


## Genetic programming for hydrological applications: to model or to forecast that is the question

Herath Mudiyansele Viraj Vidura Herath, Jayashree Chadalawada and Vladan Babovic 

### ABSTRACT

Genetic programming (GP) is a widely used machine learning (ML) algorithm that has been applied in water resources science and engineering since its conception in the early 1990s. However, similar to other ML applications, the GP algorithm is often used as a data fitting tool rather than as a model building instrument. We find this a gross underutilization of the GP capabilities. The most unique and distinct feature of GP that makes it distinctly different from the rest of ML techniques is its capability to produce explicit mathematical relationships between input and output variables. In the context of theory-guided data science (TGDS) which recently emerged as a new paradigm in ML with the main goal of blending the existing body of knowledge with ML techniques to induce physically sound models. Hence, TGDS has evolved into a popular data science paradigm, especially in scientific disciplines including water resources. Following these ideas, in our prior work, we developed two hydrologically informed rainfall-runoff model induction toolkits for lumped modelling and distributed modelling based on GP. In the current work, the two toolkits are applied using a different hydrological model building library. Here, the model building blocks are derived from the Sugawara TANK model template which represents the elements of hydrological knowledge. Results are compared against the traditional GP approach and suggest that GP as a rainfall-runoff model induction toolkit preserves the prediction power of the traditional GP short-term forecasting approach while benefiting to better understand the catchment runoff dynamics through the readily interpretable induced models.

**Key words** | genetic programming, machine learning, rainfall-runoff model induction, Sugawara TANK model, theory-guided data science

Herath Mudiyansele Viraj Vidura Herath  
Jayashree Chadalawada  
Vladan Babovic  (corresponding author)  
Department of Civil and Environmental  
Engineering,  
National University of Singapore,  
117576 Singapore,  
Singapore  
E-mail: [vladan@nus.edu.sg](mailto:vladan@nus.edu.sg)

### HIGHLIGHTS

- GP is used as a rainfall-runoff model induction toolkit rather than as a short-term forecasting mechanism.
- Hydrological knowledge has been integrated with the machine learning algorithm.
- This guides algorithm to induce physically sound and consistent model configurations.
- The model building blocks are inspired by the Sugawara TANK model template.
- They represent the elements of incorporated hydrological knowledge.

This is an Open Access article distributed under the terms of the Creative Commons Attribution Licence (CC BY 4.0), which permits copying, adaptation and redistribution, provided the original work is properly cited (<http://creativecommons.org/licenses/by/4.0/>).

doi: 10.2166/hydro.2021.179

## INTRODUCTION

Data science methods have shown success in many scientific fields including hydrology. However, it may be argued that, compared with the level of success in commercial fields, the potential in water resources has not been fully realized. There are two major reasons for this: a lack of the availability of labelled instances for model training and the black-box nature of data-driven models where a modeler has little or no knowledge about how the model makes its prediction (Karpatne *et al.* 2017). Although the data-driven models are often more accurate than more traditional hydrological physics-based, conceptual and empirical models in terms of predictive capabilities, they contribute little towards the advancement of scientific theories due to the lack of interpretability of the model configurations. Recently, a novel modelling paradigm called theory-guided data science (TGDS) (Karpatne *et al.* 2017) or physics informed machine learning (Babovic & Keijzer 2002; Babovic 2005, 2009; Physics Informed Machine Learning Conference 2016; Chadalawada *et al.* 2020) emerged to enhance the explainability of machine learning (ML) models in general and water resources in particular. In this approach, the existing body of knowledge is blended with ML algorithms to induce physically sound and consistent models.

Genetic programming (GP) (Koza 1992) is an ML algorithm that has been used for many applications in water resources science and engineering since its invention in the early 1990s. However, as per the state-of-the-art GP applications, the algorithm is often used as a data fitting tool instead of as a model building instrument. The data fitting makes the GP applications very similar to other ML algorithms, such as artificial neural networks or support vector machines. In hydrological rainfall-runoff modelling, the most frequent use of ML, including GP, is as a short-term forecasting tool. We find this to be an under-utilization of the GP capabilities. The most unique and distinct feature of GP that makes it so different from the rest of ML techniques lies in its capability to produce explicit mathematical relationships between input and output variables.

A recent paper (Addor & Melsen 2019), based on more than 1,500 peer-reviewed research articles, concluded that the model selection in hydrological modelling is more often driven by legacy rather than adequacy. Furthermore, the so-called uniqueness of the place has been identified as one of the key aspects of hydrological modelling (Beven 2020). Hence, an automatic model induction and model selection framework may serve as an alternative to the more traditional subjective model selection and should induce a model architecture that might be more adequate for the intended application.

Following this line of thought in our prior work, we developed two hydrologically informed rainfall-runoff model induction toolkits, namely Machine Learning Rainfall-Runoff Model Induction toolkit (ML-RR-MI) (Chadalawada *et al.* 2020) and Machine Induction Knowledge Augmented-System Hydrologique Asiatique (MIKA-SHA) (Herath *et al.* 2020). Both toolkits are capable of inducing fully fledged rainfall-runoff models for a catchment of interest where the former is used for lumped modelling and the latter is used for distributed modelling. In this approach, GP has been used as the ML induction algorithm to learn both model structure and optimize parameters simultaneously. Hydrological knowledge is incorporated through the building blocks introduced into the GP framework used to govern the induction of physically sound and consistent models. In prior applications of ML-RR-MI and MIKA-SHA, building blocks of two modular modelling frameworks namely SUPERFLEX (Fenicia *et al.* 2011; Kavetski & Fenicia 2011) and FUSE (Clark *et al.* 2008) were used as the elements of incorporated hydrological knowledge.

In the current study, both ML-RR-MI and MIKA-SHA are armed with a different model building library inspired by the Sugawara TANK model template (Sugawara 1979). One of the most unique features of our hydrologically informed model induction frameworks is the capability of integrating frameworks with any internally coherent (consistent) collection of building blocks as the elements of incorporated hydrological knowledge. Therefore, we chose to utilize model building components from the

Sugawara TANK model template for the current study in contrast to the model building components of flexible modelling frameworks used in earlier ML-RR-MI and MIKA-SHA applications to demonstrate the versatility of our approach.

The main objective of this study is to compare the performance of GP as a hydrological model induction engine against the GP as a short-term forecasting tool. Both lumped and semi-distributed rainfall-runoff model induction capabilities of GP have been evaluated on two United States watersheds.

The rest of the paper is arranged as follows. The next section provides the background of different hydrological modelling strategies. GP applications in water resources engineering are discussed in the section that follows. Traditional GP has been used for runoff forecasting of two United States watersheds and the results are included in the section on GP for hydrological forecasting. The section on GP for hydrological modelling describes the applications of the ML-RR-MI and MIKA-SHA toolkits along with the Sugawara TANK model-based building block library for lumped and distributed model induction for the same two catchments. The comparison between GP for runoff forecasting vs. hydrological modelling has been given in the next section. The last section summarizes the conclusions made through the research findings. Additional details are summarized in the Supplementary Appendix.

---

## HYDROLOGICAL MODELLING

Hydrological models play an important role in understanding catchment dynamics. In this section, different hydrological model-building strategies, some of which have been used in the present study are briefly discussed.

### Theory-based models vs. data science models

Here, we refer to both conceptual and physics-based hydrological models as theory-based models. In conceptual modelling, hydrological processes are represented mathematically with the reservoir units describing the catchment storages. Fluxes, reservoirs, closure relations, and transfer functions are the main building blocks of a conceptual

model. Conceptual models are several orders less complex than the physics-based models due to the conceptual representation of catchment dynamics instead of small-scale physics used in physics-based models. However, in conceptual models, there are little or no direct relations between the model parameters and physically measurable quantities in the field. Therefore, it is required to use calibration schemes to select appropriate parameter values which provide a reasonable match between observed and model-simulated values.

Two types of modelling approaches can be identified within the conceptual modelling: the models based on a single hypothesis (fixed models) and the models based on multiple hypotheses (flexible models). Fixed models are built around a general model architecture that gives satisfactory model performances over a fairly broad range of watersheds and climatic regions. On the other hand, flexible modelling frameworks provide model building blocks that can be arranged in different ways to test many hypotheses about catchment dynamics instead of the one fixed hypothesis in fixed models. Such robust nature of any modular modelling framework allows the modeller to consider the uniqueness of the area of their application.

Data science models, such as ML, gained popularity in many disciplines including hydrology with the advancement of computing power and data availability. The main advantage of data science models is their ability to use the available data to build the input-output relationships which provide actionable models with good predictive capabilities without relying on scientific theories. The short lead time predictive power of ML models is often superior to that of traditional physics-based and conceptual hydrological models due to their ability to capture the non-linearity, non-stationarity, noise complexity and dynamism of data (Yaseen *et al.* 2015). Another major advantage of an ML model is that it requires less effort to develop and calibrate than a physics-based model. Data science models, such as deep learning (DL) models, have shown more accurate performances in hydrograph predictions than the traditional approaches in ungauged catchments (Kratzert *et al.* 2019). Furthermore, ML models can capture watershed similarities by providing satisfactory results even for the catchments that were not used for the training of those models. This implies the potential of ML models in evolving catchment

scale theories in which the traditional models were unable to do so well (Nearing *et al.* 2020).

Theory-based models are frequently admired by the community due to their interpretability which may lead to a better understanding of catchment dynamics. The simple application of data-driven techniques may produce models with high prediction accuracy yet with meaningless model configurations which do not satisfy basic hydrological understanding and may have serious complications with the explanation. The black-box nature of data science models has been criticized, leading to a few of their applications and hindered success in many scientific fields. Beven (2020) highlights the importance of the interpretability of DL models and suggests a more direct incorporation of process information into such models. Furthermore, he points out that ML models should also need to pay attention to similar issues associated with traditional modelling approaches like data and parameter uncertainties and equifinality.

Nearing *et al.* (2020) contend that there is a danger for the hydrologic community in not recognizing the potential of ML offers for the future of hydrological modelling. Furthermore, the authors reject the most common criticism on ML models (the lack of explainability) by stating that the adequacy of process representation in theory-based models can be questioned due to the poor prediction accuracy. In summary, despite having a huge potential, the modern ML capabilities have not been thoroughly tested in hydrological modelling where there is an expectation that even distributed hydrological models are to be developed principally on ML in near future. Even though the new data science techniques, such as DL, have become indispensable tools in many disciplines, their applications in water resources remain quite limited (Shen *et al.* 2018).

### **Lumped models vs. distributed models vs. semi-distributed models**

Hydrological models can be broadly categorized into lumped and distributed models based on whether a model considers the spatial heterogeneity of watershed properties (e.g. geology, topography, land use) and climate variables (e.g. rainfall distribution). Lumped models assume spatially uniform catchment characteristics and use catchment

average climate variable values (e.g. precipitation, temperature, evaporation) as model inputs. In contrast, distributed models incorporate the spatial variability in watershed properties and climate variables into the modelling process. Model parsimony due to relatively few model parameters, ease and simplicity of use have made lumped modelling a popular hydrological modelling approach.

However, the meaningfulness of lumped values may deteriorate, especially when the catchment size increases, and hence, the lumped representation may not be correctly corresponding to the actual physical reality of the catchment. A lumped model may still produce accurate results for a large catchment where the spatial heterogeneities are significant, but the inferences made through the model may not be reasonable or realistic.

The so-called uniqueness of the place has been identified as one of the crucial factors in hydrological modelling (Beven 2020). One way of addressing this phenomenon is using distributed models as they consider the spatial variabilities in their modelling process. On top of that, distributed models are required in situations where the discharge forecasting is required at multiple points within the catchment, and the impacts of the change of land use patterns within the watershed are to be tested, and critical source detections are required for contaminant or sediment transport modelling (Fenicia *et al.* 2016).

Two subcategories can be identified within distributed modelling: fully distributed and semi-distributed models. Fully distributed models discretize the watershed into regular or irregular grids and use small-scale physics to model the fluxes through the spatial elements. In the beginning, its usage was greatly limited due to computation demand and intensive data requirements. However, researchers believed that more data about the catchment properties and climate variables would be available with the advancement of technology and invested heavily on developing fully distributed models. Until today, however, the distributed models have not reached the expected outcome. Distributed models suffer from the consequences of equifinality and a large number of model parameters often lead to overparameterization. A comprehensive review on applications, challenges and future trends of fully distributed modelling in hydrology is presented in Fatichi *et al.* (2016).

In contrast, semi-distributed models represent the catchment dynamics using a network of separate conceptual models assigned to functionally distinguishable land segments (Boyle *et al.* 2001). The functionally distinguishable land segments can either be subcatchments (e.g. Boyle *et al.* 2001) or Hydrological Response Units (HRUs) (e.g. Fencia *et al.* 2016). There are no interconnections or dependencies among the models in the network, and the total model output consists of the routed sum of the individual model responses of each spatial element. This together with the use of conceptual models instead of small-scale physics makes this approach more parsimonious and several orders less complex than fully distributed models. Moreover, semi-distributed models do not suffer much from equifinality and overparameterization as in fully distributed models.

## GP APPLICATIONS IN WATER RESOURCES ENGINEERING

Over the past three decades, ML has evolved into an irreplaceable tool in many commercial and scientific disciplines, including water resources engineering. The approach outperforms the traditional approaches in many applications, such as autonomous driving, language translation, character recognition and object tracking (Karpatne *et al.* 2017). ML techniques received attention among hydrologists to a great extent during the last 20 years. The majority of ML methods used in hydrological modelling can be categorized into five groups (Yaseen *et al.* 2015): (i) Artificial Neural Networks (ANNs; Minns & Hall 1996), (ii) Fuzzy set (Lohani *et al.* 2014), (iii) Support Vector Machines (SVMs; Yu *et al.* 2004), (iv) Evolutionary Computation (EC; Babovic & Keijzer 2002) and (v) Wavelet-Artificial Intelligence models (W-AI; Moosavi *et al.* 2013). The selection of an ML technique for a particular application should be based on the objectives of the study as each of these methods has its pros and cons. A comprehensive review of different ML techniques and their applications in water resources engineering is beyond the scope of this paper. Instead, interested readers are directed to review papers by Govindaraju (2000), Yaseen *et al.* (2015), Mehr *et al.* (2018) and the textbook by Hsieh (2009).

GP (Koza 1992) is an evolutionary computation technique inspired by the basic principles of Darwin's theory of evolution. GP is capable of automatic generation of computer programs and falls under the supervised ML category. The ability to generate the input-output relationships as explicit symbolic mathematical expressions differentiates GP from the rest of ML techniques. Therefore, GP is often regarded as a grey-box data-driven technique rather than as a black-box data-driven technique, such as ANNs or SVMs. Moreover, conceptual simplicity, the ability of parallel computing and the capability of obtaining the near-global or global solution make GP a powerful ML technique. There are several variants of GP-like Monolithic GP (MGP), Multi-Gene GP (MGGP), Gene Expression Programming (GEP), Linear GP (LGP) and Grammar-based GP (GGP) (Mehr *et al.* 2018). Despite variants, the fundamental operations are much the same. The GP algorithm starts with a randomly generated initial set of candidate solutions for the task at hand. It uses mathematical functions, physical or random constants and independent variables as its building blocks to generate candidate solutions. Then, the output of each candidate solution (GP individual) is evaluated against the labelled instances using a so-called objective function. Next, the genetic operators, such as crossover and mutation, are applied to GP individuals of the current generation (parent population) to generate offsprings for the next generation. The selection mechanism used to pick suitable parents for breeding ensures that more fit individuals get higher chances of selection. This process continues while the GP algorithm minimizes the difference between labelled instances and modelled values until it reaches the termination criteria (execution time, desired level of accuracy or a maximum number of generations) (Babovic & Keijzer 2000).

The earliest GP application in water resources engineering was reported in the late 1990s (Babovic 1996). Since then GP has been used in many research directions in water resources engineering, such as rainfall-runoff modelling (Babovic & Keijzer 2002; Havlicek *et al.* 2013; Babovic *et al.* 2020), streamflow prediction (Meshgi *et al.* 2014; Karimi *et al.* 2016), water quality modelling (Savic & Khu 2005), groundwater modelling (Fallah-Mehdipour *et al.* 2013; Datta *et al.* 2014), reservoir management (Giuliani *et al.* 2015), sediment transport (Babovic 2000; Safari &

Mehr 2018), climate variables and soil properties modelling (Bautu & Bautu 2006; Elshorbagy & El-Baroudy 2009). A comprehensive review on GP applications in water resources engineering is presented in Mehr *et al.* (2018), whereas Fallah-Mehdipour & Haddad (2015) presents the applications of GP in hydrological modelling.

In the context of rainfall-runoff modelling, GP has been reported to be more accurate than the other ML techniques, such as ANN in terms of prediction efficiencies (Havlicek *et al.* 2013). However, it is worthwhile to note that, in many of the GP utilizations in water resources engineering so far, the GP algorithm is used as a short-term forecasting mechanism by paying more attention towards the prediction power and less attention to the resultant input–output relationship. Yet, the ability to produce an explicit mathematical relationship between input and output variables has made GP so different from other ML techniques. Recent studies, such as Chadalawada *et al.* (2017, 2020) and Herath *et al.* (2020), explore this potential of GP and use it as a rainfall-runoff model induction toolkit to induce interpretable models with good predictive powers.

### Physics informed GP

One effective way to address the main criticism of ML models (lack of interpretability) would be to use the existing hydrological knowledge to guide ML models (Babovic & Keijzer 2002; Babovic 2009). Presently, blending the existing body of knowledge with learning algorithms has been identified as a novel modelling paradigm in ML. Despite the different taxonomy used by different researchers, such as TGDS (Karpatne *et al.* 2017), physics informed machine learning (Physics Informed Machine Learning Conference 2016; Chadalawada *et al.* 2020) and AI-neuroscience (Voosen 2017), this modelling paradigm aims to enhance the explainability of ML models by inducing more generalizable and physically consistent models. Karpatne *et al.* (2017) suggest five ways of combining scientific knowledge with data science: (i) theory-guided design of data science models, (ii) theory-guided learning of data science models, (iii) theory-guided refinement of data science outputs, (iv) learning hybrid models of theory and data science and (v) augmenting theory-based models using data science. A typical theory-guided machine learning approach may

follow one or more above-named strategies to couple scientific knowledge with learning algorithms.

McGovern *et al.* (2019) highlight that an increasing trend can be observed by using TGDS models in water resources engineering applications including optimal process-oriented hydroclimatic model building and multiple-source data fusion for hydrologic process simulation (e.g. Snauffer *et al.* 2018; Solander *et al.* 2019; Chadalawada *et al.* 2020; Herath *et al.* 2020) in contrast to limited such applications in past (e.g. Cannon & Mckendry 2002; Keijzer *et al.* 2001; Keijzer & Babovic 2002; Fleming 2007). Although there are efforts in nearly all machine learning techniques to integrate existing knowledge into the basic frameworks, in the sequel, we will focus our discussion on efforts in GP. For a complete discussion on theory-guided machine learning applications in water resources engineering, refer to Karpatne *et al.* (2017).

Even though the TGDS was only recently identified as a novel modelling paradigm in the context of GP, there were efforts over the past 20 years to couple the hydrological knowledge into the basic GP framework to induce more physically reliable hydrological models. Babovic & Keijzer (2000, 2002) and Keijzer & Babovic (2002) augmented the basic GP algorithm by introducing preferential bias and declarative bias into the model building process of GP to induce more physically sound and consistent models that consist of dimensionally accurate equations with minimal physical violations. It is reported that the dimensionally aware GP approach performs better than the traditional GP by achieving faster convergence and parsimonious model equations. In another study (Babovic *et al.* 2000), the same approach was used to induce hydraulic equations from the data. Babovic (2009) incorporated high-level concepts with the GP algorithm to evolve sediment transport process equations and noticed that the resulted equations perform equally or better than the equations derived by human experts. GP was used to analyse the configurations of hydrological models by Selle & Muttill (2011) to gain insights about the dominant hydrological processes in environmental systems. Basic hydrological concepts were incorporated with the GP algorithm as special functions (e.g. a simple reservoir function, simple moving average function, delay operator and cumulative sum operator) in Havlicek *et al.* (2013), to improve the runoff forecasting capabilities of GP. They found that the GP with special functions performs better than the traditional GP

and ANNs in terms of prediction efficiencies. Chadalawada *et al.* (2017) used GP to identify the optimal TANK model configuration (Sugawara 1979) for the catchment of interest by optimizing the number of reservoir blocks (tanks), connections and outflows based on field measurements. In our prior work (Chadalawada *et al.* 2020; Herath *et al.* 2020), an automatic lumped and semi-distributed conceptual rainfall-runoff model induction toolkit was developed using GP, and the building blocks available in two modular modelling frameworks (SUPERFLEX and FUSE) were used as the aspects of hydrological insights.

## GP FOR HYDROLOGICAL FORECASTING

In the traditional setting, when GP is used for hydrological forecasting, the terminal set of GP algorithm consists of past and current states of meteorological variables and randomly generated constants. The function set consists of basic mathematical functions. The GP algorithm uses these building blocks to generate its solution set for predicting catchment response into the future.

### Runoff forecasting

For example, to produce one-day ( $Q_{t+1}$ ) and five-day ( $Q_{t+5}$ ) runoff forecasts, one would use precipitation ( $P_t$ ), evaporation ( $E_t$ ) and antecedent runoff ( $Q_t$ ) as the input variables. Furthermore, lagged precipitation ( $P_{t-lag}$ ) and evaporation ( $E_{t-lag}$ ) vectors up to a certain period (lag) may also be used to incorporate precipitation and evaporation history into the forecasting process (in the current study, lag = 5 days). Although the runoff history can also be included in the forecasting process, in this study, no lagged runoff vectors are used as the forecasted runoff is largely correlated with the last observed runoff value ( $Q_t$ ). The expected relationship between the forecasted runoff and input variables can be captured as in Equation (1). GP may create its candidate equations either using one or more input vectors along with the random constants and mathematical functions. GP settings used for the runoff forecasts are given in Table 1.

$$Q_{t+1} = Q_{t+5} = f(P_t, \dots, P_{t-5}, E_t, \dots, E_{t-5}, Q_t) \quad (1)$$

**Table 1** | GP settings used for runoff forecasting

Option	Setting
Independent runs	100
Individuals in one generation	500
Maximum generations	100
Method of initialization	Ramped half and half
Function set	+, −, /, *, sin, tan, ln, exp, sqrt
Constant range – Min/Max	−10/10
Tree depth – Initial/Max	3/7
Method of selection	Tournament (Size = 4)
Probability of crossover	0.7
Mutation probabilities Constant//Tree/Node/ Separation	0.5/0.5/0.3/0.3

### Case studies

Two catchments located in the Eastern United States, namely Sipsey Fork catchment, near Grayson, Alabama and Red Creek catchment, near Vestry, Mississippi, were selected for the runoff forecasting using traditional GP. The catchment characteristics of both watersheds are summarized in Table 2. Precipitation ( $P$ ), Streamflow ( $Q$ ) and Evaporation ( $E$ ) data for 11 years from 1 January 2004 to

**Table 2** | Catchment details

Parameter	Sipsey Fork	Red Creek
Basin area (km <sup>2</sup> )	239	1,144
Outlet coordinates	34.28538°, −87.39891°	30.73611°, −88.78111°
Sub catchment area (%)	–	Sub1: 39%, Sub2: 37.9%, Sub3: 23.1%
Annual mean discharge (mm/day)	1.610	1.755
Annual mean potential evaporation (mm/day)	3.292	3.689
Annual mean temperature (°C)	16.3	19.57
Annual mean precipitation (mm/day)	3.796	4.201
Mean slope (m/km)	12.27	5.85
Fraction of forest	0.84	0.89

31 December 2014 were used for model training (1 January 2004–31 December 2009) and model testing (1 January 2010–31 December 2014).

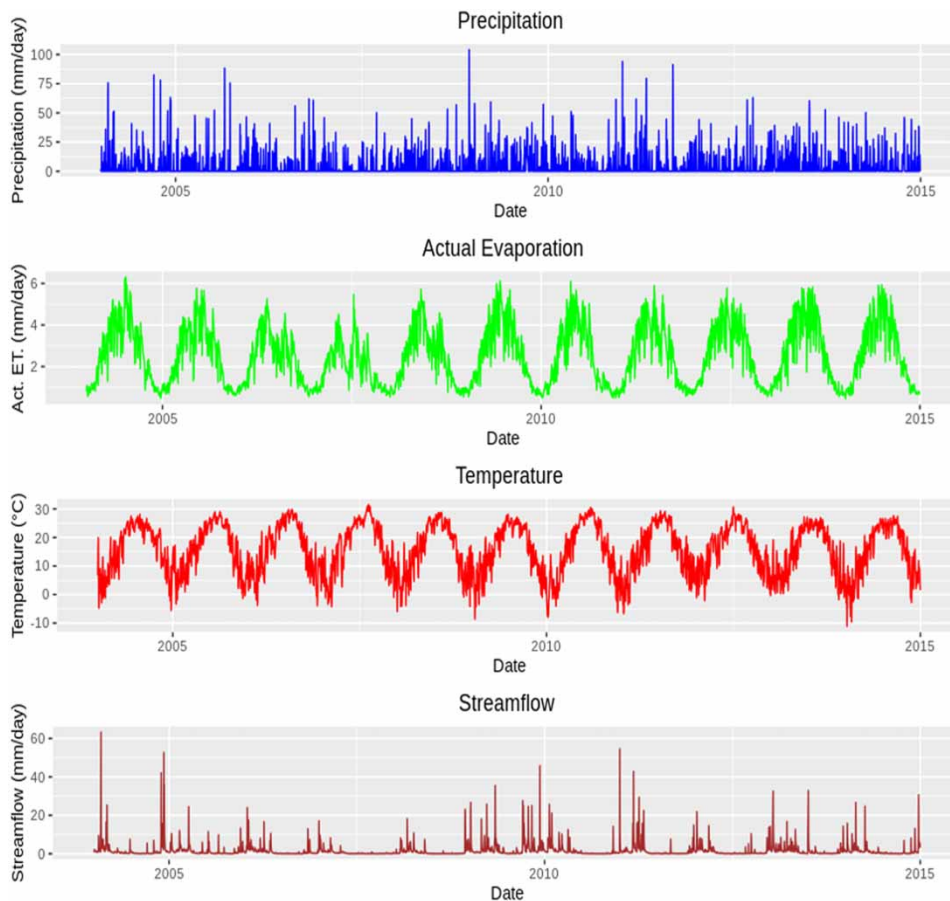
The catchment average daily data of  $P$ ,  $Q$  and  $E$  of both the catchments were downloaded from CAMELS dataset (Newman *et al.* 2015). Time-series plots of forcing terms and streamflow data of the Sipsey Fork catchment and the Red Creek catchment are shown in Figures 1 and 2, respectively.

### Objective functions

The Nash–Sutcliffe efficiency (NSE; Nash & Sutcliffe 1970) is used as the objective function to measure the performance of GP individuals. The GP individual (equation describing the input–output relationship) with the highest NSE value (optimum at 1) for the training period is selected as the optimal runoff forecasting equation for the catchment of interest. Then, the performance of the optimal equation is evaluated

on the testing period. For the comparison purpose, the training and testing performances of the optimal equation on three other objective criteria, namely Volumetric Efficiency (VE; Criss & Winston 2008), Kling–Gupta Efficiency (KGE; Gupta *et al.* 2009) and log Nash–Sutcliffe Efficiency (logNSE; Krause *et al.* 2005) are evaluated. Equations of the four objective functions are given below ( $N$  is the time steps,  $Q_{ot}$  is the observed streamflow,  $Q_{st}$  is the simulated streamflow,  $r$  is the linear correlation coefficient,  $\alpha = \sigma_s/\sigma_o$ ,  $\beta = \mu_s/\mu_o$ ,  $\sigma$  is the standard deviation,  $\mu$  is the mean,  $\overline{Q_{ot}}$  is the mean of observed discharge values and  $\ln$  is the natural logarithm). They are responsive to contrasting flow segments of simulated and observed hydrographs.

$$NSE = 1 - \frac{\sum_{t=1}^N (Q_{ot} - Q_{st})^2}{\sum_{t=1}^N (Q_{ot} - \overline{Q_{ot}})^2} \quad (2)$$



**Figure 1** | Forcing terms and streamflow data of the Sipsey Fork catchment.



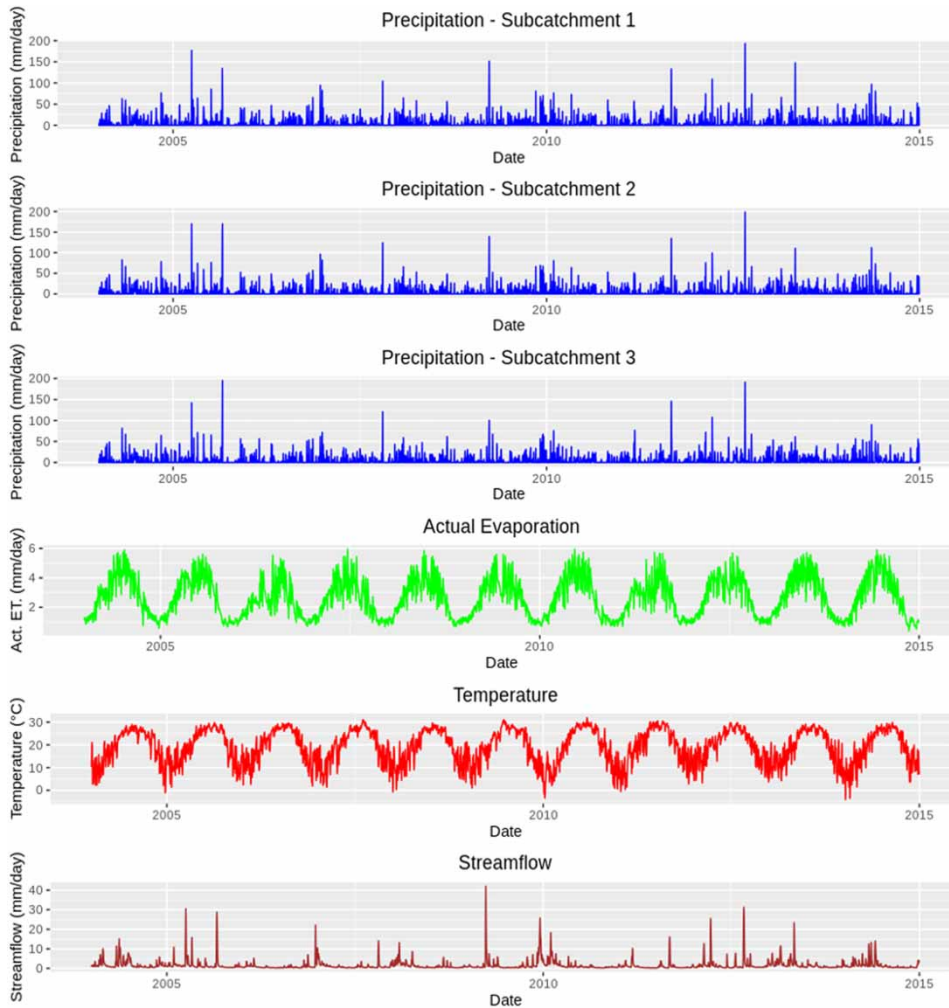


Figure 2 | Forcing terms and streamflow data of the Red Creek catchment.

$$VE = 1 - \frac{\left| \sum_{t=1}^N (Q_{ot} - Q_{st}) \right|}{\sum_{t=1}^N Q_{ot}} \quad (3)$$

$$KGE = 1 - \sqrt{(r-1)^2 + (\alpha-1)^2 + (\beta-1)^2} \quad (4)$$

$$\log NSE = 1 - \frac{\sum_{t=1}^N (\ln Q_{ot} - \ln Q_{st})^2}{\sum_{t=1}^N (\ln Q_{ot} - \ln \overline{Q_{ot}})^2} \quad (5)$$

## Results

Four equations were derived by using GP as a forecasting tool for one-day and five-day runoff prediction of the two

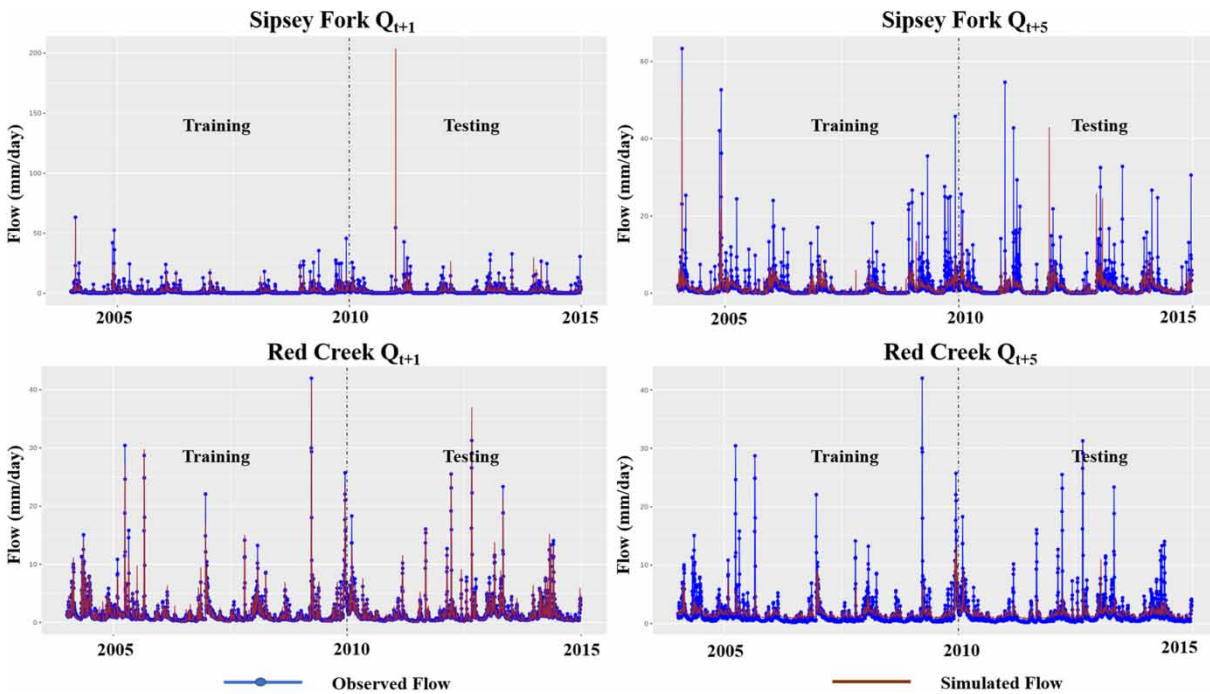
catchments. Equations (6) and (7) are for the Sipsy Fork catchment and Equations (8) and (9) are for the Red Creek catchment. The performance matrices of the four equations are given in Table 3. Figure 3 demonstrates the forecasted hydrographs with the observed hydrographs of the catchments.

$$Q_{t+1} = \frac{Q_t}{\sqrt{P_{t-4} + \frac{3.348 + E_{t-2}}{Q_t} + \frac{2 \times P_{t-5} + P_{t-1}}{1.731 + 2 \times P_{t-5} + P_{t-4}}} + \sqrt{Q_t}} \quad (6)$$

$$Q_{t+5} = \max \left( 0, \frac{\ln(Q_t) - \frac{0.558}{Q_t} + \frac{0.037}{\ln(E_{t-3})} - \ln(E_t) + 2.712}{E_{t-4}} \right) \quad (7)$$

**Table 3** | Performance matrices of runoff forecasts

	$Q_{t+1}$				$Q_{t+5}$			
	VE	KGE	NSE	logNSE	VE	KGE	NSE	logNSE
<i>Sipsey Fork</i>								
Training	0.43	0.54	0.45	0.69	0.19	0.31	0.26	0.26
Testing	0.35	0.12	-1.42	0.68	0.14	0.09	-0.09	0.13
<i>Red Creek</i>								
Training	0.80	0.93	0.92	0.91	0.28	0.13	0.15	0.17
Testing	0.81	0.95	0.92	0.91	0.26	-0.03	0.06	0.13

**Figure 3** | Forecasted hydrographs with the observed hydrographs of the Sipsey Fork and Red Creek catchments.

$$Q_{t+1} = \max \left( 0, 0.786 \times Q_t + 0.081 \times P_t - 0.03 \times \left( \frac{P_t}{Q_t} \right) + 0.005 \times Q_t (P_{t-1} - Q_t) \right) \quad (8)$$

$$Q_{t+5} = \sqrt{Q_t} + \exp(\sin \sqrt{Q_t} - 1.504) + \exp(5.373 - E_t - \exp(5.234 - Q_t) - \exp(E_t)) \quad (9)$$

Following observations can be made through the forecasting results of the two catchments.

- One-day runoff forecasts of the Red Creek catchment achieve high-efficiency values for both the training and

testing periods under all four objective functions. This demonstrates the ability of GP to achieve high-efficiency values in runoff forecasting. Furthermore, the optimal equation (Equation (8)) shows no signs of overfitting due to its consistent performance across both the training and testing periods.

- As it can be seen with observed hydrographs, the runoff signature of the Sipsey Fork catchment is relatively more complex than that of the Red Creek catchment. In this context, GP with only mathematical functions is unable to achieve satisfactory performance in terms of the training NSE value. Additionally, the optimal

equation for one-day forecasts of the Sipsy Fork catchment (Equation (6)) is overfitted to its training data as the NSE value falls below zero for the testing period. This highlights the importance of cautious usage of traditional GP forecasts when they are used with out of sample values.

- As per the efficiency values, one-day forecasts of both catchments perform significantly better than five-day forecasts. Therefore, the performance of GP as a forecasting tool deteriorates significantly as the lead time increases.
- The optimal equations provide little or no knowledge about the underlying physical phenomena of runoff generation. Furthermore, the equations tend to become lengthier with insignificant components. For example, the third term of Equation (9) ( $\exp(5.373 - E_t) - \exp(5.234 - Q_t) - \exp(E_t)$ ) results in insignificant contribution towards the total forecasted runoff.

## GP FOR HYDROLOGICAL MODELLING

In this section, GP is used for automatic rainfall-runoff model induction. In contrast to runoff forecasting, GP here maps the input forcing variables at the time  $t$  to predict the runoff at the time  $t$  which involves no lead time.

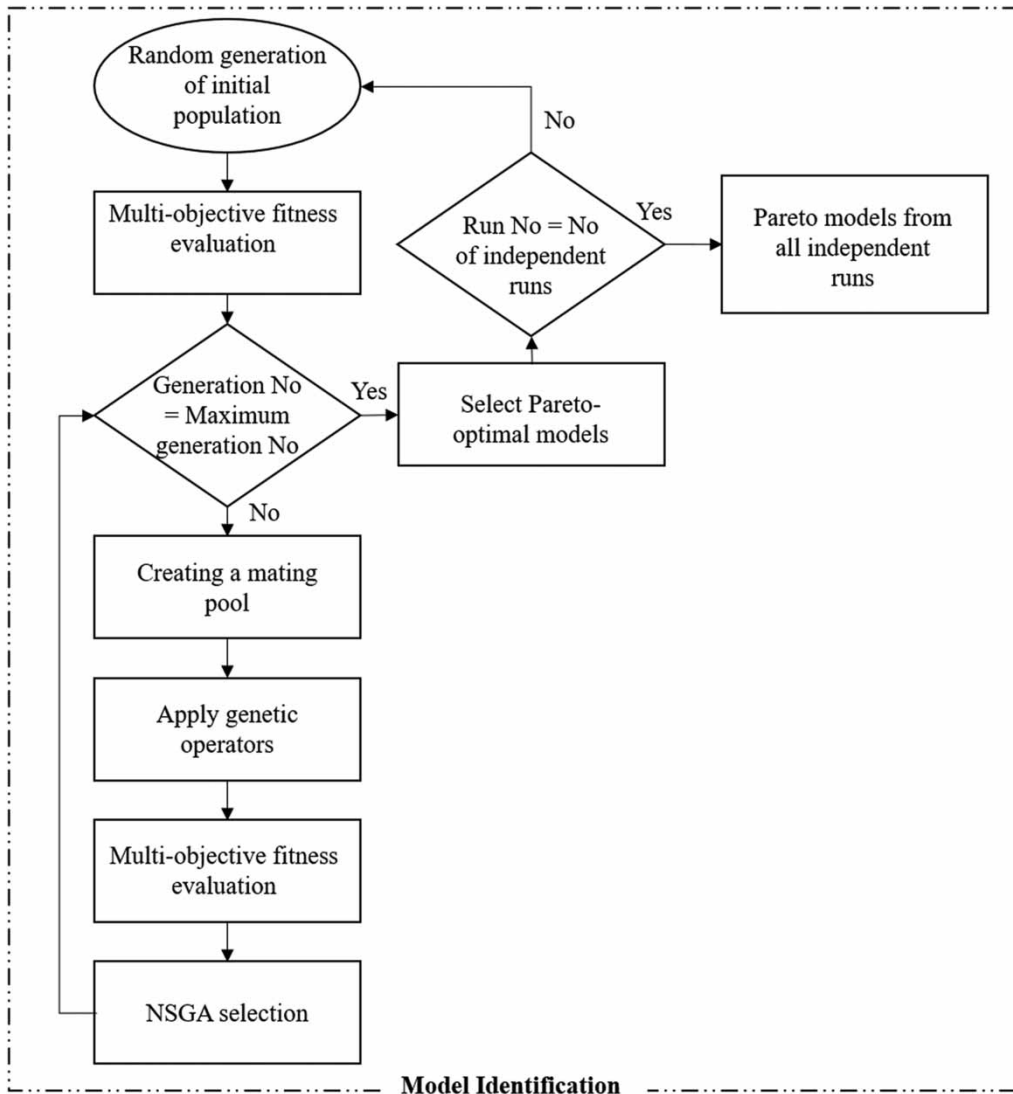
### ML-RR-MI AND MIKA-SHA

In the present study, ML-RR-MI (Chadalawada *et al.* 2020) and MIKA-SHA (Herath *et al.* 2020) toolkits are used to identify the optimal rainfall-runoff model configurations for catchments of interest. Both toolkits are based on GP and classified as hydrologically informed ML techniques where existing hydrological knowledge is incorporated into the learning algorithm (as special functions) to induce physically sound and consistent models. ML-RR-MI is used for lumped modelling and MIKA-SHA is used for distributed modelling. Building blocks used in the GP frameworks of both toolkits represent the elements of available hydrological knowledge. In their earlier applications, the model

components of FUSE and SUPERFLEX flexible modelling frameworks were used as model building blocks. The most important feature of these two model induction frameworks is that they can be coupled with any internally coherent collection of building blocks. Hence, in this paper, both ML-RR-MI and MIKA-SHA are using different model inventory based on the Sugawara TANK model template. As per the taxonomy defined in Karpatne *et al.* (2017), our approach can be categorized as a hybrid TGDS approach.

Both toolkits are primarily coded in R programming language (R Core Team 2013) based on the canonical GP approach proposed in Havlicek *et al.* (2013). The multi-objective optimization framework of the toolkits is based on Non-dominated Sorting Genetic Algorithm-II (NSGA-II; Deb *et al.* 2002). Parallel computation has been used at the fitness computation stage of GP individuals to reduce the overall computation time. However, the genetic operators are applied to the whole generation in series (using one core) to have more diversity among breeding individuals. Both toolkits consist of four major stages. They are data pre-processing stage, model identification stage, model selection stage and uncertainty and sensitivity analysis stage. Each step is briefly described in the following section, while for a detailed explanation refer to Chadalawada *et al.* (2020) and Herath *et al.* (2020).

The data pre-processing stage involves the quality control of independent (forcing terms) and dependent variable (streamflow) data. More steps are required with MIKA-SHA, such as watershed delineation, and are discussed in the next section. The core of both toolkits is the GP-based model identification stage where both the model structure and model parameters of GP individuals are optimized simultaneously. The workflow diagram of the model identification stage is given in Figure 4. The model identification stage starts with randomly generated models (GP individuals) to capture the runoff signature of the catchment of interest by using independent variables, random constants, mathematical functions and special functions. Then, the fitness of each individual is assessed based on the multi-objective criterion used. Both toolkits are equipped with an objective function library which consists of most of the performance measures in practice. Candidate models are evolved through generations as the genetic operators are applied. Optimization terminates once the number of



**Figure 4** | Workflow diagram of the model identification stage of ML-RR-MI and MIKA-SHA.

generations reaches the maximum number of generations. The whole process is repeated to a user-defined number of iterations (independent runs) to have more coverage of the solution space. The Pareto optimality concept used here results in a group of model structures that are not dominated by each other on the user-defined performance measures as the final output of the model identification stage.

The model selection stage involves the optimal model selection out of the Pareto-optimal models identified in the model identification stage. Fitness function values of the same multi-objective criterion over the validation period are calculated and Pareto-optimal models are re-identified

based on both the calibration and validation performance values. Then, the standardized signature index sum value (SIS; [Ley et al. 2016](#)) of each model is calculated and only the models with negative SIS values are passed to the next selection step (the SIS value is a relative performance measure which evaluates how well a particular model approximates the measured flow duration curve (FDC) relative to the other models. The more negative the value the better the performance). Then, the models with unique model structures are identified (known as competitive models). Finally, each competitive model is ranked based on three relative performance indicators (model parsimony

based on the number of reservoir units/model parameters, dynamic time warping distance and cross-sample entropy). The model with the least sum-up rank is selected as the optimal model for the watershed of interest.

The uncertainty and sensitivity analysis of both toolkits is based on generalized likelihood uncertainty estimation (GLUE; Beven & Binley 1992). The model parameter values of the selected optimal model are changed uniformly across their parameter ranges and a user-defined number of satisfactory model parameter sets are identified (known as behavioural models and the performance is evaluated using user-specified objective criteria). Then, the percentage of the observed values that exist within the 90% confidence interval bands are used as a gauge of the parameter uncertainty of the optimal model. Sensitivity scatter plots are drawn for each model parameter and their shape is used to identify the model sensitive parameters.

### Special functions

The elements of hydrological knowledge are incorporated through the addition of special functions to the GP function set. The special functions used in the current study are inspired by the Sugawara TANK model template elements and semi-distributed modelling concepts.

### Sugawara TANK model

Originally, the TANK model (Sugawara 1979) was developed to represent the hydrological processes in small humid catchments. In the TANK framework, several reservoir units (maximum four reservoirs) are arranged vertically to conceptualize the different soil layers in a catchment (hereinafter referred to as a series TANK model). Later, a parallel version of the TANK model (hereinafter referred to as a parallel TANK model) was proposed for large non-humid catchments. The parallel TANK model consists of several branches (maximum four branches) of series TANK models. Each branch represents a different zone of the watershed. The outermost zone represents hill/upstream side and the innermost represents the river/downstream side of the catchment basin. The parallel TANK model architecture with the maximum possible reservoir units (four branches each with four reservoirs) is shown in Figure 5. A typical

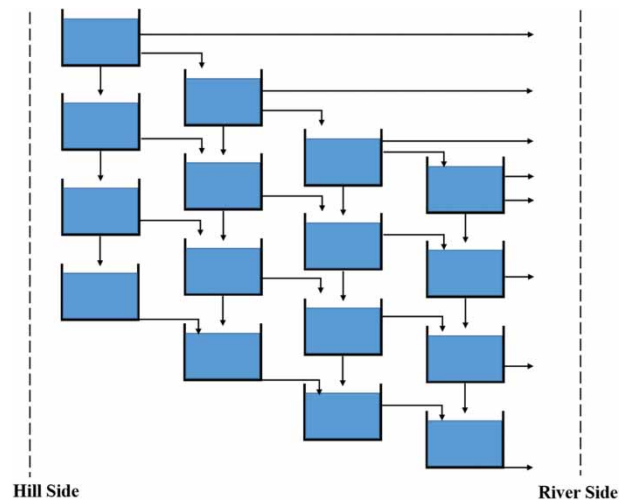


Figure 5 | Sugawara parallel TANK template.

reservoir unit in a TANK model consists of at least one side flow and one bottom flow (bottom flow is zero for the last reservoir in each branch). Side flows are used to represent surface runoff, interflow, sub baseflow and baseflow, while the bottom flows are used to represent interception, infiltration and deep percolation. By considering the uniqueness of the area of interest, one can test many hypotheses by altering the number of branches in the parallel TANK model configuration and by varying the number of reservoir units in each branch. Hence, the parallel TANK model provides more granularity in the model building than the other conceptual models with a fixed structure. Computational and conceptual simplicity, higher predictive power and easy interpretability have made the TANK model a widely used rainfall-runoff model (Song *et al.* 2017). More details and applications of the TANK model can be found in Nie *et al.* (2017) and Chadalawada *et al.* (2017).

### RB function

RB function represents a single reservoir in a TANK model configuration. As shown in Equation (10), RB has nine function arguments.

$$Q = \text{RB}(\text{RI}, h_1, a_1, h_2, a_2, b, L_1, L_2, S) \quad (10)$$

where  $Q$  is the discharge components (two side flows and one bottom flow) of the reservoir (mm/day); RI is all input

variables, such as precipitation, evaporation and flows from the other reservoirs (mm/day);  $h_1, h_2$  are the height of side-outlet one and two (mm).  $h_2 = 0$  for the bottom reservoir in each branch;  $a_1, a_2$  and  $b$  is the linear discharge coefficients of the two side flows and the bottom flow.  $a_1 > 0$  for the top reservoir in each branch and  $a_1 = 0$  elsewhere.  $a_1 + a_2 + b \leq 1$  for every reservoir;  $L_1$  and  $L_2$  is the time delay in lag functions of two side flows (days);  $S$  is the initial storage of the reservoir (mm).

### RJOIN function

RJOIN function (Equation (11)) connects the individual reservoirs and calculates the total discharge from a one parallel TANK model configuration. RJOIN function can have a maximum of four functional arguments to represent the four possible branches of a parallel TANK model configuration.

$$Q = \text{RJOIN}(\text{RB}(\dots), \text{RB}(\dots), \dots) \quad (11)$$

### DISTRIBUTED function

DISTRIBUTED function (Equation (12)) is used to induce semi-distributed rainfall-runoff models. It calculates the total catchment response according to the semi-distributed modelling paradigm. The length of the functional arguments depends on the number of HRUs used in the study. The last two functional arguments in DISTRIBUTED function ( $L_{\text{HRU}}$  and  $L_{\text{sub}}$ ) are the time delays of lag functions used to route the HRU outflows and subcatchment outflows.

$$Q = \text{DISTRIBUTED}(\text{RJOIN}(\dots), \text{RJOIN}(\dots), \dots, L_{\text{HRU}}, L_{\text{sub}}) \quad (12)$$

Equations (13)–(15) are used to calculate the side flows, bottom flow and evaporation from a reservoir element, respectively (used in RB function). In these equations,  $q_{k,t}$  is the flow of side-outlet  $k$  at time  $t$ ,  $a_k$  and  $b$  are the outlet coefficients of side-outlet  $k$  and bottom outlet,  $h_k$  is the height of side-outlet  $k$ ,  $i_t$  is the bottom flow,  $E_{i,t}$  is the evaporation from reservoir  $i$  at time  $t$ ,  $E_t$  is the actual evaporation

at time  $t$  and  $S_{i,t}$  is the storage of reservoir  $i$  at time  $t$ .

$$q_{k,t} = a_k(S_t - h_k) \quad (13)$$

$$i_t = b(S_t) \quad (14)$$

$$E_{i,t} = E_t - \sum_{j=1}^{i-1} E_{j,t} \quad \text{for } E_{i,t} < S_{i,t}; \quad E_{i,t} = S_{i,t} \quad (15)$$

for  $E_{i,t} \geq S_{i,t}$

In the current study, RB function and RJOIN function along with other basic mathematical functions are used for lumped modelling (with ML-RR-MI), whereas RB, RJOIN and DISTRIBUTED functions along with basic mathematical functions are used for semi-distributed modelling (with MIKA-SHA). The parse tree representation of RB, RJOIN and DISTRIBUTED functions are given in Figure 6. The two-parameter gamma function is used as the lag functions in RB and DISTRIBUTED functions.

### Case studies

The Sipsey Fork catchment, near Grayson, Alabama, was used for lumped modelling with ML-RR-MI, while the Red Creek catchment, near Vestry, Mississippi, was used for semi-distributed modelling with MIKA-SHA.  $P$ ,  $Q$  and  $E$  data for 11 years from 1 January 2004 to 31 December 2014 were used for model spin-up (1 January 2004–31 December 2004), model calibration (1 January 2005–31 December 2009), model validation (1 January 2010–31 December 2012) and model testing (1 January 2013–31 December 2014).

The catchment average daily data of  $P$ ,  $Q$  and  $E$  of the Sipsey Fork catchment were downloaded from CAMELS dataset (Newman et al. 2015). Watershed delineation was required for the Red Creek catchment to identify subcatchments and HRUs. In the current study, SWAT+ plugin in QGIS software (QGIS 2020) was used for the watershed delineation. The whole watershed was divided into three subcatchments (Sub 1, Sub 2 and Sub 3) and HRUs were identified based on the geology of the area. Four HRUs were identified to represent the four major soil types of the Red Creek catchment (Figure 7). The area percentages of each HRU within three

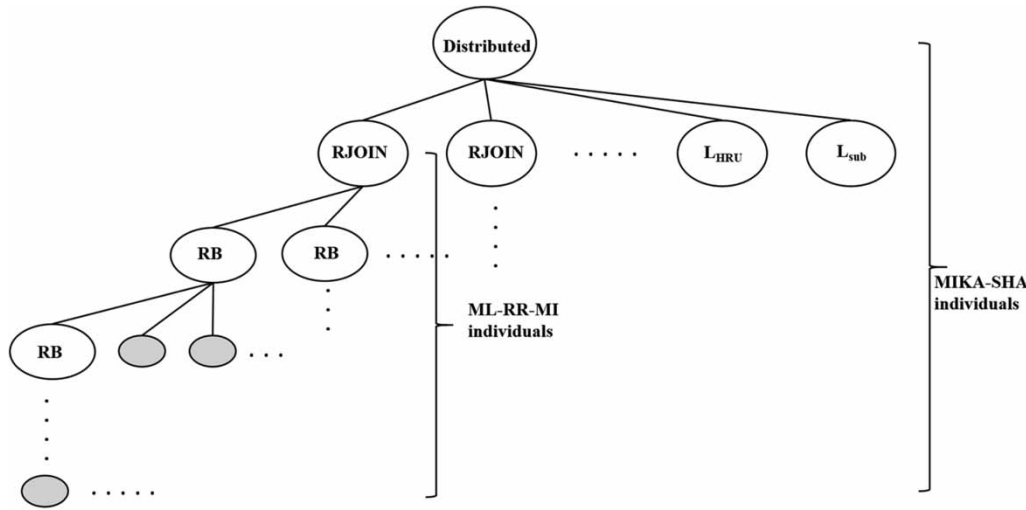


Figure 6 | Parse tree representation of the special functions.

subcatchments are given in Table 4. The spatial distribution of daily precipitation data of the Red Creek catchment was considered and lumped at the subcatchment scale (three precipitation time series for the three subcatchments). Soil and land use data (resolution 30 m × 30 m), digital elevation data (resolution 30 m × 30 m), evaporation and streamflow data (catchment average daily data) and precipitation data (daily time step,

1 km × 1 km resolution) of the Red Creek catchment were downloaded from the United States Department of Agriculture’s (USDA’s) Geospatial Data Gateway (USDA’s Geospatial Data Gateway 2020), Shuttle Radar Topography Mission (SRTM) data from the United States Geological Survey (USGS) EarthExplorer (USGS EarthExplorer 2020), CAMELS dataset (Newman et al. 2015) and Daymet dataset (Daymet 2020), respectively.

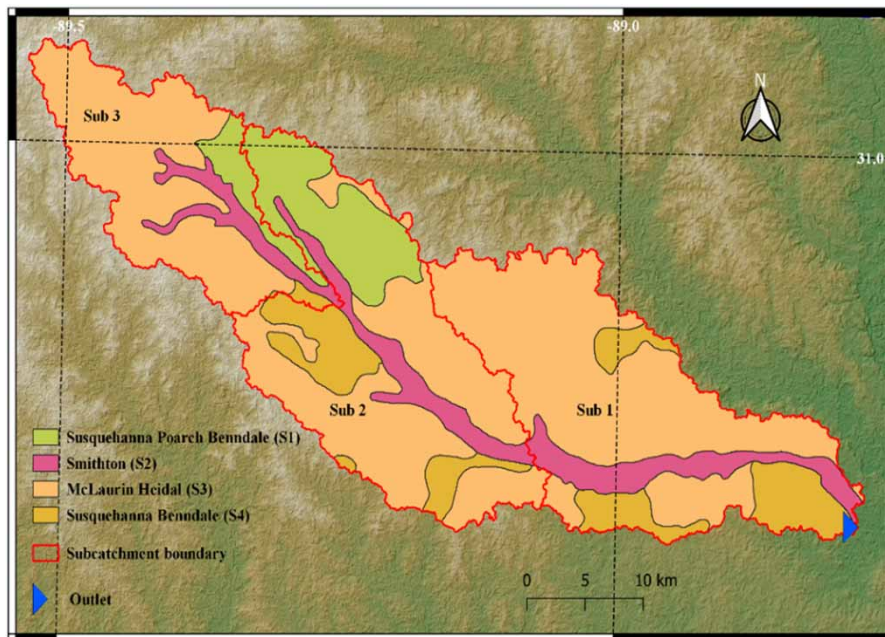


Figure 7 | Geology-based HRU classification, Red Creek catchment.

**Table 4** | Geology-based HRU details, Red Creek catchment

Subcatchment	Susquehanna Poarch Benndale (S1) (%)	Smithton (S2) (%)	McLaurin Heidal (S3) (%)	Susquehanna Benndale (S4) (%)
1	0.00	12.46	69.88	17.67
2	23.20	10.54	51.81	14.46
3	13.32	15.30	69.29	2.10

## Rainfall-runoff modelling

ML-RR-MI was used to identify an optimal lumped rainfall-runoff model for the Sipsey Fork catchment, while MIKA-SHA was used to identify an optimal semi-distributed rainfall-runoff model for the large Red Creek catchment. As the zonal representation of the parallel TANK modelling template captures the topography of the area to a certain extent, the HRUs were identified based on the geology of the Red Creek catchment. Therefore, MIKA-SHA model architectures derived by semi-distributed modelling are expected to represent the impacts of both topography and geology towards the runoff generation of the catchment. GP settings used with two model induction toolkits are summarized in Table 5. Optimal model selection was carried out as described in earlier sections, while 5,000 behavioural models (parameter sets) were identified for uncertainty analysis using NSE as the performance indicator with a threshold value of 0.6.

**Table 5** | GP settings of ML-RR-MI and MIKA-SHA

Option	Setting
Independent runs	20
Individuals in one generation	2,000
Maximum generations	50
Method of initialization	Ramped half and half
Function set	DISTRIBUTED, RJOIN, RB, +, -, /, *
Constant range – Min/Max	0/1
Tree depth – Initial/Max	Lumped: 5/6, Distributed: 6/7
Method of selection	Tournament (Size = 4)
Probability of crossover	0.7
Mutation probabilities Constant//Tree//Node/ Separation	0.5/0.5/0.3/0.3
Number of cores	20
Parallel computation level	Fitness evaluation Stage

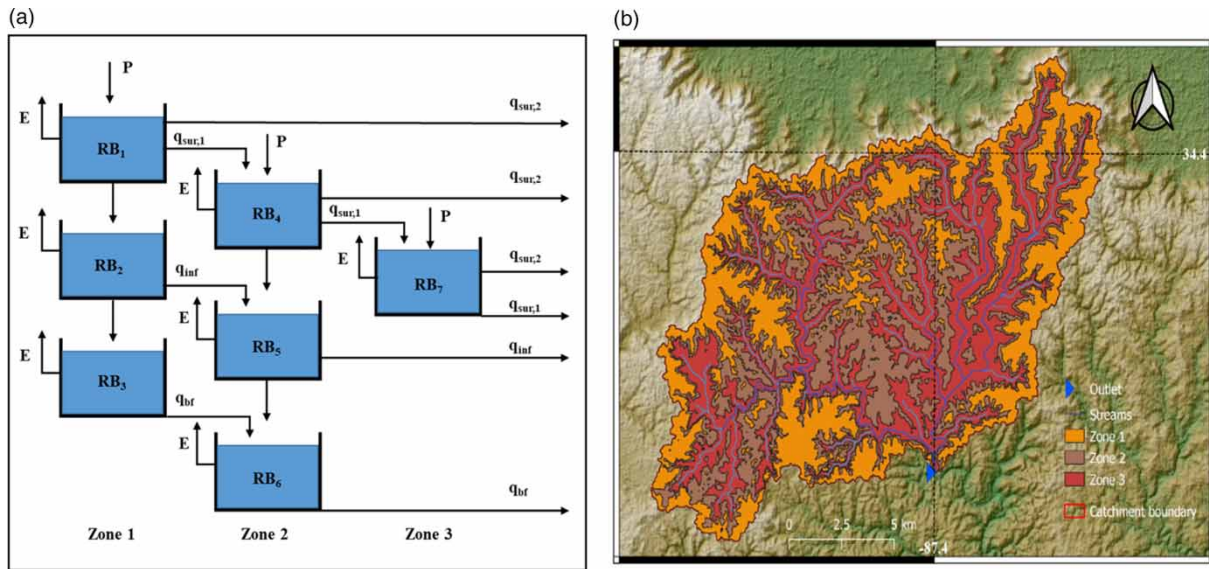
## Results

### Lumped modelling

Following the methodology described in earlier sections, ML-RR-MI identified a three-branch model configuration (Figure 8(a)) as the optimal model structure representing the watershed dynamics of the Sipsey Fork catchment. The induced model configuration divides the watershed into three equal zones (each comprises 1/3 of the catchment area). Zone 1 represents the hillside of the catchment, while Zone 3 represents the riverside. The three-zone representation of the Sipsey Fork catchment is illustrated in Figure 8(b). The first two branches have three reservoirs each and the last branch has only one reservoir. The total catchment outflow consists of four surface runoff components ( $q_{sur2}$  of RB<sub>1</sub>,  $q_{sur2}$  of RB<sub>4</sub>,  $q_{sur2}$  and  $q_{sur1}$  of RB<sub>7</sub>), one interflow component ( $q_{inf}$  of RB<sub>5</sub>) and one baseflow component ( $q_{bf}$  of RB<sub>6</sub>). Out of the total model outflow during the calibration period, 59% consists of surface flow components and 41% consists of sub-surface flow components (baseflow and interflow).

The performance matrices of the optimal model are given in Table 6. As per the efficiency values, the optimal model shows a consistent performance over calibration, validation and testing periods. Hence, it shows no sign of overfitting issues to its training data. The simulated hydrograph along with the observed hydrograph of the Sipsey Fork catchment is given in Figure 9 and a good visual match between the two hydrographs can be observed. However, on some occasions, the optimal model underestimates high discharge values. The simulated FDC (Figure 10) of the calibration period closely follows the measured FDC in all flow segments, while the simulated FDCs of the validation and testing periods tend to deviate from the observed FDC in low flow regimes. According to the uncertainty analysis, 67.1% of observed discharge values during the calibration period fall within the 90% uncertainty bounds. This high





**Figure 8** | (a) Optimal TANK model configuration. (b) Three-zone representation of the Sipsey Fork catchment (RB<sub>*i*</sub>: *i*th reservoir, P: precipitation, E: evaporation,  $q_{sur,1}$  and  $q_{sur,2}$ : surface runoff one and two,  $q_{inf}$ : interflow,  $q_{bf}$ : baseflow).

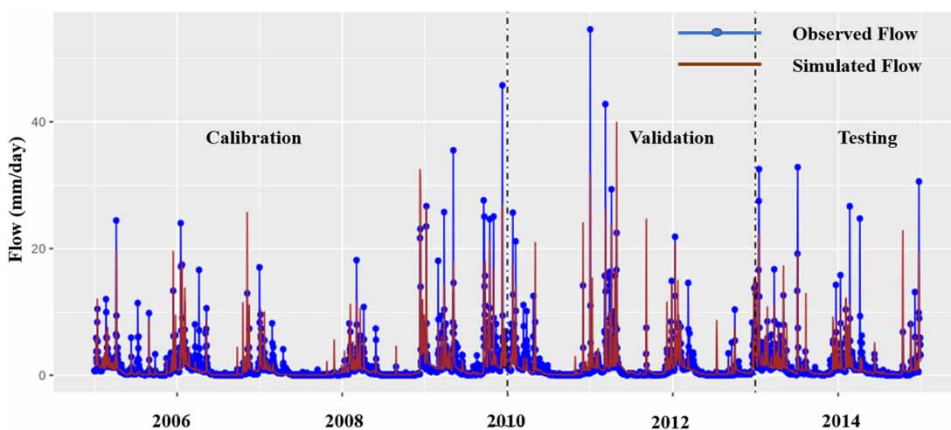
**Table 6** | Performance matrices of the ML-RR-MI optimal model – Sipsey Fork catchment

Period	VE	KGE	NSE	logNSE
Calibration	0.54	0.81	0.69	0.80
Validation	0.45	0.79	0.64	0.74
Testing	0.54	0.71	0.64	0.77

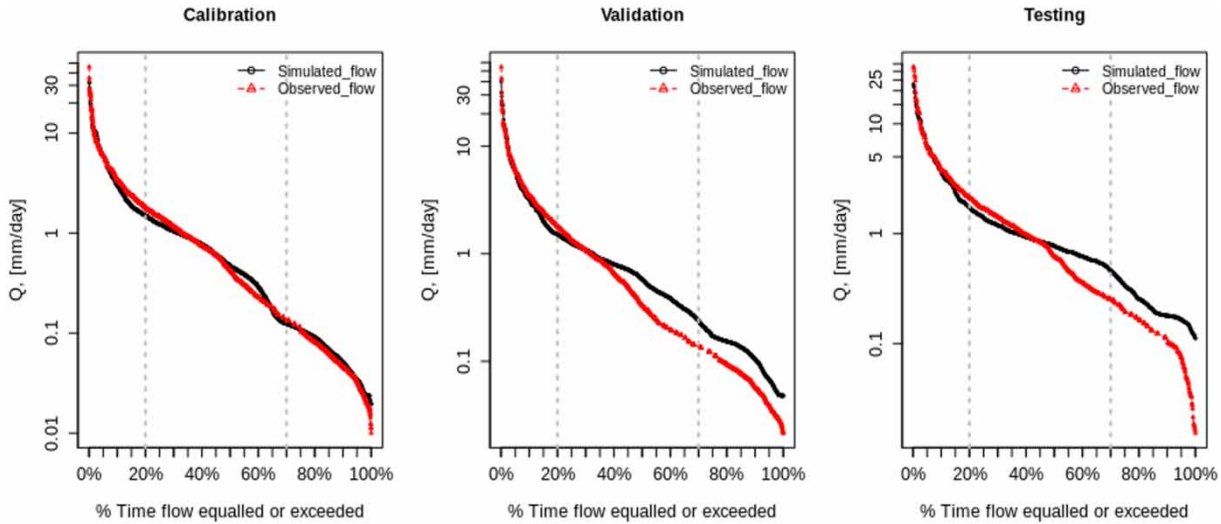
percentage implies that the parameter uncertainty solely is competent enough to capture most of the total uncertainty of the optimal model (Boukezzzi *et al.* 2016). Out of the 38 model parameters of the optimal model, seven parameters ( $b_1, b_4, b_5, L_{2,5}, a_{2,5}, a_{1,7}, a_{2,7}$ ) can be identified as the

model sensitive parameters towards achieving high model performances. Model parameter details are given in the Supplementary Appendix.

The calibrated values of the optimal model reveal that almost the entire surface runoff consists of three surface runoff components:  $q_{sur,2}$  of RB<sub>1</sub>,  $q_{sur,1}$  and  $q_{sur,2}$  of RB<sub>7</sub>. The associated time delays of their lag functions are 0.04, 0.05 and 0.01 days, respectively. This suggests that the optimal model has a quick discharge response to its forcing terms. Interestingly, the experimental insights of the Sipsey Fork catchment indicate that the most distinct feature of the catchment is its rapid flow response to the precipitation events.



**Figure 9** | Simulated and observed hydrographs of the Sipsey Fork catchment – hydrological modelling.



**Figure 10** | Simulated and observed FDCs of the Sipsey Fork catchment – hydrological modelling.

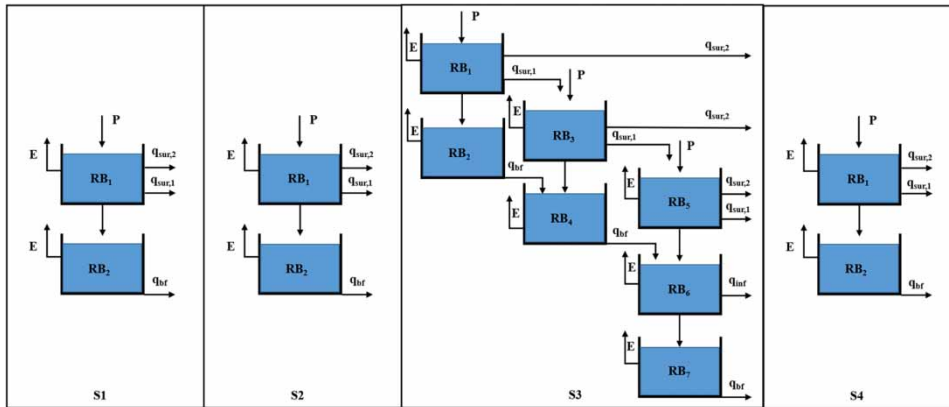
Shallow soils, deeply incised stream channels and steep topography have been identified as the causes of such flashy behaviour (Mast & Turk 1999). On the other hand, the main stream of the catchment is characterized as a perennial river. This indicates that there may be a steady subsurface flow to sustain the streamflow during dry periods. Hence, having a significant subsurface flow component (baseflow + interflow) in the optimal model architecture is meaningful. Furthermore, the main soil type of the Sipsey Fork catchment belongs to the Montevallo Enders Townley soil association which is characterized as a deep and moderately permeable soil type (Official Soil Series Descriptions 2020). This may cause to have vertical flow processes, such as infiltration and deep percolation, and hence, water may reach deeper soil layers. Therefore, it is quite logical to have three reservoir configurations for both Zone 1 and Zone 2. However, as the flow reaches towards the riverside of the catchment (Zone 3), it can be expected to have more surface flow responses, such as saturation excess overland flow due to the presence of a shallow water table. Hence, having one reservoir with only surface runoff components in the optimal model configuration for Zone 3 is reasonable.

### Semi-distributed modelling

MIKA-SHA identified the following model architecture (Figure 11) to represent the watershed dynamics of the

Red Creek catchment. Here, S1, S2, S3 and S4 stand for the four major soil associations of the catchment. S1, S2 and S4 model structures consist of series model configurations (only one branch) with two reservoirs each. The S3 model structure has a parallel model configuration with three branches. First two branches consist of two reservoirs each and the last branch has three reservoirs. The model outflows of S1, S2 and S4 model configurations consist of two surface runoff components ( $q_{sur1}$  and  $q_{sur2}$  of RB<sub>1</sub>) and one baseflow component ( $q_{bf}$  of RB<sub>2</sub>). The simulated discharge values during the calibration period reveal that S1, S2 and S4 model outflows are dominated by their surface runoff components (89% in S1, 77% in S2 and 74% in S4). In contrast, the outflow of S3 model configuration consists of four surface runoff components ( $q_{sur1}$  of RB<sub>5</sub> and  $q_{sur2}$  of RB<sub>1</sub>, RB<sub>3</sub> and RB<sub>5</sub>), one interflow component ( $q_{inf}$  of RB<sub>6</sub>) and one baseflow component ( $q_{bf}$  of RB<sub>7</sub>). Furthermore, the total discharge of the S3 model configuration is dominated by subsurface discharge components (interflow 33% and baseflow 49%).

The performance matrices of the optimal model are given in Table 7. The optimal model shows a consistent performance throughout calibration, validation and testing periods. Hence, we may expect no overfitting issues. It can be noted that the testing fitness values are higher than the validation fitness values in all four absolute performance measures. This may be due to the relatively simpler



**Figure 11** | MIKA-SHA optimal model for the Red Creek catchment (RB<sub>*i*</sub>: *i*th reservoir, *P*: precipitation, *E*: evaporation,  $q_{sur1}$  and  $q_{sur2}$ : surface runoff one and two,  $q_{inf}$ : interflow,  $q_{br}$ : baseflow, S1: Susquehanna Poarch Benndale, S2: Smithton, S3: McLaurin Heidal, S4: Susquehanna Benndale).

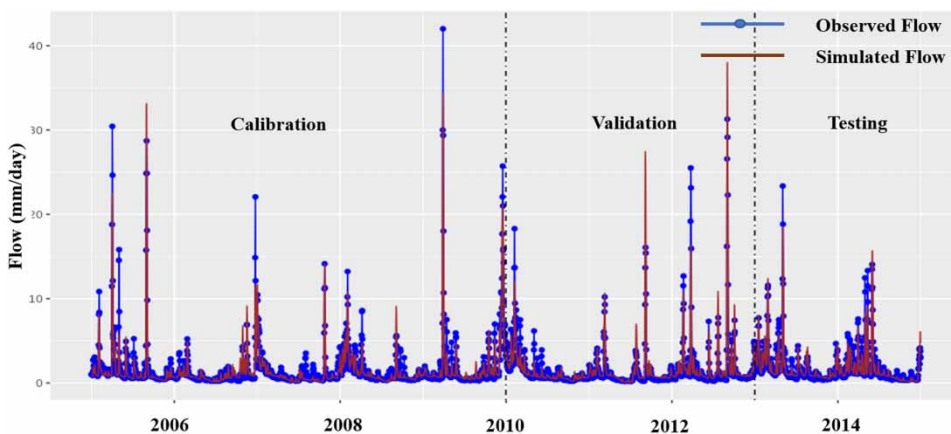
**Table 7** | Performance matrices of the MIKA-SHA optimal model – Red Creek catchment

Period	VE	KGE	NSE	logNSE
Calibration	0.75	0.94	0.91	0.84
Validation	0.70	0.85	0.85	0.85
Testing	0.76	0.92	0.88	0.88

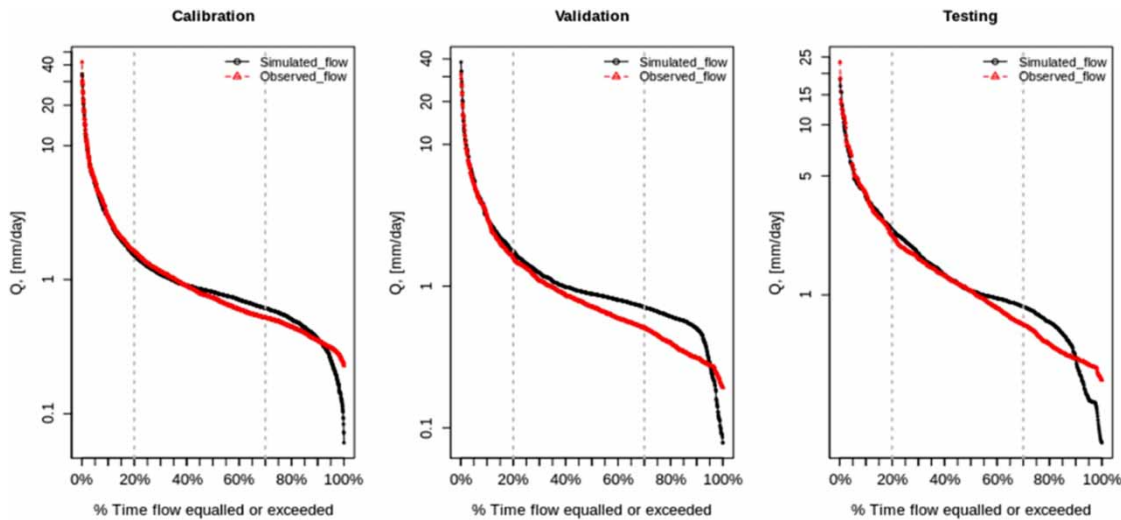
discharge signature of the Red Creek catchment during the testing period. A good visual match can be observed between simulated and observed hydrographs (Figure 12). The simulated FDC follows the observed FDC closely in the calibration period and tends to deviate at low flows in the validation and testing periods (Figure 13). As per the uncertainty analysis, 78% of the observed discharge values during the calibration period fall within the 90%

uncertainty bounds. Hence, it would be safe to accept that the parameter uncertainty of the optimal model is capable to estimate the total output uncertainty satisfactorily. By studying the shape of the sensitivity scatterplots, nine model parameters can be recognized as the model sensitive parameters of the optimal model ( $b_1$  of S1,  $b_1$  of S2,  $b_1$ ,  $b_2$  and  $b_3$  of S3,  $a_{2,1}$  and  $b_1$  of S4,  $L_{HRU}$  and  $L_{sub}$ ). Refer to the Supplementary Appendix for the model parameter details.

Most of the catchment area belongs to soil type S3 (60%). Due to the large spatial extent of S3 soil type (McLaurin Heidal) from hillside to the riverside of the catchment, the three-branch configuration of the MIKA-SHA optimal model is meaningful. The remaining three soil types have approximately the same spatial coverage.



**Figure 12** | Simulated and observed hydrographs of the Red Creek catchment – hydrological modelling.



**Figure 13** | Simulated and observed FDCs of the Red Creek catchment – hydrological modelling.

Furthermore, the S3 soil type is characterized as a sandy, well-drained soil type with a moderate permeability (Official Soil Series Descriptions 2020). This nature may encourage more subsurface type response due to the vertical drainage, such as infiltration and deep percolation. Interestingly, the outflow of the S3 model structure of the optimal model is also dominated by subsurface flow components (82% of the simulated discharge during the calibration period). On the other hand, the remaining three model configurations (S1, S2 and S4) consist of surface flow dominated discharge responses. The S2 soil type (Smithton) is formed in loamy alluvial sediments and characterized as a very deep, poorly drained soil type with a slow permeability. On top of that, as the S2 soil type is located close to the river network of the catchment, a shallow water table can also be expected. These properties may result in a high percentage of surface flow in runoff generation of S2 soil type. Both S1 (Susquehanna Poarch Benndale) and S4 (Susquehanna Benndale) soil types consist of Susquehanna soil series which is characterized as a soil type with a very slow permeability. Therefore, it can be expected to have surface runoff components like infiltration excess overland flow in both soil types. Furthermore, the bottom reservoirs of each S1, S2 and S4 model configurations show large storage throughout the calibration period. The experiment insights of the catchment reveal that S1, S2 and S4 soil types have deep to very deep soil stratum which may act as aquifers. All four model configurations have at least one subsurface flow component

and hence provide a stable baseflow in runoff generation. Interestingly, a stable ground flow response can also be expected in the catchment dynamics of the Red Creek watershed as its main river channel is categorized as a perennial type river that has a continuous runoff throughout the year.

## FORECASTING VS. MODELLING

The most common application of GP in rainfall-runoff modelling so far is based on the use of the algorithm as a short-term forecasting tool. As shown in this and many previous studies, GP is capable of achieving high-efficiency values when it is used as a forecasting tool. However, in this study, we explore the potential of GP as a model induction toolkit. In contrast to the traditional GP as a forecasting tool, the two model induction toolkits (ML-RR-MI and MIKA-SHA) consist of the following augmentations when they are used as model induction engines.

- GP is used to simulate the runoff at the time  $t$  using the forcing terms at the time  $t$  which involves no lead times. The precipitation, evaporation and streamflow histories are automatically taken into account by the reservoir storage and lag functions.
- Existing hydrological knowledge is incorporated as special functions (RB, RJOIN and DISTRIBUTED) into the GP function set. The objective of adding special

functions is to make the induced models readily interpretable and to capture the complex runoff dynamics.

- Input variable data are divided into three segments (calibration, validation and testing) and the performance of the validation period is also considered in the optimal model selection stage. This is used to avoid the selection of overfitted models as the optimal model.
- A multi-objective optimization scheme is used to optimize both model structure and parameters simultaneously. Models selected using single-objective optimization may be biased towards a specific flow segment. For example, the popular NSE is sensitive towards the high flows. Hence, a model derived through a multi-objective optimization is expected to perform well in many flow characteristics.
- Optimal model selection is not only based on the best training fitness but also based on validation fitness, relative performance in capturing observed FDC, model parsimony, time-series complexity and pattern matching.
- Induced models are readily interpretable and provide insights about catchment dynamics.
- Parallel computation is used at the fitness calculation stage. As the special functions require considerably longer computational times than the basic mathematical functions, the use of parallel computation greatly helps to reduce overall computational time.

As it can be seen with the Red Creek catchment, GP as a model induction toolkit can achieve the same high-efficiency values similar to runoff forecasts (one-day forecasts). More importantly, when the runoff signature is much more complex like with the Sipsev Fork catchment, the model induction toolkit was able to achieve satisfactory prediction power which runoff forecasts through traditional GP were unable to achieve. In the present study, a relatively shorter period (6 years) was used for the model training (calibration). However, both model induction toolkits were able to capture the runoff signature reasonably well within that period. Obtaining a long quality dataset for model training is often challenging in hydrological modelling. Therefore, it may be safe to assume GP as a model induction toolkit performs better than forecasting when the discharge signatures are more complex and

difficult to model and/or when the training labelled instances are limited.

## CONCLUSIONS

We recognize the potential offered by ML algorithms towards hydrological modelling. We also recognize that simplistic black-box type data-driven models may lead to the evolution of accurate; yet, senseless models with serious difficulties with interpretation may not serve towards the advancement of hydrological knowledge. Therefore, we chart the most promising way ahead through the integration of existing hydrological knowledge with learning algorithms to induce more generalizable and physically coherent models. This was the motivation behind the development of GP-based model induction frameworks which have been founded on both ML and theory-driven models. Therefore, we expect this work will strengthen the link between two major, historically, quite separate communities in water resource science and engineering: those working with physics-based process simulation modelling and those working with machine learning.

Results demonstrated in this contribution the potential of GP as a model induction toolkit in contrast to its most frequent usage as a short-term forecasting tool. More importantly, GP as a model induction engine preserves its high prediction accuracies shown with runoff forecasting while benefiting hydrologists to gain a better understanding of the watershed dynamics through the resultant interpretable models. Furthermore, GP as a model induction toolkit produces more accurate results than forecasting when the discharge signatures are more complex and difficult to model and/or when the training labelled instances are limited. Furthermore, in the current study, GP is used to its full capacity as a model induction engine to simultaneously optimize both model configuration and parameters.

GP model induction toolkits (MIKA-SHA and ML-RR-MI) may serve as an alternative to the traditional subjective model selection approach by automatically inducing optimal model configurations for a watershed of concern based on adequacy rather than legacy. Therefore, the modular approach of these model induction toolkits may address

the so-called uniqueness of the place in hydrological modeling. The unique perhaps more important feature of these model induction toolkits would be their capability to couple with any internally coherent collection of building blocks representing the elements of hydrological knowledge. The optimal model configurations derived in this study are in agreement with experimental insights and previously disclosed research findings of the catchments. Hence, GP as a model induction toolkit is smart enough to mine knowledge from data which makes it viable to depend on the induced models with more than just statistical confidence.

The optimal models derived by the two GP-based model induction toolkits are readily interpretable by domain experts. This makes the approach introduced here different from the other ML utilizations in rainfall-runoff modelling. We need data to build models. In the absence of data, human experts will be able to construct models based on their intuition since even human-built models require some sort of validation against data. With the limited data, both human and our model induction toolkits (both ML-RR-MI and MIKA-SHA) face the same challenge. DL models will not be useful in such a situation due to the very large number of free parameters. From the perspective of the bias-variance dilemma, DL models, such as Long Short-Term Memory networks (LSTM), will not be particularly useful under the poor data availability circumstances. However, the present work is just the beginning of coupling the strengths of human awareness with those of computational discovery techniques. We expect further research studies on theory-guided machine learning to be directed towards knowledge discovery and automatic model building in hydrological modelling.

## DATA AVAILABILITY STATEMENT

All relevant data are included in the paper or its Supplementary Information.

## REFERENCES

- Addor, N. & Melsen, L. A. 2019 *Legacy, rather than adequacy, drives the selection of hydrological models*. *Water Resources Research* **55**. <https://doi.org/10.1029/2018WR022958>.
- Babovic, V. 1996 *Hydroinformatics: Emergence, Evolution, Intelligence: A Study of Distributed and Decentralised Computing Using Intelligent Agents*. A. A. Balkema Publishers, Rotterdam.
- Babovic, V. 2000 *Data mining and knowledge discovery in sediment transport*. *Computer-Aided Civil and Infrastructure Engineering* **15** (5), 383–389.
- Babovic, V. 2005 *Data mining in hydrology*. *Hydrological Processes* **19** (7), 1511–1515. <https://doi.org/10.1002/hyp.5862>.
- Babovic, V. 2009 *Introducing knowledge into learning based on genetic programming*. *Journal of Hydroinformatics* **11** (3–4), 181–195.
- Babovic, V. & Keijzer, M. 2000 *Genetic programming as a model induction engine*. *Journal of Hydroinformatics* **2** (1), 35–60.
- Babovic, V. & Keijzer, M. 2002 *Rainfall-runoff modelling based on genetic programming*. *Hydrology Research* **33** (5), 331–346.
- Babovic, V., Keijzer, M. & Stefansson, M. 2000 *Optimal embedding using evolutionary algorithms*. In: *Proc. 4th Int. Conference on Hydroinformatics*, Cedar Rapids.
- Babovic, V., Li, X. & Chadalawada, J. 2020 *Rainfall-runoff modeling based on genetic programming*. In: *Encyclopedia of Water: Science, Technology, and Society, 5 Volume Set* (P. Maurice, ed.). Wiley, New York, USA, p. 1081.
- Bautu, A. & Bautu, E. 2006 *Meteorological data analysis and prediction by means of genetic programming*. In: *Proceedings of the 5th Workshop on Mathematical Modeling of Environmental and Life Sciences Problems*, Constanta, Romania, pp. 35–42.
- Beven, K. 2020 *Deep learning, hydrological processes and the uniqueness of place*. *Hydrological Processes* **34** (16), 3608–3613. <https://doi.org/10.1002/hyp.13805>.
- Beven, J. K. & Binley, M. A. 1992 *The future of distributed models: model calibration and uncertainty prediction*. *Hydrological Processes* **6**, 278–298.
- Boukezzzi, Z. L., Boukezzzi, L. & Errih, M. 2016 *Uncertainty analysis of HEC-HMS model using the glue method for flash flood forecasting of Mekerra watershed, Algeria*. *Arabian Journal of Geosciences* **9** (20), 1–12.
- Boyle, D. P., Gupta, H. V., Sorooshian, S., Koren, V., Zhang, Z. & Smith, M. 2001 *Toward improved streamflow forecasts: value of semi-distributed modelling*. *Water Resources Research* **37**, 2749–2759. [doi:10.1029/2000wr000207](https://doi.org/10.1029/2000wr000207).
- Cannon, A. J. & Mckendry, I. G. 2002 *A graphical sensitivity analysis for statistical climate models: application to Indian monsoon rainfall prediction by artificial neural networks and multiple linear regression models*. *International Journal of Climatology* **22** (13), 1687–1708. <https://doi.org/10.1002/joc.811>.
- Chadalawada, J., Havlicek, V. & Babovic, V. 2017 *A genetic programming approach to system identification of rainfall-runoff models*. *Water Resources Management* **31** (12), 3975–3992. [doi:10.1007/s11269-017-1719-1](https://doi.org/10.1007/s11269-017-1719-1).
- Chadalawada, J., Herath, H. M. V. V. & Babovic, V. 2020 *Hydrologically informed machine learning for rainfall-runoff modeling: a genetic programming-based toolkit for automatic*

- model induction. *Water Resources Research* **56**. <https://doi.org/10.1029/2019WR026933>.
- Clark, M. P., Slater, A. G., Rupp, D. E., Woods, R. A., Vrugt, J. A., Gupta, H. V., Wagener, T. & Hay, L. E. 2008 **Framework for understanding structural errors (FUSE): a modular framework to diagnose differences between hydrological models**. *Water Resources Research* **44**, W00B02. <https://doi.org/10.1029/2007WR006735>.
- Criss, R. E. & Winston, W. E. 2008 **Do Nash values have value? Discussion and alternate proposals**. *Hydrological Processes* **22** (14), 2723.
- Datta, B., Prakash, O. & Sreekanth, J. 2014 *Application of Genetic Programming Models Incorporated in Optimization Models for Contaminated Groundwater Systems Management, Evolve – A Bridge Between Probability, Set Oriented Numerics, and Evolutionary Computation* V. Springer, NY, USA, pp. 183–199.
- Daymet 2020 <https://daymet.ornl.gov/> (accessed 20 March 2020).
- Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T. 2002 **A fast and elitist multiobjective genetic algorithm: NSGA-II**. *IEEE Transactions on Evolutionary Computation* **6** (2), 182–197.
- Elshorbagy, A. & El-Baroudy, I. 2009 **Investigating the capabilities of evolutionary data-driven techniques using the challenging estimation of soil moisture content**. *Journal of Hydroinformatics* **11** (3–4), 237–251.
- Fallah-Mehdipour, E. & Haddad, O. B. 2015 **Application of genetic programming in hydrology**. In: *Handbook of Genetic Programming Applications* (A. Gandomi, A. Alavi & C. Ryan, eds.). Springer, Cham, pp. 59–70.
- Fallah-Mehdipour, E., Haddad, O. B. & Mariño, M. 2013 **Prediction and simulation of monthly groundwater levels by genetic programming**. *Journal of Hydro-Environment Research* **7** (4), 253–260.
- Faticchi, S., Vivoni, E. R., Ogden, F. L., Ivanov, V. Y., Mirus, B., Gochis, D., Downer, C. W., Camporese, M., Davison, J. H., Ebel, B., Jones, N., Kim, J., Mascaro, G., Niswonger, R., Restrepo, P., Rigon, R., Shen, C., Sulis, M. & Tarboton, D. 2016 **An overview of current applications, challenges, and future trends in distributed process-based models in hydrology**. *Journal of Hydrology* **537**, 45–60. doi:10.1016/j.jhydrol.2016.03.026.
- Fenicia, F., Kavetski, D. & Savenije, H. H. G. 2011 **Elements of a flexible approach for conceptual hydrological modeling: 1. motivation and theoretical development**. *Water Resources Research* **47**, W11510. <https://doi.org/10.1029/2010WR010174>.
- Fenicia, F., Kavetski, D., Savenije, H. H. & Pfister, L. 2016 **From spatially variable streamflow to distributed hydrological models: analysis of key modeling decisions**. *Water Resources Research* **52**, 954–989. <https://doi.org/10.1002/2015WR017398>.
- Fleming, S. W. 2007 **Artificial neural network forecasting of nonlinear Markov processes**. *Canadian Journal of Physics* **85** (3), 279–294. <https://doi.org/10.1139/p07-037>.
- Giuliani, M., Castelletti, A., Pianosi, F., Mason, E. & Reed, P. M. 2015 **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). Article number 4015050. [https://doi.org/10.1061/\(asce\)wr.1943-5452.0000570](https://doi.org/10.1061/(asce)wr.1943-5452.0000570).
- Govindaraju, R. S. 2000 **Artificial neural networks in hydrology. II: Hydrologic applications**. *Journal of Hydrologic Engineering* **5**, 124–137.
- Gupta, H. V., Kling, H., Yilmaz, K. K. & Martinez, G. F. 2009 **Decomposition of the mean squared error and NSE performance criteria: implications for improving hydrological modelling**. *Journal of Hydrology* **377** (1), 80–91.
- Havlíček, V., Hanel, M., Máca, P., Kuráž, M. & Pech, P. 2013 **Incorporating basic hydrological concepts into genetic programming for rainfall-runoff forecasting**. *Computing* **95** (1), 363–380.
- Herath, H. M. V. V., Chadalawada, J. & Babovic, V. 2020 **Hydrologically informed machine learning for rainfall-runoff modelling: towards distributed modelling**. *Hydrology and Earth System Sciences Discussions*. in review. <https://doi.org/10.5194/hess-2020-487>.
- Hsieh, W. W. 2009 *Machine Learning in the Environmental Sciences*. Cambridge University Press, Cambridge, UK.
- Karimi, S., Shiri, J., Kisi, O. & Shiri, A. A. 2016 **Short-term and long-term streamflow prediction by using ‘wavelet-gene expression’ programming approach**. *ISH Journal of Hydraulic Engineering* **22**, 148–162.
- Karpatne, A., Atluri, G., Faghmous, J. H., Steinbach, M., Banerjee, A., Ganguly, A., Shekhar, S., Samatova, N. & Kumar, V. 2017 **Theory-guided data science: a new paradigm for scientific discovery from data**. *IEEE Transactions on Knowledge and Data Engineering* **29** (10), 2318–2331. doi:10.1109/TKDE.2017.2720168.
- Kavetski, D. & Fenicia, F. 2011 **Elements of a flexible approach for conceptual hydrological modeling: 2. application and experimental insights**. *Water Resources Research* **47**, W11511. <https://doi.org/10.1029/2011WR010748>.
- Keijzer, M. & Babovic, V. 2002 **Declarative and preferential bias in GP-based scientific discovery**. *Genetic Programming and Evolvable Machines* **3**, 41–79.
- Keijzer, M., Babovic, V., Ryan, C., O’Neill, M. & Cattolico, M. 2001 **Adaptive logic programming**. In: *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*, San Francisco, USA, pp. 42–49.
- Koza, J. R. 1992 *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, vol. 1. MIT Press, Cambridge, MA.
- Kratzert, F., Klotz, D., Herrnegger, M., Sampson, A. K., Hochreiter, S. & Nearing, G. 2019 **Toward improved predictions in ungauged basins: exploiting the power of machine learning**. *Water Resources Research* **55**, 11344–11354. <https://doi.org/10.1029/2019WR026065>.
- Krause, P., Boyle, D. & Bäse, F. 2005 **Comparison of different efficiency criteria for hydrological model assessment**. *Advances in Geosciences* **5**, 89–97.
- Ley, R., Hellebrand, H., Casper, M. C. & Fenicia, F. 2016 **Comparing classical performance measures with signature indices derived from flow duration curves to assess model**

- structures as tools for catchment classification. *Hydrology Research* **47** (1), 1–14.
- Lohani, A. K., Goel, N. K. & Bhatia, K. K. S. 2014 **Improving real time flood forecasting using fuzzy inference system**. *Journal of Hydrology* **509**, 25–41. <http://dx.doi.org/10.1016/j.jhydrol.2013.11.021>.
- Mast, M. A. & Turk, J. T. 1999 Environmental characteristics and water quality of hydrologic benchmark network stations in the Eastern United States, 1963-95. *U.S. Geological Survey Circular* **1173-A**, 158.
- McGovern, A., Lagerquist, R., Gagne, D. J., Jergensen, G. E., Elmore, K. L., Homeyer, C. R. & Smith, T. 2019 **Making the black box more transparent: understanding the physical implications of machine learning**. *Bulletin of the American Meteorological Society* **100** (11), 2175–2199.
- Mehr, A. D., Nourani, V., Kahya, E., Hrnjica, B., Sattar, A. M. A. & Yaseen, Z. M. 2018 **Genetic programming in water resources engineering: a state-of-the-art review**. *Journal of Hydrology* **566**, 643–667.
- Meshgi, A., Schmitter, P., Babovic, V. & Chui, T. F. M. 2014 **An empirical method for approximating stream baseflow time series using groundwater table fluctuations**. *Journal of Hydrology* **519**, 1031–1041.
- Minns, A. W. & Hall, M. J. 1996 **Artificial neural networks as rainfall-runoff models**. *Hydrological Sciences Journal* **41** (3), 399–417.
- Moosavi, V., Vafakhah, M., Shirmohammadi, B. & Behnia, N. 2013 **A wavelet-ANFIS hybrid model for groundwater level forecasting for different prediction periods**. *Water Resources Management* **27**, 1301–1321. <http://dx.doi.org/10.1007/s11269-012-0239-2>.
- Nash, J. E. & Sutcliffe, J. V. 1970 **River flow forecasting through conceptual models part I – a discussion of principles**. *Journal of Hydrology* **10** (3), 282–290.
- Nearing, G. S., Kratzert, F., Sampson, A. K., Pelissier, C. S., Klotz, D., Frame, J. M. & Gupta, H. V. 2020 **What role does hydrological science play in the age of machine learning?** *Water Resources Research* **57** (3), e2020WR028091.
- Newman, A. J., Clark, M. P., Sampson, K., Wood, A., Hay, L. E., Bock, A., Viger, R. J., Blodgett, D., Brekke, L., Arnold, J. R. & Hopson, T. 2015 **Development of a large-sample watershed-scale hydrometeorological dataset for the contiguous USA: dataset characteristics and assessment of regional variability in hydrologic model performance**. *Hydrology and Earth System Sciences* **19**, 209–223.
- Nie, W., Krautblatter, M., Leith, K., Thuro, K. & Festl, J. 2017 **A modified tank model including snowmelt and infiltration time lags for deep-seated landslides in alpine environments (Aggenalm, Germany)**. *Natural Hazards and Earth System Sciences* **17** (9), 1595–1610. doi:10.5194/nhess-17-1595-2017.
- Official Soil Series Descriptions 2020 <https://soilseries.sc.egov.usda.gov/> (accessed 29 July 2020).
- Physics Informed Machine Learning Conference 2016, Santa Fe, New Mexico, USA. Available from: <http://www.cvent.com/events/physics-informed-machine-learning/event-summary-7cd2f46ebc144bdeb6e5f4106887ea04.aspx>.
- QGIS.org 2020 *QGIS Geographic Information System, Open Source Geospatial Foundation Project*. Available from: <https://qgis.org/en/site/> (accessed 25 August 2020).
- R Core Team 2013 *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. Available from: <http://www.R-project.org/>.
- Safari, M. J. S. & Mehr, A. D. 2018 **Multigene genetic programming for sediment transport modeling in sewers at non-deposition with deposited bed condition**. *International Journal of Sediment Research*. <https://doi.org/10.1016/j.ijsrc.2018.04.007>.
- Savic, D. & Khu, S. T. 2005 Evolutionary computing in hydrological sciences. In: *Encyclopedia of Hydrological Sciences* (M. G. Anderson, ed.). Wiley, New York, pp. 331–348.
- Selle, B. & Muttill, N. 2011 **Testing the structure of a hydrological model using genetic programming**. *Journal of Hydrology* **397** (1), 1–9.
- Shen, C., Laloy, E., Elshorbagy, A., Albert, A., Bales, J., Chang, F. J., Ganguly, S., Hsu, K. L., Kifer, D., Fang, Z. & Fang, K. 2018 **HESS opinions: incubating deep-learning-powered hydrologic science advances as a community**. *Hydrology and Earth System Sciences* **22**, 5639–5656. doi:10.5194/hess-22-5639-2018.
- Snauffer, A. M., Hsieh, W. W., Cannon, A. J. & Schnorbus, M. A. 2018 **Improving gridded snow water equivalent products in British Columbia, Canada: multi-source data fusion by neural network models**. *The Cryosphere* **12** (3), 891–905.
- Solander, K. C., Bennett, K. E., Fleming, S. W. & Middleton, R. S. 2019 **Estimating hydrologic vulnerabilities to climate change using simulated historical data: a proof-of-concept for a rapid assessment algorithm in the Colorado river basin**. *Journal of Hydrology: Regional Studies* **26** (100), 642.
- Song, J., Her, Y., Park, J., Lee, K. & Kang, M. 2017 **Simulink implementation of a hydrologic model: a tank model case study**. *Water* **9** (9), 639. doi:10.3390/w9090639.
- Sugawara, M. 1979 **Automatic calibration of the tank model/l'etalonnage automatique d'un modele a cisterne**. *Hydrological Sciences Journal* **24** (3), 375–388. <https://doi.org/10.1080/02626667909491876>.
- USDA's Geospatial Data Gateway 2020 <http://datagateway.nrcs.usda.gov/> (accessed 21 March 2020).
- USGS EarthExplorer 2020 <https://earthexplorer.usgs.gov/> (accessed 20 March 2020).
- Voosen, P. 2017 **The AI detectives**. *Science* **357** (6346), 22–27.
- Yaseen, Z. M., El-shafie, A., Jaafar, O. & Afan, H. A. 2015 **Artificial intelligence based models for stream-flow forecasting: 2000–2015**. *Journal of Hydrology* **530**, 829–844. <http://dx.doi.org/10.1016/j.jhydrol.2015.10.038>.
- Yu, X., Liong, S. Y. & Babovic, V. 2004 **EC-SVM approach for real-time hydrologic forecasting**. *Journal of Hydroinformatics* **6** (3), 209–202.