Journal of Operational Research, 259(3), 1156–1168. https://doi.org/10.1016/j.ejor.2016.11.028

Tucci, C.E.M., Clarke, R.T., Collischonn, W., da Silva Dias, P.L. and de Oliveira, G.S., 2003. Long-term flow forecasts based on climate and hydrologic modeling: Uruguay River basin. Water Resources Research, 39(7).

White, C., 2015, April. Understanding water markets: Public vs. private goods. In Global Water Forum (pp. 1-8).

Xu, B., Zhong, P.A., Zambon, R.C., Zhao, Y. and Yeh, W.W.G., 2015. Scenario tree reduction in stochastic programming with recourse for hydropower operations. Water Resources Research, 51(8), pp.6359-6380.

Yeh, W.W.G. and Becker, L., 1982. Multiobjective analysis of multireservoir operations. Water resources research, 18(5), pp.1326-1336.

Yeh, W.W.G., 1985. Reservoir management and operations models: A state−of−the−art review. Water resources research, 21(12), pp.1797-1818.

Zhu, F., Zhong, P.A., Sun, Y. and Yeh, W.W.G., 2017. Real‐time optimal flood control decision making and risk propagation under multiple uncertainties. Water Resources Research, 53(12), pp.10635-10654.