Modular Assessment of Rainfall-Runoff Models Toolbox (MARRMoT) v1.2: an open-source, extendable framework providing implementations of 46 conceptual hydrologic models as continuous state-space formulations

Wouter J. M. Knoben[1], Jim E. Freer[2], Keirnan J. A. Fowler[3], Murray C. Peel[3], Ross A Woods[1]

[1]Department of Civil Engineering, University of Bristol, Bristol, BS8 1TR, United Kingdom
[2]School of Geographical Science, University of Bristol, Bristol, BS8 1BF, United Kingdom
[3]Department of Infrastructure Engineering, University of Melbourne, Melbourne, Parkville VIC 3052, Australia

Correspondence to: Wouter J. M. Knoben (w.j.m.knoben@bristol.ac.uk)

Abstract. This paper presents the Modular Assessment of Rainfall-Runoff Models Toolbox (MARRMoT): a modular open-source toolbox containing documentation and model code based on 46 existing conceptual hydrologic models. The toolbox is developed in Matlab and works with Octave. MARRMoT models are based solely on traceable published material and model documentation, not on already existing computer code. Models are implemented following several good practices of model development: definition of model equations (the mathematical model) is kept separate from the numerical methods used to solve these equations (the numerical model) to generate clean code that is easy to adjust and debug; the Implicit Euler time-stepping scheme is provided as the default option to numerically approximate each model's Ordinary Differential Equations in a more robust way than (common) Explicit schemes would; threshold equations are smoothed to avoid discontinuities in the model's objective function space; and the model equations are solved simultaneously, avoiding physically unrealistic sequential solving of fluxes. Generalized parameter ranges are provided to assist with model inter-comparison studies. In addition to this paper and its Supporting Materials, a User Manual is provided together with several workflow scripts that show basic example applications of the toolbox. The toolbox and User Manual are available from https://github.com/wknoben/MARRMoT (DOI: 10.5281/zenodo.3235664). Our main scientific objective in developing this toolbox is to facilitate the inter-comparison of conceptual hydrological model structures which are in widespread use, in order to ultimately reduce the uncertainty in model structure selection.

1 Introduction

Rainfall-runoff modelling is useful to extrapolate our hydrologic understanding beyond measurement availability (Beven, 2009, 2012). We can challenge and improve our understanding of the way catchments function through model-based hypothesis testing (Beven, 2002; Clark et al., 2011; Fenicia et al., 2008b; Kirchner, 2006, 2016) and simulate the impact of changes in climatic conditions and catchment characteristics such as land use change (Bathurst et al., 2004; Ewen and Parkin, 1996; Klemeš, 1986; Peel and Blöschl, 2011; Seibert and van Meerveld, 2016; Wagener et al., 2010). Many different modelling approaches are possible, ranging from lumped, empirical, deterministic bucket-style models to distributed, process-oriented, stochastic, 3D physics-based models (Beven, 2012). Each of these approaches has its own advantages and drawbacks, concerning the level of spatial detail, amount of model 'realism' in terms of processes represented, input data requirements and computational time. The toolbox presented in this paper uses deterministic, spatially lumped bucket-style models, also referred to as conceptual hydrological models. Note that this definition of a conceptual model is different from the definition used by authors discussing the modelling process, where the conceptual model is a step between having a mental, perceptual model of a catchment and the collection of equations referred to as a mathematical/procedural model (e.g. Beven, 2012; Clark and Kavetski, 2010; Gupta et al., 2012; Refsgaard and Henriksen, 2004).

Every application of a rainfall-runoff model is complicated by various aspects of uncertainty (e.g. Beven and Freer, 2001b; Pechlivanidis et al., 2011; Peel and Blöschl, 2011). Uncertainty is introduced during measurement of model input variables such as precipitation (e.g. Oudin et al., 2006) and temperature (e.g. Bárdossy and Singh, 2008) and derived variables such as potential evapotranspiration (e.g. Andréassian et al., 2004; Oudin et al., 2005, 2006). Uncertainty is also present in measurements against which model output is compared, such as streamflow (e.g. Di Baldassarre and Montanari, 2009; McMillan et al., 2010), water table depth (e.g. Freer et al., 2004) and water quality (e.g. McMillan et al., 2012). Values of model parameters can be uncertain due to dependency of 'optimal' parameter values on climatic conditions during model calibration (e.g. Coron et al., 2012; Fowler et al., 2016), due to the choice of calibration algorithm (Arsenault et al., 2014) or due to the performance metric used (e.g. Efstratiadis and Koutsoyiannis, 2010; Gupta et al., 2009). Finally, the choice of model structure (i.e. the collection of equations and their internal connections that make up the model) itself is uncertain (Andréassian et al., 2009; Coron et al., 2012; Van Esse et al., 2013; Fenicia et al., 2008a, 2014; Krueger et al., 2010). Currently, a wide variety of models are available. They may be different in spatial and temporal resolution, or include different processes, be deterministic or stochastic, might be based on top-down or bottom-up philosophies, or be different in some other way. This paper contributes to the investigation of model structure uncertainty of lumped, deterministic conceptual models. We hope to make progress towards answering a core question in hydrologic modelling: out of the overwhelming number of available models, which one is the most appropriate choice for a given catchment?

Conceptual models tend to have low data requirements (catchment-averaged forcing instead of spatially explicit) and are less computationally intensive than spatially explicit models. They are used in both scientific and operational settings (Perrin et al., 2001). A wide range of conceptual model structures exists, e.g. SACRAMENTO (Burnash, 1995; National Weather Service, 2005), TOPMODEL (Beven and Freer, 2001a), SIMHYD (Chiew et al., 2002), the TANK model (Sugawara, 1995) and many more, but there is no clear basis to choose between the different models (Beven, 2012). Models are different both in their internal structure (i.e. which storages are represented and how they are connected) and in their choice of flux equations (i.e. whether and how any given flux is quantified with a mathematical equation). Choosing the right model for a catchment where hydrological responses are measured is difficult because achieving a 'good' value on a performance metric is a necessary but not sufficient condition to determine whether a model produces the "right results for the right reasons" (Kirchner, 2006).

Different model structures can achieve superficially similar performance metrics, but might reach this point by wildly different internal dynamics (de Boer-Euser et al., 2017; Goswami and O'Connor, 2010; Perrin et al., 2001). Therefore, good simulation metrics do not necessarily tell us which model structure is more appropriate for this catchment. Choosing a suitable model structure where the catchment is ungauged is even more challenging. This model structure uncertainty is largely unquantified, even for existing models with a long legacy of 'successful' (often defined as having achieved a high value for some performance metric) applications. However, comparison of different models can be an expensive task if each model needs to be set up individually. Model inter-comparison studies are further complicated by the fact that documented computer code is unavailable for many model structures.

In recent years multi-model frameworks have received considerable attention. These provide a standardized framework in which several models are presented, or users can construct new models, or both. This reduces the time cost of a model comparison study, allows fair comparison of different model structures in a test case and allows the investigator to isolate choices in the model development process. Examples include the Modular Modelling System (MMS, Leavesley et al., 1996), the Rainfall-Runoff Modelling Toolbox (RRMT, Wagener et al., 2002), the Framework for Understanding Structural Errors (Clark et al., 2008), a fuzzy model selection framework (Bai et al., 2009), SUPERFLEX (Fenicia et al., 2011; Kavetski and Fenicia, 2011), the Catchment Modelling Framework (CMF, Kraft et al., 2011) and the Structure for Unifying Multiple Modelling Alternatives (SUMMA, Clark et al., 2015a, 2015b). These frameworks are either limited to a small number of existing models (e.g. MMS, RRMT), use a pre-defined internal organization of stores (FUSE), consist of generic model elements (i.e. stores, fluxes and lags) that are not easily recognizable as existing models (e.g. CMF, SUPERFLEX), or are more physics-based and thus difficult to use with conceptual models (e.g. SUMMA). Thus, despite these many existing frameworks, there is a need for a new framework that provides a user-friendly, standardized way to construct and compare existing, widely-used conceptual models, without constraining the allowed model architecture a priori.

This paper introduces the Modular Assessment of Rainfall-Runoff Models Toolbox (MARRMoT) to fill a gap in the current selection of multi-model frameworks. MARRMoT provides an open-source, easy-to-use, expandable framework that currently includes 46 different conceptual model formulations. This provides all the benefits of a multi-model framework: models are constructed in a modular fashion from separate flux equations, which allows easy modification of provided models and expansion of the framework with new models or fluxes; good practices for numerical model solving are implemented as standard options; and all MARRMoT models require and provide standardized inputs and outputs. The large number of models in the framework will facilitate studies that lead to more generalizable conclusions about model and/or catchment functioning. This work also provides a pragmatic overview of the wide variety of different flux equations and model structures that are currently used, facilitating studies and discussion beyond direct model comparison. Due to the code being open source, transparency and repeatability of research is encouraged, additions to the framework are possible, and the community can find and correct any mistakes. Finally, MARRMoT is provided with extensive documentation about the models included, the conversion of flux equations to computer code, recommendations for generalized parameter ranges for model sensitivity analysis and/or calibration, a User Manual explaining framework setup, functioning and use, and several example workflow scripts that allow use of the framework even with minimal programming experience.

2 MARRMoT design considerations

MARRMoT takes inspiration from earlier modular frameworks (e.g. FUSE (Clark et al., 2008), FLEX (Fenicia et al., 2011)) and uses modular code with individual flux equations as the basic building blocks. Multi-model frameworks benefit from modular implementation because this simplifies programming of the framework and makes it easier to (i) re-use components of a model in a different context, including cases where the same basic equation is used by multiple models; and (ii) add new options to the framework (Clark et al., 2008). Section 2.1 gives a brief outline of the project scope and design philosophy. MARRMoT follows several good practices for model development which are briefly described in sub-sections 2.2 to 2.5.

2.1 Scope

MARRMoT's scope is limited to conceptual hydrological models and the code currently includes no spatial discretization of inputs or catchment response. Models are expected to be used in a lumped fashion, although users could create their own interface to use MARRMoT code to represent within-catchment variability using multiple lumped model structures. Required model inputs are standardized across all MARRMoT models and every model only requires time series of precipitation and potential evapotranspiration, and optionally of temperature (used by certain snow modules). Model outputs are equally standardized and provide time series of simulated flow and total evaporation fluxes, and optionally time series of model states and internal fluxes. The models are set up such that they can use a user-specified time step size (e.g. daily, hourly) which is currently effectively the temporal resolution of the forcing data. Models and flux equations internally account for this time step size, so that parameter values can use consistent units, regardless of the temporal resolution of the forcing data. The main goal of this set up is ease-of-use, so that it is straightforward to switch between different model structures within an experiment. MARRMoT models are based on written documentation only, not on existing computer code. This choice is motivated by our aim to produce traceable code and by several practical concerns. The documentation we base our models on is traceable through our cited sources. Computer code of hydrologic models tends to be less traceable than their documentation: code might be unavailable, code might not be accompanied by a persistent identifier, or multiple versions of the same model (using the same model name) might be available which complicates finding the 'original' computer code. This is supported by various authors who developed the original models: "Today many versions of the HBV model exist, and new codes are constantly developed by different groups …" (Lindström et al., 1997) and " … TOPMODEL is not a single model structure […] but more a set of conceptual tools" (Beven et al., 1995).

2.2 Separation of model equations and equation solving

First, MARRMoT uses a distinct separation of model equations as state-space formulations and the numerical approach used to solve these equations. In the theoretical process of developing a new hydrological model, the modeller ideally goes through several distinct steps (e.g. Beven, 2012; Clark and Kavetski, 2010; Gupta et al., 2012). To start, the modeller develops a mental, *perceptual* model of catchment behaviour based on observations and/or other knowledge (i.e. expert opinion). Next, this model is simplified into an abstraction that shows the connection of the most important fluxes and storages (also termed a *conceptual* model, but this is a distinctly different meaning than when applied to a bucket-type hydrologic model). These relations are then formalized as Ordinary Differential Equations (ODEs) and their constitutive functions in a *mathematical* model. Finally, creating computer code to solve these equations sequentially as a time series is done with the *procedural* model. In practice however, these stages are often not distinct and tend to overlap (e.g. Kavetski et al., 2003), a process referred to as "ad hoc" modelling. Overlap of the *mathematical* and *procedural* model can lead to altered model behaviour and difficulty with parameter estimation (Clark and Kavetski, 2010; Kavetski and Clark, 2010; Kavetski et al., 2003). A clear separation between model equations and the code used to solve those equations gives computer code that is easier to understand and update with new time-stepping schemes or flux equations, relative to code where the model equations are interwoven with the numerical scheme.

2.3 Robust numerical approximation of model equations

Second, MARRMoT gives the possibility to choose a numerical method to approximate the ODEs in discrete time steps. Currently, a fixed-step Implicit Euler method is recommended as default, and an Explicit Euler method is provided for result matching with previous studies. Many implementations of hydrologic models use the Explicit Euler method to approximate storage changes (Schoups et al., 2010; Singh and Woolhiser, 2002). The Explicit Euler method relies on storage values at the start of a time step to estimate flux sizes in the current time step: FLUX(t) = f(STORE(t-1)). This method is easy to implement and fast to compute, but has several disadvantages: it has low accuracy and only conditional stability, which can lead to large numerical errors and amplification of such errors under certain conditions (Clark and Kavetski, 2010; Kavetski and Clark, 2010; Schoups et al., 2010). Implicit methods such as Implicit Euler instead rely on an iterative procedure that relates flux size to storage at the end of a time step: FLUX(t) = f(STORE(t)). These methods require more intensive iterative computation, but avoid the aforementioned issues even when implemented with fixed time step sizes (Kavetski et al., 2006; Schoups et al., 2010). Higher-order numerical approximation methods are currently not provided in MARRMoT but can be included in a straightforward manner. Note that fixed time step size refers to the use of a single time step size throughout a simulation (i.e. no adaptive sub-stepping is used; see section 5.3.5) and does not prescribe the time step size (e.g. hourly, daily)

2.4 Smoothing of threshold discontinuities in model equations

Third, MARRMoT removes threshold discontinuities in model equations through logistic smoothing (Clark et al., 2008; Kavetski and Kuczera, 2007). Hydrologic processes are often characterized by thresholds, e.g. snowmelt starts when a certain temperature is exceeded, and saturation excess flow occurs when the soil is saturated. Introducing threshold behaviour into hydrologic models leads to discontinuities in the model's objective function, which can complicate parameter estimation when small changes in parameter values may lead to large changes in objective function value or in the gradient thereof (Kavetski and Kuczera, 2007). Smoothing model equations avoids these discontinuities but also involves a fundamental change to the model equations. Kavetski and Kuczera (2007) recommend logistic functions to smooth threshold equations that closely resemble the original threshold function but are continuous throughout the function's domain. MARRMoT smooths storage-based thresholds with a logistic function (Clark et al., 2008):

$$Q_o = Q_{in}\left(1 - \Phi(S, S_{max}, \rho_S, \varepsilon)\right) \tag{1}$$

$$\Phi(S, S_{max}, \rho_S, \varepsilon) = \frac{1}{1+e^{\frac{S - S_{max} \mp \omega\varepsilon}{\omega}}} \tag{2}$$

Where Q$_o$ and Q$_{in}$ are flux output and input respectively and ϕ(..) the smoothing operator. S and S$_{max}$ are current and maximum storage respectively, ω represents the degree of smoothing according to $\omega = \rho_S S_{max}$, and ε is a coefficient that ensures that S does not exceed S$_{max}$. ρ_S and ε can be specified by the user, or used with default values of 0.01 and 5.00 respectively (Clark et al., 2008). Temperature-based thresholds are smoothed with a different logistic function (Kavetski and Kuczera, 2007):

$$P_S = P\Phi(T, T_t, \rho_T) \tag{3}$$

$$\Phi(T, T_0, \rho_T) = \frac{1}{1+e^{\frac{T - T_0}{\rho_T}}} \tag{4}$$

Where P$_S$ is precipitation as snow, P incoming precipitation and ϕ(..) the smoothing operator. T and T$_0$ are the current and threshold temperatures respectively, and ρ_T is the smoothing parameter with default value 0.01.

2.5 Simultaneous solving of model equations

Fourth, MARRMoT solves all model equations simultaneously rather than sequentially. Operator-splitting (OS) numerical approximations integrate fluxes sequentially and can be useful in cases such as large systems of partial differential equations, where computational speed would otherwise be a limiting factor (Fenicia et al., 2011). Sequential calculation of model fluxes is common practice in many hydrologic models (e.g. SACRAMENTO and GR4J) but this approach assumes that fluxes occur in a pre-determined order. It is preferable to integrate model fluxes simultaneously to avoid "physically unsatisfying assumption[s]" (Fenicia et al., 2011; Santos et al., 2018). MARRMoT follows this recommendation, barring certain cases where the model is divided into two distinct parts due to a delay function, in which case simultaneous solving of the first and second part of the model is impossible.

3 MARRMoT

MARRMoT provides Matlab code for 46 conceptual models following the good model development practices outlined in Section 2. This section provides a summary of the framework because it is infeasible to discuss every individual model here. References to the Supporting Materials guide the interested reader to a more in-depth discussion of each model and its implementation in MARRMoT. In addition to this paper, the MARRMoT documentation includes the following:

- Supporting Material S2 - Model descriptions. This document contains descriptions of all 46 models in a standardized format. Each description includes a short introduction to the model, a list of parameters, a model schematic and a discussion of the ODEs and constitutive functions that describe the model's storage changes and fluxes.
- Supporting Material S3 - Flux equation code. This document contains an overview of the 105 different flux equations used in MARRMoT, and their implementation as computer code.
- Supporting Material S4 - Unit Hydrograph overview. This document contains an overview of the 8 different Unit Hydrograph routing schemes used in MARRMoT.
- Supporting Material S5 - Parameter ranges. This document contains an overview of recommended parameter ranges for the 46 models based on published literature about hydrologic process and model application studies. The ranges are standardized across models, so that similar processes use similar parameter ranges. Use of the recommended ranges is optional.
- User Manual: This document helps a user set up MARRMoT for use in either Matlab or Octave, outlines the inner workings of the standardized models, provides several workflow examples and provides examples on how to create a new flux equation or model.

3.1 General MARRMoT outline

Figure 1 shows the setup of the MARRMoT framework and what the framework requires (i.e. data, model options, etc.) and provides for a given modelling study. Each model has its own separate model function, which contains both the numerical implementation of the model (i.e. the ODEs and fluxes that make up this model, as given in Supporting Material S2, S3 and S4) and the necessary code to handle user input, run the model to produce a time series and generate output. The user is expected to provide the following inputs: time series of climate variables, initial values for each model store, choice of numerical integration method and settings for Matlab solvers, and values for each model parameter. Note that the solver selection relates to time-stepping numerics, not parameter selection / optimisation. Optionally, MARRMoT's provided parameter range guidance (Supporting Material S5**)** can inform the choice of parameter values. Parameter ranges have been standardized as much as possible across all models, such that similar processes use the same range of possible parameter values across models (e.g. this ensures that all models that have an interception component with a maximum capacity can use the same range, 0-5mm, for their respective interception capacity parameter). Each model generates a time series of total simulated flow and total simulated evaporation as default output. Optionally, users can request variables with time series of storages and internal fluxes, as well as a summary of the main water balance components. The User Manual provides several workflow examples that showcase possible uses of MARRMoT: the examples cover (i) application of a single model, with a single parameter set to a single catchment, (ii) random parameter sampling from provided parameter ranges for a single model, (iii) application of three different models to a single catchment, and (iv) calibration of a single parameter set for a single model. These examples can easily be adapted to work with multiple catchments if desired.

The basic building blocks inside each model function are flux functions. Each flux function describes a single flux, for example evaporation from an interception store, water exchange between two soil moisture stores or baseflow from groundwater. Flux functions are kept separate from the model functions, and each model calls several flux functions as needed. This allows for consistency across models (if errors are present in any flux function, at least they are the same in all models), easy implementation of new flux equations and facilitation of studies that are specifically interested in differences between various mathematical equations that all represent the same flux or process. The inputs required, and output returned by each flux function varies. See Supporting Material S3 for a full overview of the mathematical functions used to represent fluxes in each model description, relevant constraints, numerical implementation of each flux in MARRMoT and a list of models that use each flux function). Various models use a Unit Hydrograph approach to delay flows within the model and/or simulate flow routing. See Supporting Material S4 for a full overview of Unit Hydrographs currently implemented in MARRMoT.

3.2 Summary of included models

Table 1 shows an overview of model structures currently implemented in MARRMoT and the main reference(s) that these model structures are based on (see section 5.3.3 for a discussion of the comparability of MARRMoT models and their original counterparts). Some of the source models have a long history of application, others are part of model comparison or development studies. MARRMoT development was not guided by a specific modelling objective (e.g. droughts, floods) and the current selection of model structures mainly aims for variety in the range of model structures. The User Manual provides guidance on changing and expanding the framework and, due to its open nature, these additions can be shared with the wider community. Each model is internally different from the others, either through using different configurations of stores and their connections, or through using different flux equations, or both. Models with sequential numbering (e.g. mopex1, mopex2) are part of the same study and tend to be similar but more elaborate as the number increases. Detailed model descriptions can be found in Supporting Material S2. The model code as currently provided was extensively checked for water balance errors during development, using multiple parameter sets for each model, both randomly sampled and using all combinations of extreme values using MARRMoT’s provided parameter ranges. These errors were generally in the order of 1E-12 or smaller, showing that the water balance is properly accounted for in each model.

Figure 2 provides a summarized overview of the model differences, expressed through the number of stores, number of parameters and hydrological processes represented. Models use between 1 and 8 stores, and between 1 and 23 parameters. The number of parameters tends to increase with the number of stores, but exceptions exist. Most models’ stores are used to track moisture availability (i.e. across all models 162 stores are used, 155 of which track moisture availability); deficit stores are much rarer (i.e. only 7 out of 162 stores are used to track moisture deficit). Soil moisture storage is the most commonly modelled concept, occurring in every model. Routing stores (e.g. “fast flow routing”) are included in 18 models, groundwater stores in 13 models, snow storage in 12, interception in 10, unit hydrograph routing also in 10, surface depression storage in 2 and channel storage in 1 model. However, these numbers should not be seen as representative of all conceptual models, because our model overview is necessarily incomplete and some of our models are part of model development studies (where a model is modified until satisfactory performance is obtained). These studies skew the number of stores in certain categories.

4 46 model application test case

To demonstrate the potential of the framework, we calibrated all 46 MARRMoT models to flow observations at Hickory Creek near Brownstown, Illinois (USGS ID: 05592575). This catchment was randomly selected from the CAMELS data set (Addor et al., 2017). The catchment is small with an area of approximately 115 km^2, located at 176 m.a.s.l. at latitude 38.9°. It has a strong seasonal cycle with temperatures varying between -20°C in extreme winters, up to nearly 30°C in summers. Average annual rainfall is approximately 1117mm, 6.4% of which occurs as snowfall. The runoff ratio is around 29% of precipitation. The flow regime is flashy (baseflow index is 0.18) and ephemeral (no flow is observed 18% of the time), High flows (95th percentile flow is 3.7mm/d) are more common in winter and spring, while low flows (5th percentile flow is 0mm/d) are more common in summer and autumn. Soils are a mixture of silt (60%), clay (24%) and sand (16%).

PET input was estimated using climate data included in CAMELS and the Priestley-Taylor method (Priestley and Taylor, 1972). Model calibration uses the time period 1989-1998, model evaluation uses the period 1999-2009. Initial states are found by iteratively running each model with data from the year 1989, until model states reach an equilibrium. The calibration algorithm is the Covariance Matrix Adaptation Evolution Strategy (CMA-ES, Hansen et al., 2003), using the Kling-Gupta Efficiency (Gupta et al., 2009) as the objective function. CMA-ES optimizes a single parameter set per model using MARRMoT’s provided parameter ranges. Note that parameter optimization and sampling are currently not part of the provided tools but connecting MARRMoT to various calibration algorithms or Monte Carlo sampling strategies is straightforward (the User Manual provides several basic workflow examples).

Figure 3a shows KGE values during calibration and evaluation for each model. Each result is coloured to indicate the number of calibrated parameters. The number of model parameters seems unrelated to model performance and several models with higher numbers of parameters are outperformed by the simplest 1-parameter bucket model. After analysing the components present in most successful models (not shown), we can speculate that a saturation excess mechanism is key to achieve satisfactory calibration efficiency values in this catchment, and that this catchment’s flashy behaviour could be related to rainfall events on soil with low available storage.

Figure 3b shows values for two common hydrologic signatures, calculated for time series of simulated flow by each model (blue/white dots, shading showing the KGE value during calibration) and for observations (red dot). These signatures are calculated for the calibration period. There is significant scatter around the observed signature values and models with “good” calibration efficiency (darker shades) are not necessarily closer to observed signature values than models with lower calibration performance. From this we can conclude that even though certain model structures can achieve “high” values for a given objective function, there is no guarantee that the simulated flow series have the same statistical properties as the observed time series the models were calibrated against. Furthermore, this shows that a saturation-excess model can achieve high efficiency values, but that the full hydrologic behaviour in this catchment is likely more nuanced than a single runoff generation mechanism.

Note that our findings in this test case are not new, but this test case highlights the power of multi-model comparison frameworks: from two simple plots we have deduced a plausible important runoff mechanism in this catchment, found that this mechanism alone cannot satisfactorily explain the catchment’s hydrologic behaviour, and that a higher number of model parameters does not necessarily result in more realistic or better performing models. Further investigation of the model structures and their performance could lead us to more insights about hydrologic behaviour and inter-model differences, but that is beyond the scope of this test case.

5 Discussion

5.1 Encouraging debate about reproducibility

Reproducibility of computational hydrology is rarely achieved, primarily because data and code are not regularly made available (Hutton et al., 2016). In the case of hydrologic models, this results in many different versions of the same model being in circulation, made either by different people with different interpretations of the original publication and/or including their own model variant. Without publicly available code, only stating a model’s name in a study is insufficient for knowing which equations and numerical methods make up that particular instance of the model. Conclusions from any modelling study are thus conditional on a certain set of equations that are unknown to the reader, which makes generalizability of findings low. However, there is a trend in hydrology towards open and shareable research. Large-scale hydrologic datasets (e.g. CAMELS (Addor et al., 2017), CAMELS-CL (Alvarez-Garreton et al., 2018), GSIM (Do et al., 2018; Gudmundsson et al., 2018)) are commonly made available and certain journals already enforce better coding and sharing practices. Much work is being done on benchmarking data uncertainty (e.g. McMillan et al., 2012) and model performance (e.g. Seibert et al., 2018) which encourages objective conclusions about the strengths and weaknesses of any model and investigation. By making a multi-model toolbox based on various established models available as open source code, we hope to contribute to this trend of more transparent and reproducible science. Furthermore, this toolbox lowers the threshold for model comparison studies and can help to diminish “legacy” reasons for model application (i.e. choosing to use a certain model for reasons other than the model’s perceived appropriateness for the task at hand, such as convenience or past experience; Addor and Melsen, 2019).

5.2 The state of conceptual hydrologic models

Our model overview (Supporting Material S2) and compilation of these models in a single framework allows unique lessons and insights into the current state of conceptual models (conditional on the sample of model structures we have selected).

The core of this selection of conceptual models is a soil moisture accounting (SMA) module. Every model includes some form of soil moisture store where moisture is kept and evaporated from. Despite this, surface processes, rather than those in the subsurface (both vadose and groundwater zones), tend to be modelled in the greatest detail. For example, intricate snow (e.g. Lindström et al., 1997; Schaefli et al., 2005), interception (e.g. Fukushima, 1988) and surface depression storage (e.g. Chiew and McMahon, 1994; Leavesley et al., 1983; Markstrom et al., 2015) conceptualizations exist among the models, but subsurface processes tend to be much more abstract. This is the same observation as made in Vinogradov et al. (2011). This is understandable because surface processes are easier to observe and formulate hypotheses about, but the subsurface is a crucial component in the water balance (as evidenced by the presence of a SMA component in every single model). A next step in conceptual modelling can be to explicitly formulate hypotheses of subsurface catchment configurations and testing these. For example, the ‘fill-and-spill’ hypothesis (Tromp-Van Meerveld and McDonnell, 2006) could be compared to more traditional subsurface conceptualizations such as linear reservoirs. Framing research as testing alternative hypotheses (Clark et al., 2011) and using modelling tools such as MARRMoT allows testing of these ideas in a controlled manner.

A striking difference exists among models that take evaporation from multiple stores. Certain models use the potential evapotranspiration (PET) rate to limit evaporation from each individual store (e.g. MODHYDROLOG (Chiew and McMahon, 1994), NAM (Nielsen and Hansen, 1973), HYCYMODEL (Fukushima, 1988)), whereas others use PET as the maximum that can be evaporated from all stores combined (e.g. ECHO (Schaefli et al., 2014), PRMS (Leavesley et al., 1983; Markstrom et al., 2015), CLASSIC (Crooks and Naden, 2007)). This can lead to situations where a model evaporates water at a net rate higher than PET. Depending on the way PET is estimated (see e.g. McMahon et al. (2013) for an overview of PET estimation methods) and which reference crop is used compared to the vegetation in the catchment being modelled, either assumption might be appropriate. Evaporation is a significant component of the water balance (McMahon et al., 2013) and a proper choice in any modelling effort is thus important.

Another difference is the distinction between process-aggregated and process-explicit models. Process-aggregated models (e.g. GR4J (Perrin et al., 2003), IHACRES (Croke and Jakeman, 2004; Littlewood et al., 1997)) do not attempt to model individual hydrologic processes but focus on the flows resulting from an aggregation of overall catchment behaviour. Process-explicit models (e.g. MODHYDROLOG (Chiew and McMahon, 1994), FLEX-Topo (Savenije, 2010)) explicitly include a variety of hydrologic processes deemed important for a certain modelling purpose. Process-aggregated models tend to have a small number of parameters which can be preferable when calibrating a model to streamflow only. Process-explicit models are more intuitive when simulating changing conditions due to their explicit process representation, under the strong assumption that the model’s equations and parameters can be related to the real-world processes the model intends to simulate.

Summarizing, even within this subset of all hydrologic models, conceptual models exist in a wide variety of shapes and sizes. They are easy-to-use tools to test whether detailed findings from experimental catchments are applicable to many different catchment types world-wide. This approach combines the thorough understanding developed in well-monitored catchments with the ability to generalise conclusions through extensive testing of these findings in other places.

5.3 MARRMoT considerations

5.3.1 Reliance on imperfect methods

MARRMoT uses built-in Matlab root-finding methods to solve the ODE approximations on every time step. Currently, *fzero* is the default option for models with one store and *fsolve* is the default in multi-store models. *lsqnonlin* is used as a slower but more robust alternative if the former methods are not sufficiently accurate (compared to a user-specified accuracy tolerance). In most cases, this setup performs within acceptable bounds of accuracy. However, for special cases (e.g. very small maximum storage values), the root-finding method might return solutions that are outside the bounds of expected model behaviour (e.g. storages values below 0, storages higher than their maximum capacity or complex numbers), even if "realistic" solutions also exist. Additional constraints must be introduced into the flux equations to prevent this behaviour, because in a large-sample study these issues are difficult to troubleshoot if they occur during the sampling of several thousands of combinations of models and catchments. This involves a fundamental change to model equations necessitated by the use of these solvers. More robust solvers such as *lsqnonlin* allow specification of bounds to the solution space but are less computationally efficient. The current trade-off favours constraints implemented into the fluxes and default use of faster root-finding methods over the more elegant, but much slower, solution provided by *lsqnonlin.* Further optimization of the root-finding methods is considered outside the scope of this version of MARRMoT. Note that settings for these root-finding methods are specified within each model file because certain settings are model-dependent. Progress display is disabled for all three functions (*fzero*, *fsolve*, *lsqnonlin*) by default but can be enabled by the user. The model-dependent Jacobian matrix is specified for *fsolve* and *lsqnonlin*. The maximum number of function evaluations is capped at 1000 for *lsqnonlin*. All other root-finding options are left at default Matlab values (see Matlab documentation of the root-finding methods for further details). Users are encouraged to experiment with these settings to find those that work for their specific problem.

5.3.2 Speed versus readability

Several considerations during MARRMoT design have been heavily influenced by readability and user-friendliness over computational efficiency. Implementing fluxes as anonymous functions rather than regular functions leads to reduced computational speed but increased clarity of the code.

Matlab was chosen out of similar concerns. Fortran or similar compiled language would grant significant speed-ups but reduce user-friendliness.

5.3.3 Correspondence between MARRMoT and original publications

During MARRMoT development, we have tried to stay close to the original publications that introduced the models. Differences are unavoidable however, due to our criteria of creating a uniform framework. Most changes have to do with spatial discretization, where we reduced the level of detail in a model to make all 46 models lumped.

For certain models (e.g. SACRAMENTO (Burnash, 1995; National Weather Service, 2005)) model code and numerical implementation are so interwoven that far-reaching changes were required to make these models fit into this generalized framework. For all models, it is likely that the use of the default Implicit Euler scheme will provide different results to previous studies that use the (much more common) Explicit Euler scheme. Furthermore, the smoothing of model equations will also cause differences to arise with previous studies. We strongly recommend readers to compare the original publication of each model with the version given in this toolbox, to place results from the MARRMoT models in a proper context of earlier work with these models. We emphasize that our models are based on publications that describe existing models, not on existing computer code. Thus, we neither guarantee nor expect that our code performs exactly like the original version of each model's code (if indeed such a version exists and can be found and agreed upon for any given model). To illustrate this point, we compare performance of MARRMoT model m07 (based on the GR4J model) with the R implementation of GR4J (part of the airGR package; Coron et al., 2017, 2019), and we compare MARRMoT model m37 (based on HBV-96) with HBV Light (Seibert and Vis, 2012). MARRMoT m07 is an example of a model that has changed significantly from the original source as a result of combining the original documentation (Perrin et al., 2003) with a more recent state-space version of GR4J (Santos et al., 2018), while both MARRMoT m37 and HBV Light are similar to HBV-96. We thus expect larger deviations between simulations from MARRMoT m07 and airGR-GR4J than we expect between simulations from MARRMoT m37 and HBV-Light. In both cases, we selected 10000 parameter sets from MARRMoT's parameter ranges through Latin Hypercube sampling. In the case of GR4J, both MARRMoT and airGR versions use the same four parameters. In case of HBV, the MARRMoT version has several additional snow parameters and a capillary rise parameter, while HBV Light has various elevation and input correction factors. These have all been fixed at values that effectively disable their impact on model simulations. We then simulated 5 years of streamflow in the earlier described Hickory Creek using both versions of both models. For comparison purposes, we use the Kling-Gupta Efficiency (KGE; Gupta et al., 2009) to express the similarity between simulations and observations. Figure 4 shows the results of this comparison.

Figure 4a shows that for the best performing parameter set in our sample (in terms of KGE value), the hydrographs generated by MARRMoT m37 and HBV Light are relatively similar. Figures 4c-4e show a decomposition of KGE values into its three constitutive components, that express the linear correlation (KGE_r), the ratio of simulated and observed standard deviations (KGE_a) and the ratio of simulated and observed means (KGE_b) respectively. For a given parameter set, MARRMoT m37 and HBV Light generate simulations that are relatively similar (i.e. close to the 1:1 line). HBV Light tends to produce more variable flows than MARRMoT m37 does (high standard deviation and mean of simulated flows). The reason for this is difficult to investigate because although HBV Light is freely available, its source code is not. Differences between both models' equations and numerical approximation of these equations are likely explanations.

Figure 4b shows that for the best performing parameter set in our sample (in terms of KGE value), the hydrographs generated by MARRMoT m07 and airGR-GR4J are relatively different. Most notable, MARRMoT m07 recessions are much slower and higher than those from airGR-GR4J. Figures 4f-4h indicate that for parameter sets close to the optimal points (i.e. (0,0)), MARRMoT m07 and airGR-GR4J show similar performance. For parameter sets further away from the perfect simulation, MARRMoT m07 shows an increasing tendency to simulate more variable flows (higher standard deviation and mean components) than airGR-GR4J does. However, differences between MARRMoT m07 and airGR-GR4J are not unexpected because MARRMoT m07 also uses equations from state-space GR4J (Santos et al., 2018) and the models' equations are thus not identical.

Concluding, we emphasize again that MARRMoT models are based on existing publications only and not on computer code. Differences with other models using the same name are unavoidable. We hope that by making MARRMoT available as open source code, future studies can go beyond simply stating the model name without publishing any model code, and instead can refer to an open-source, traceable version of the model(s) used.

5.3.4 Parameter optimization and sampling

MARRMoT provides model code and recommended parameter ranges but does not include any parameter optimisation, parameter sampling or sensitivity analysis methods. This is a conscious choice because these methods continue to be developed and keeping a latest, state-of-the-art version of each packaged in the MARRMoT distribution is infeasible. We refer the reader to e.g. Arsenault et. al. (2014) for a recent discussion of various optimization methods, to e.g. Beven and Binley (2014) for a recent discussion of GLUE-based uncertainty analysis and to e.g. Pianosi et. al. (2015) for a recent publication of an open-source sensitivity analysis toolbox. Application of any of these methods with MARRMoT models is straightforward. The User Manual provides workflow examples for parameter sampling and parameter calibration, which can be used as a starting point to integrate parameter optimization, sampling or sensitivity analysis methods.

5.3.5 Possible extensions

Lists of contemporary relevant hydrologic models are hard to come by. Such a list would always be incomplete because new models and model variants continue to be developed. As such, there is no reason to assume that the current 46 models in MARRMoT showcase all possible lumped conceptual hydrologic models. Likewise, although MARRMoT includes a wide variety of flux equations, this list should not be assumed to be complete. The MARRMoT User Manual therefore provides detailed guidance on creating new model and flux functions, and the code's location and licensing on Github allows these new models to be shared freely. Extensions to the framework are thus possible and encouraged.

Currently lacking in the code is the possibility to use adaptive time stepping. Fixed-step Implicit Euler approximations are sufficiently accurate for most applications (Clark and Kavetski, 2010; Kavetski and Clark, 2010; Schoups et al., 2010) but adaptive time-stepping can provide additional benefits (Clark et al., 2008; Kavetski and Clark, 2011; Schoups et al., 2010). Our initial assessment is that it would be relatively straightforward to replace the current fixed-step time-stepping implementation with adaptive time-stepping (see e.g. Clark and Kavetski (2010) for further reading on adaptive time-stepping).

6 Conclusions

This paper introduces the Modular Assessment of Rainfall-Runoff Models Toolbox (MARRMoT). This modelling framework is based on a review of conceptual hydrologic models. Across these models, over 100 different flux equations and 8 different Unit Hydrographs (UHs) are used. These are implemented as separate functions and each model draws from this library to select the fluxes and UHs it needs. This results in standardized implementations of 46 unique, lumped model structures. The framework is implemented in Matlab, can be used in Octave, and is provided as open source software (https://github.com/wknoben/MARRMoT ; DOI: 10.5281/zenodo.3235664). Requirements for running a model are simple: (i) time series of precipitation, potential evapotranspiration and optionally temperature, (ii) initial storage values, (iii) settings that specify the numerical integration method (currently provided are Implicit Euler (recommended) and Explicit Euler) and Matlab solver behaviour, and (iv) values for the model parameters (these can be sampled or optimized from parameter ranges provided as part of MARRMoT). MARRMoT comes with documentation that describes (i) each model and its equations, (ii) the conversion from model equations to computer code, (iii) the implementation of 8 different types of Unit Hydrographs, and (iv) the references used to inform standardized parameter ranges,. The User Manual provides guidance on navigating the Matlab functions in which each model is implemented, several examples of how the framework can be used (with workflow scripts that show the Matlab code required for these analyses), information on how to create new models or flux functions, and several small modifications that can speed up the model code by disabling certain output messages from Matlab’s built-in solvers. The main purpose of MARRMoT is to enable multi-model comparison studies and objective testing of model hypotheses. Additional benefits can be gained from the framework’s documentation, which provides an easy-to-navigate comparison of 46 unique conceptual hydrologic models. MARRMoT is provided to the community in the hopes that it will be useful and to encourage a growing trend of open and reproducible science.

7 Code availability and dependencies

MARRMoT is provided under the terms of the GNU General Public License version 3.0. MARRMoT code and User Manual can be downloaded from https://github.com/wknoben/MARRMoT (DOI: 10.5281/zenodo.3235664). Additional documentation can be found in the Supplementary Materials to this paper. MARRMoT has been developed on Matlab version 9.2.0.538062 (R2017a), with the Optimization Toolbox Version 7.6 (R2017a). The Octave distribution has been tested with Octave 4.4.1 and requires the “optim” package. See the User Manual for some detail regarding running MARRMoT in Octave.

8 Author contribution

This work is part of WK's PhD project at the University of Bristol, supervised by RW and JF. WK, RW and JF developed the idea for this framework during discussions. This idea was further developed in discussions between WK, MP and KF, who provided supervision during WK's visit to the University of Melbourne. WK collected and structured an overview of available models, designed and coded the framework and wrote the original draft and final version of this manuscript and the framework documentation. KF and RW assisted with conceptualization and implementation of time step sizes in the framework. RW, JF, MP and KF reviewed and edited the manuscript and documentation drafts.

9 Competing interests

The authors declare they have no conflict of interest.

10 Acknowledgements

First and foremost, we express our gratitude to all the hydrologic modellers who have chosen to make their models' documentation publicly available. Without their hard work this paper could never have been written. This work was funded by the EPSRC WISE CDT, grant reference number EP/L016214/1. WK's visit to the University of Melbourne was co-funded by the Melbourne School of Engineering Visiting Fellows scheme. We are thankful to Philip Kraft and one anonymous reviewer, whose comments have helped improve this manuscript and the MARRMoT code. We also express our thanks to Sebastian Gnann for his thorough testing of the code and for pointing out a variety of typo's, inconsistencies and possible improvements.

11 References

Addor, N. and Melsen, L. A.: Legacy, Rather Than Adequacy, Drives the Selection of Hydrological Models, Water Resour. Res., 55(1), 378–390, doi:10.1029/2018WR022958, 2019.

Addor, N., Newman, A. J., Mizukami, N. and Clark, M. P.: The CAMELS data set: catchment attributes and meteorology for large-sample studies, Hydrol. Earth Syst. Sci., 21, 5293–5313, doi:10.5194/hess-2017-169, 2017.

Alvarez-Garreton, C., Mendoza, P. A., Boisier, J. P., Addor, N., Galleguillos, M., Zambrano-Bigiarini, M., Lara, A., Puelma, C., Cortes, G., Garreaud, R., McPhee, J. and Ayala, A.: The CAMELS-CL dataset: catchment attributes and meteorology for large sample studies – Chile dataset, Hydrol. Earth Syst. Sci., 22(11), 5817–5846, doi:10.5194/hess-22-5817-2018, 2018.

Andréassian, V., Perrin, C. and Michel, C.: Impact of imperfect potential evapotranspiration knowledge on the efficiency and parameters of watershed models, J. Hydrol., 286(1-4), 19–35, doi:10.1016/j.jhydrol.2003.09.030, 2004.

Andréassian, V., Perrin, C., Berthet, L., Le Moine, N., Lerat, J., Loumagne, C., Oudin, L., Mathevet, T., Ramos, M. H. and

Valéry, A.: Crash tests for a standardized evaluation of hydrological models, Hydrol. Earth Syst. Sci., 13(10), 1757–1764, doi:10.5194/hess-13-1757-2009, 2009.

Arsenault, R., Poulin, A., Côté, P. and Brissette, F.: Comparison of Stochastic Optimization Algorithms in Hydrological Model Calibration, J. Hydrol. Eng., 19(7), 1374–1384, doi:10.1061/(ASCE)HE.1943-5584.0000938, 2014.

Atkinson, S. E., Woods, R. A. and Sivapalan, M.: Climate and landscape controls on water balance model complexity over changing timescales, Water Resour. Res., 38(12), 50–1–50–17, doi:10.1029/2002WR001487, 2002.

Atkinson, S. E., Sivapalan, M., Woods, R. A. and Viney, N. R.: Dominant physical controls on hourly flow predictions and the role of spatial variability: Mahurangi catchment, New Zealand, Adv. Water Resour., 26(3), 219–235, doi:10.1016/S0309-1708(02)00183-5, 2003.

Bai, Y., Wagener, T. and Reed, P.: A top-down framework for watershed model evaluation and selection under uncertainty, Environ. Model. Softw., 24(8), 901–916, doi:10.1016/j.envsoft.2008.12.012, 2009.

Di Baldassarre, G. and Montanari, A.: Uncertainty in river discharge observations: A quantitative analysis, Hydrol. Earth Syst. Sci., 13(6), 913–921, doi:10.5194/hess-13-913-2009, 2009.

Bárdossy, A. and Singh, S. K.: Robust estimation of hydrological model parameters, Hydrol. Earth Syst. Sci., 12(6), 1273–1283, doi:10.5194/hess-12-1273-2008, 2008.

Bathurst, J. C., Ewen, J., Parkin, G., O'Connell, P. E. and Cooper, J. D.: Validation of catchment models for predicting land-use and climate change impacts. 3. Blind validation for internal and outlet responses, J. Hydrol., 287(1-4), 74–94, doi:10.1016/j.jhydrol.2003.09.021, 2004.

Beven, K.: Towards a coherent philosophy for modelling the environment, Proc. R. Soc. London. Ser. A Math. Phys. Eng. Sci., 458(2026), 2465–2484, doi:10.1098/rspa.2002.0986, 2002.

Beven, K.: Environmental modelling: an uncertain future?, Routledge, London., 2009.

Beven, K.: Rainfall-Runoff Modelling: The Primer, 2nd ed., John Wiley and Sons Ltd., 2012.

Beven, K. and Binley, A.: GLUE: 20 years on, Hydrol. Process., 28(24), 5897–5918, doi:10.1002/hyp.10082, 2014.

Beven, K. and Freer, J.: A dynamic topmodel, Hydrol. Process., 15(10), 1993–2011, doi:10.1002/hyp.252, 2001a.

Beven, K. and Freer, J.: Equifinality, data assimilation, and uncertainty estimation in mechanistic modelling of complex environmental systems using the GLUE methodology, J. Hydrol., 249, 11–29, 2001b.

Beven, K., Lamb, R., Quinn, P., Romanowicz, R. and Freer, J.: TOPMODEL, in Computer Models of Watershed Hydrology, edited by V. P. Singh, pp. 627–668, Water Resources Publications, USA, Baton Rouge., 1995.

de Boer-Euser, T., Bouaziz, L., De Niel, J., Brauer, C., Dewals, B., Drogue, G., Fenicia, F., Grelier, B., Nossent, J., Pereira, F., Savenije, H., Thirel, G. and Willems, P.: Looking beyond general metrics for model comparison - lessons from an international model intercomparison study, Hydrol. Earth Syst. Sci., 21(1), 423–440, doi:10.5194/hess-21-423-2017, 2017.

Boyle, D. P.: Multicriteria calibration of hydrologic models, PhD thesis, University of Arizona., 2001.

Burnash, R. J. C.: The NWS River Forecast System - catchment modeling, in Computer Models of Watershed Hydrology, edited by V. P. Singh, pp. 311–366., 1995.

Chiew, F. and McMahon, T.: Application of the daily rainfall-runoff model MODHYDROLOG to 28 Australian catchments, J. Hydrol., 153(1-4), 383–416, doi:10.1016/0022-1694(94)90200-3, 1994.

Chiew, F. H. S.: Estimating groundwater recharge using an integrated surface and groundwater model, University of Melbourne., 1990.

Chiew, F. H. S., Peel, M. C. and Western, A. W.: Application and testing of the simple rainfall-runoff model SIMHYD, in Mathematical Models of Small Watershed Hydrology, edited by V. P. Singh and D. K. Frevert, pp. 335–367, Water Resources Publications LLC, USA, Chelsea, Michigan, USA., 2002.

Clark, M. P. and Kavetski, D.: Ancient numerical daemons of conceptual hydrological modeling: 1. Fidelity and efficiency of time stepping schemes, Water Resour. Res., 46(10), doi:10.1029/2009WR008894, 2010.

Clark, M. P., Slater, A. G., Rupp, D. E., Woods, R. a., Vrugt, J. a., Gupta, H. V., Wagener, T. and Hay, L. E.: Framework for Understanding Structural Errors (FUSE): A modular framework to diagnose differences between hydrological models, Water Resour. Res., 44(12), doi:10.1029/2007WR006735, 2008.

Clark, M. P., Kavetski, D. and Fenicia, F.: Pursuing the method of multiple working hypotheses for hydrological modeling, Water Resour. Res., 47(9), doi:10.1029/2010WR009827, 2011.

Clark, M. P., Nijssen, B., Lundquist, J. D., Kavetski, D., Rupp, D. E., Woods, R. A., Freer, J. E., Gutmann, E. D., Wood, A. W., Brekke, L. D., Arnold, J. R., Gochis, D. J. and Rasmussen, R. M.: A unified approach for process-based hydrologic modeling: 1. Modeling concept, Water Resour. Res., 51(4), 2498–2514, doi:10.1002/2015WR017198, 2015a.

Clark, M. P., Nijssen, B., Lundquist, J. D., Kavetski, D., Rupp, D. E., Woods, R. A., Freer, J. E., Gutmann, E. D., Wood, A. W., Gochis, D. J., Rasmussen, R. M., Tarboton, D. G., Mahat, V., Flerchinger, G. N. and Marks, D. G.: A unified approach for process-based hydrologic modeling: 2. Model implementation and case studies, Water Resour. Res., 51, 2515–2542, doi:10.1002/2015WR017198.A, 2015b.

Coron, L., Andréassian, V., Perrin, C., Lerat, J., Vaze, J., Bourqui, M. and Hendrickx, F.: Crash testing hydrological models in contrasted climate conditions: An experiment on 216 Australian catchments, Water Resour. Res., 48(5), W05552, doi:10.1029/2011WR011721, 2012.

Coron, L., Thirel, G., Delaigue, O., Perrin, C. and Andréassian, V.: The suite of lumped GR hydrological models in an R package, Environ. Model. Softw., 94, 166–171, doi:10.1016/j.envsoft.2017.05.002, 2017.

Coron, L., Delaigue, O., Thirel, G., Perrin, C. and Michel, C.: airGR: Suite of GR Hydrological Models for Precipitation-Runoff Modelling., 2019.

Croke, B. and Jakeman, A.: A catchment moisture deficit module for the IHACRES rainfall-runoff model, Environ. Model. Softw., 19(1), 1–5, doi:10.1016/j.envsoft.2003.09.001, 2004.

Crooks, S. M. and Naden, P. S.: CLASSIC: a semi-distributed rainfall-runoff modelling system, Hydrol. Earth Syst. Sci., 11(1), 516–531, doi:10.5194/hess-11-516-2007, 2007.

Do, H. X., Gudmundsson, L., Leonard, M. and Westra, S.: The Global Streamflow Indices and Metadata Archive (GSIM) – Part 1: The production of a daily streamflow archive and metadata, Earth Syst. Sci. Data, 10(2), 765–785, doi:10.5194/essd-

10-765-2018, 2018.

Eder, G., Sivapalan, M. and Nachtnebel, H. P.: Modelling water balances in an Alpine catchment through exploitation of emergent properties over changing time scales, Hydrol. Process., 17(11), 2125–2149, doi:10.1002/hyp.1325, 2003.

Efstratiadis, A. and Koutsoyiannis, D.: One decade of multi-objective calibration approaches in hydrological modelling: a review, Hydrol. Sci. J., 55(1), 58–78, doi:10.1080/02626660903526292, 2010.

Van Esse, W. R., Perrin, C., Booij, M. J., Augustijn, D. C. M., Fenicia, F., Kavetski, D. and Lobligeois, F.: The influence of conceptual model structure on model performance: A comparative study for 237 French catchments, Hydrol. Earth Syst. Sci., 17(10), 4227–4239, doi:10.5194/hess-17-4227-2013, 2013.

Ewen, J. and Parkin, G.: Validation of catchment models for predicting land-use and climate change impacts. 1. Method, J. Hydrol., 175, 583–594, doi:10.1016/S0022-1694(96)80026-6, 1996.

Farmer, D., Sivapalan, M. and Jothityangkoon, C.: Climate, soil, and vegetation controls upon the variability of water balance in temperate and semiarid landscapes: Downward approach to water balance analysis, Water Resour. Res., 39(2), doi:10.1029/2001WR000328, 2003.

Fenicia, F., McDonnell, J. J. and Savenije, H. H. G.: Learning from model improvement: On the contribution of complementary data to process understanding, Water Resour. Res., 44(6), 1–13, doi:10.1029/2007WR006386, 2008a.

Fenicia, F., Savenije, H. H. G., Matgen, P. and Pfister, L.: Understanding catchment behavior through stepwise model concept improvement, Water Resour. Res., 44(1), doi:10.1029/2006WR005563, 2008b.

Fenicia, F., Kavetski, D. and Savenije, H. H. G.: Elements of a flexible approach for conceptual hydrological modeling: 1. Motivation and theoretical development, Water Resour. Res., 47(11), doi:10.1029/2010WR010174, 2011.

Fenicia, F., Kavetski, D., Savenije, H. H. G., Clark, M. P., Schoups, G., Pfister, L. and Freer, J.: Catchment properties, function, and conceptual model representation: is there a correspondence?, Hydrol. Process., 28(4), 2451–2467, doi:10.1002/hyp.9726, 2014.

Fowler, K. J. A., Peel, M. C., Western, A. W., Zhang, L. and Peterson, T. J.: Simulating runoff under changing climatic conditions: Revisiting an apparent deficiency of conceptual rainfall-runoff models, Water Resour. Res., 52, 1820–1846, doi:10.1002/2015WR018068, 2016.

Freer, J. E., McMillan, H., McDonnell, J. J. and Beven, K. J.: Constraining dynamic TOPMODEL responses for imprecise water table information using fuzzy rule based performance measures, J. Hydrol., 291(3-4), 254–277, doi:10.1016/j.jhydrol.2003.12.037, 2004.

Fukushima, Y.: A model of river flow forecasting for a small forested mountain catchment, Hydrol. Process., 2(2), 167–185, 1988.

Goswami, M. and O’Connor, K. M.: A “monster” that made the SMAR conceptual model “right for the wrong reasons,” Hydrol. Sci. J., 55(6), 913–927, doi:10.1080/02626667.2010.505170, 2010.

Gudmundsson, L., Do, H. X., Leonard, M. and Westra, S.: The Global Streamflow Indices and Metadata Archive (GSIM)-Part 2: Quality control, time-series indices and homogeneity assessment, Earth Syst. Sci. Data, 10(2), 787–804, doi:10.5194/essd-

10-787-2018, 2018.

Gupta, H. V., Kling, H., Yilmaz, K. K. and Martinez, G. F.: Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling, J. Hydrol., 377(1-2), 80–91, doi:10.1016/j.jhydrol.2009.08.003, 2009.

Gupta, H. V., Clark, M. P., Vrugt, J. a., Abramowitz, G. and Ye, M.: Towards a comprehensive assessment of model structural adequacy, Water Resour. Res., 48(8), doi:10.1029/2011WR011044, 2012.

Hansen, N., Müller, S. D. and Koumoutsakos, P.: Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES), Evol. Comput., 11(1), 1–18, doi:10.1162/106365603321828970, 2003.

Hutton, C., Wagener, T., Freer, J., Han, D., Duffy, C. and Arheimer, B.: Most computational hydrology is not reproducible, so is it really science?, Water Resour. Res., 52(10), 7548–7555, doi:10.1002/2016WR019285, 2016.

Jothityangkoon, C., Sivapalan, M. and Farmer, D. .: Process controls of water balance variability in a large semi-arid catchment: downward approach to hydrological model development, J. Hydrol., 254(1-4), 174–198, doi:10.1016/S0022-1694(01)00496-6, 2001.

Kavetski, D. and Clark, M. P.: Ancient numerical daemons of conceptual hydrological modeling: 2. Impact of time stepping schemes on model analysis and prediction, Water Resour. Res., 46(10), 1–27, doi:10.1029/2009WR008896, 2010.

Kavetski, D. and Clark, M. P.: Numerical troubles in conceptual hydrology: Approximations, absurdities and impact on hypothesis testing, Hydrol. Process., 25(4), 661–670, doi:10.1002/hyp.7899, 2011.

Kavetski, D. and Fenicia, F.: Elements of a flexible approach for conceptual hydrological modeling: 2. Application and experimental insights, Water Resour. Res., 47(11), n/a–n/a, doi:10.1029/2011WR010748, 2011.

Kavetski, D. and Kuczera, G.: Model smoothing strategies to remove microscale discontinuities and spurious secondary optima in objective functions in hydrological calibration, Water Resour. Res., 43(3), n/a–n/a, doi:10.1029/2006WR005195, 2007.

Kavetski, D., Kuczera, G. and Franks, S. W.: Semidistributed hydrological modeling: A “saturation path” perspective on TOPMODEL and VIC, Water Resour. Res., 39(9), n/a–n/a, doi:10.1029/2003WR002122, 2003.

Kavetski, D., Kuczera, G. and Franks, S. W.: Calibration of conceptual hydrological models revisited: 1. Overcoming numerical artefacts, J. Hydrol., 320(1-2), 173–186, doi:10.1016/j.jhydrol.2005.07.012, 2006.

Kirchner, J. W.: Getting the right answers for the right reasons: Linking measurements, analyses, and models to advance the science of hydrology, Water Resour. Res., 42(3), doi:10.1029/2005WR004362, 2006.

Kirchner, J. W.: Aggregation in environmental systems: catchment mean transit times and young water fractions under hydrologic nonstationarity, Hydrol. Earth Syst. Sci. Discuss., 12(3), 3105–3167, doi:10.5194/hessd-12-3105-2015, 2016.

Klemeš, V.: Operational testing of hydrological simulation models, Hydrol. Sci. J., 31(1), 13–24, doi:10.1080/02626668609491024, 1986.

Kraft, P., Vaché, K. B., Frede, H.-G. and Breuer, L.: CMF: A Hydrological Programming Language Extension For Integrated Catchment Models, Environ. Model. Softw., 26(6), 828–830, doi:10.1016/j.envsoft.2010.12.009, 2011.

Krueger, T., Freer, J., Quinton, J. N., Macleod, C. J. A., Bilotta, G. S., Brazier, R. E., Butler, P. and Haygarth, P. M.: Ensemble evaluation of hydrological model hypotheses, Water Resour. Res., 46(7), doi:10.1029/2009WR007845, 2010.

Leavesley, G. H., Lichty, R. W., Troutman, B. M. and Saindon, L. G.: Precipitation-Runoff Modeling System: User's Manual, U.S. Geol. Surv. Water-Resources Investig. Rep. 83-4238, 207, 1983.

Leavesley, G. H., Restrepo, P. J., Markstrom, S. L., Dixon, M. and Stannard, L. G.: The Modular Modeling System - MMS, User's Manual, Denver, Col., 1996.

Liang, X., Lettenmaier, D. P., Wood, E. F. and Burges, S. J.: A simple hydrologically based model of land surface water and energy fluxes for general circulation models, J. Geophys. Res., 99, 14415–14428, 1994.

Lindström, G., Johansson, B., Persson, M., Gardelin, M. and Bergström, S.: Development and test of the distributed HBV-96 hydrological model, J. Hydrol., 201, 272–288, doi:https://doi.org/10.1016/S0022-1694(97)00041-3, 1997.

Littlewood, I. G., Down, K., Parker, J. R. and Post, D. A.: IHACRES v1.0 User Guide., 1997.

Markstrom, S. L., Regan, S., Hay, L. E., Viger, R. J., Webb, R. M. T., Payn, R. A. and LaFontaine, J. H.: PRMS-IV, the Precipitation-Runoff Modeling System, Version 4, in U.S. Geological Survey Techniques and Methods, book 6, chap. B7, p. 158., 2015.

McMahon, T. A., Peel, M. C., Lowe, L., Srikanthan, R. and McVicar, T. R.: Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: A pragmatic synthesis, Hydrol. Earth Syst. Sci., 17(4), 1331–1363, doi:10.5194/hess-17-1331-2013, 2013.

McMillan, H., Freer, J., Pappenberger, F., Krueger, T. and Clark, M.: Impacts of uncertain river flow data on rainfall-runoff model calibration and discharge predictions, Hydrol. Process., 24(10), 1270–1284, doi:10.1002/hyp.7587, 2010.

McMillan, H., Krueger, T. and Freer, J.: Benchmarking observational uncertainties for hydrology: rainfall, river discharge and water quality, Hydrol. Process., 26(26), 4078–4111, doi:10.1002/hyp.9384, 2012.

Moore, R. J. and Bell, V. A.: Comparison of rainfall-runoff models for flood forecasting. Part 1: Literature review of models, Environment Agency, Bristol., 2001.

Nathan, R. J. and McMahon, T. A.: SFB model part l . Validation of fixed model parameters, in Civil Eng. Trans., pp. 157–161., 1990.

National Weather Service: II.3-SAC-SMA: Conceptualization of the Sacramento Soil Moisture Accounting model, in National Weather Service River Forecast System (NWSRFS) User Manual, pp. 1–13., 2005.

Nielsen, S. A. and Hansen, E.: Numerical simulation of he rainfall-runoff process on a daily basis, Nord. Hydrol., (4), 171–190, doi:https://doi.org/10.2166/nh.1973.0013, 1973.

Nijzink, R., Hutton, C., Pechlivanidis, I., Capell, R., Arheimer, B., Freer, J., Han, D., Wagener, T., McGuire, K., Savenije, H. and Hrachowitz, M.: The evolution of root zone moisture capacities after land use change: a step towards predictions under change?, Hydrol. Earth Syst. Sci. Discuss., 20(August), 4775–4799, doi:10.5194/hess-2016-427, 2016.

O'Connell, P. E., Nash, J. E. and Farrell, J. P.: River flow forecasting through conceptual models part II - the Brosna catchment at Ferbane, J. Hydrol., 10, 317–329, 1970.

Oudin, L., Hervieu, F., Michel, C., Perrin, C., Andréassian, V., Anctil, F. and Loumagne, C.: Which potential evapotranspiration input for a lumped rainfall-runoff model? Part 2 - Towards a simple and efficient potential evapotranspiration model for rainfall-runoff modelling, J. Hydrol., 303(1-4), 290–306, doi:10.1016/j.jhydrol.2004.08.026, 2005.

Oudin, L., Perrin, C., Mathevet, T., Andréassian, V. and Michel, C.: Impact of biased and randomly corrupted inputs on the efficiency and the parameters of watershed models, J. Hydrol., 320(1-2), 62–83, doi:10.1016/j.jhydrol.2005.07.016, 2006.

Pechlivanidis, I. G., Jackson, B. M., McIntyre, N. R. and Wheater, H. S.: Catchment scale hydrological modelling: a review of model types, calibration approaches and uncertainty analysis methods in the context of recent developments in technology and applications, Glob. NEST, 13(3), 193–214, 2011.

Peel, M. C. and Blöschl, G.: Hydrological modelling in a changing world, Prog. Phys. Geogr., 35(2), 249–261, doi:10.1177/0309133311402550, 2011.

Penman, H. L.: The Dependence of Transpiration on Weather and Soil Conditions, J. Soil Sci., 1(1), 74–89, doi:10.1111/j.1365-2389.1950.tb00720.x, 1950.

Perrin, C., Michel, C. and Andréassian, V.: Does a large number of parameters enhance model performance? Comparative assessment of common catchment model structures on 429 catchments, J. Hydrol., 242(3-4), 275–301, doi:10.1016/S0022-1694(00)00393-0, 2001.

Perrin, C., Michel, C. and Andréassian, V.: Improvement of a parsimonious model for streamflow simulation, J. Hydrol., 279(1-4), 275–289, doi:10.1016/S0022-1694(03)00225-7, 2003.

Pianosi, F., Sarrazin, F. and Wagener, T.: SAFE Toolbox v1.0, 2015.

Priestley, C. H. B. and Taylor, R. J.: On the Assessment of Surface Heat Flux and Evaporation Using Large-Scale Parameters, Mon. Weather Rev., 100(2), 81–92, doi:10.1175/1520-0493(1972)100<0081:OTAOSH>2.3.CO;2, 1972.

Refsgaard, J. C. and Henriksen, H. J.: Modelling guidelines - Terminology and guiding principles, Adv. Water Resour., 27(1), 71–82, doi:10.1016/j.advwatres.2003.08.006, 2004.

Santos, L., Thirel, G. and Perrin, C.: Continuous state-space representation of a bucket-type rainfall-runoff model: a case study with the GR4 model using state-space GR4 (version 1.0), Geosci. Model Dev., 11(4), 1591–1605, doi:10.5194/gmd-11-1591-2018, 2018.

Savenije, H. H. G.: “Topography driven conceptual modelling (FLEX-Topo),” Hydrol. Earth Syst. Sci., 14(12), 2681–2692, doi:10.5194/hess-14-2681-2010, 2010.

Schaefli, B., Hingray, B., Niggli, M. and Musy, A.: A conceptual glacio-hydrological model for high mountainous catchments, Hydrol. Earth Syst. Sci., 9(1/2), 95–109, doi:10.5194/hess-9-95-2005, 2005.

Schaefli, B., Nicotina, L., Imfeld, C., Da Ronco, P., Bertuzzo, E. and Rinaldo, A.: SEHR-ECHO v1.0: A spatially explicit hydrologic response model for ecohydrologic applications, Geosci. Model Dev., 7(6), 2733–2746, doi:10.5194/gmd-7-2733-2014, 2014.

Schoups, G., Vrugt, J. A., Fenicia, F. and Van De Giesen, N. C.: Corruption of accuracy and efficiency of Markov chain Monte Carlo simulation by inaccurate numerical implementation of conceptual hydrologic models, Water Resour. Res., 46(10), doi:10.1029/2009WR008648, 2010.

Seibert, J. and van Meerveld, H. J. I.: Hydrological change modeling: Challenges and opportunities, Hydrol. Process., 30(26), 4966–4971, doi:10.1002/hyp.10999, 2016.

Seibert, J. and Vis, M. J. P.: Teaching hydrological modeling with a user-friendly catchment-runoff-model software package, Hydrol. Earth Syst. Sci., 16(9), 3315–3325, doi:10.5194/hess-16-3315-2012, 2012.

Seibert, J., Vis, M. J. P., Lewis, E. and van Meerveld, H. J.: Upper and lower benchmarks in hydrological modelling, Hydrol. Process., (February), 1120–1125, doi:10.1002/hyp.11476, 2018.

Singh, V. P. and Woolhiser, D. A.: Mathematical Modeling of Watershed Hydrology, J. Hydrol. Eng., 7(4), 270–292, doi:10.1061/(ASCE)1084-0699(2002)7:4(270), 2002.

Sivapalan, M., Ruprecht, J. K. and Viney, N. R.: Water and salt balance modelling to predict the effects of land-use changes in forested catchments. 1. Small catchment water balance model, Hydrol. Process., 10(3), 1996.

Son, K. and Sivapalan, M.: Improving model structure and reducing parameter uncertainty in conceptual water balance models through the use of auxiliary data, Water Resour. Res., 43(1), doi:10.1029/2006WR005032, 2007.

Sugawara, M.: Automatic calibration of the tank model, Hydrol. Sci. Bull., 24(3), 375–388, doi:10.1080/02626667909491876, 1979.

Sugawara, M.: Tank model, in Computer models of watershed hydrology, edited by V. P. Singh, pp. 165–214, Water Resources Publications, USA., 1995.

Tan, B. Q. and O’Connor, K. M.: Application of an empirical infiltration equation in the SMAR conceptual model, J. Hydrol., 185(1-4), 275–295, doi:10.1016/0022-1694(95)02993-1, 1996.

Tromp-Van Meerveld, H. J. and McDonnell, J. J.: Threshold relations in subsurface stormflow: 2. The fill and spill hypothesis, Water Resour. Res., 42(2), 1–11, doi:10.1029/2004WR003800, 2006.

Vinogradov, Y. B., Semenova, O. M. and Vinogradova, T. A.: An approach to the scaling problem in hydrological modelling: The deterministic modelling hydrological system, Hydrol. Process., 25(7), 1055–1073, doi:10.1002/hyp.7901, 2011.

Wagener, T., Boyle, D. P., Lees, M. J., Wheater, H. S., Gupta, Hoshin, V. and Sorooshian, S.: A framework for development and application of hydrological models, Hydrol. Earth Syst. Sci., 5, 13–26, 2001.

Wagener, T., Lees, M. J. and Wheater, H. S.: A toolkit for the development and application of parsimonious hydrological models, in Mathematical Models of Small Watershed Hydrology - Volume 2, edited by Singh, Frevert, and Meyer, pp. 91–139, Water Resources Publications LLC, USA., 2002.

Wagener, T., Sivapalan, M., Troch, P. A., McGlynn, B. L., Harman, C. J., Gupta, H. V., Kumar, Rao, P. S. C., Basu, N. B. and Wilson, J. S.: The future of hydrology: An evolving science for a changing world, Water Resour. Res., 46(5), doi:10.1029/2009WR008906, 2010.

Ye, S., Yaeger, M., Coopersmith, E., Cheng, L. and Sivapalan, M.: Exploring the physical controls of regional patterns of flow duration curves - Part 2: Role of seasonality, the regime curve, and associated process controls, Hydrol. Earth Syst. Sci., 16(11), 4447–4465, doi:10.5194/hess-16-4447-2012, 2012.

Ye, W., Bates, B. C., Viney, N. R. and Sivapalan, M.: Performance of conceptual rainfall-runoff models in low-yielding ephemeral catchments, Water Resour. Res., 33(1), 153–166, doi:doi:10.1029/96WR02840, 1997.

Zhao, R.-J.: The Xinanjiang model applied in China, J. Hydrol., 135(1-4), 371–381, doi:10.1016/0022-1694(92)90096-E, 1992.

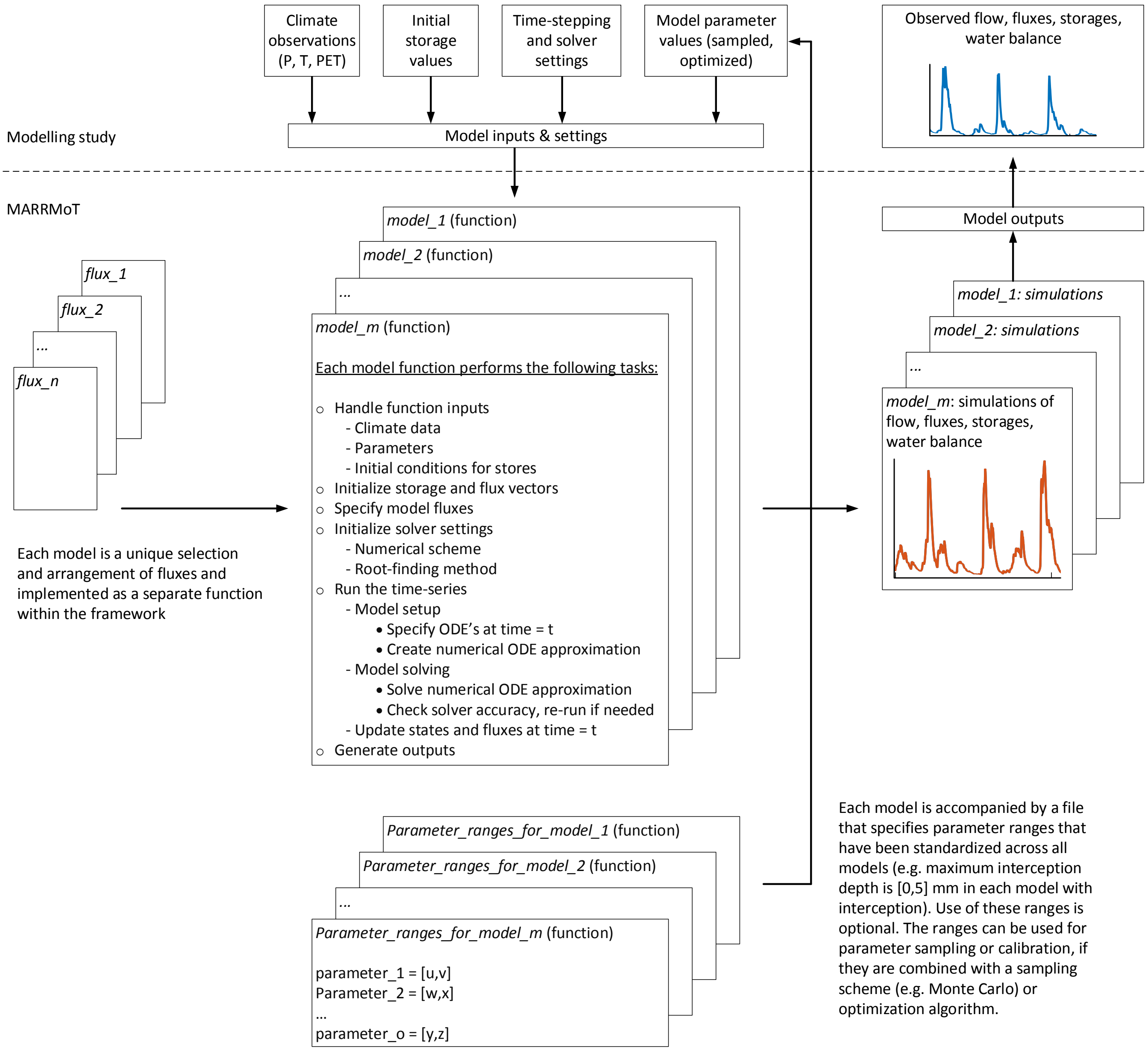
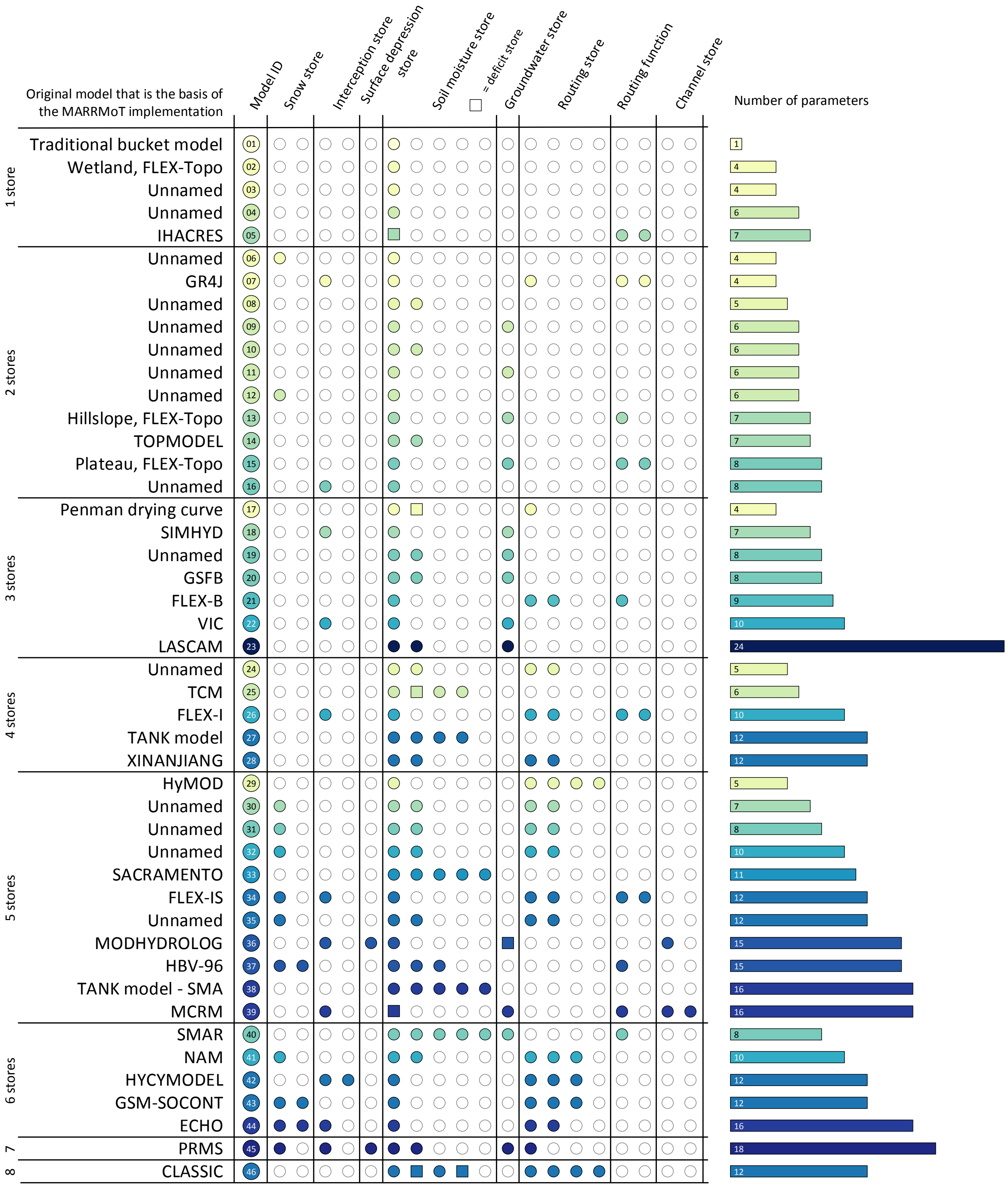

Climate observations (P, T, PET)
Initial storage values
Time-stepping and solver settings
Model parameter values (sampled, optimized)
Observed flow, fluxes, storages, water balance
Modelling study
Model inputs & settings
MARRMoT
Model outputs
model_1 (function)
model_2 (function)
...
model_m (function)
Each model function performs the following tasks:
○ Handle function inputs
- Climate data
- Parameters
- Initial conditions for stores
○ Initialize storage and flux vectors
○ Specify model fluxes
○ Initialize solver settings
- Numerical scheme
- Root-finding method
○ Run the time-series
- Model setup
• Specify ODE's at time = t
• Create numerical ODE approximation
- Model solving
• Solve numerical ODE approximation
• Check solver accuracy, re-run if needed
- Update states and fluxes at time = t
○ Generate outputs
flux_1
flux_2
...
flux_n
Each model is a unique selection and arrangement of fluxes and implemented as a separate function within the framework
model_1: simulations
model_2: simulations
...
model_m: simulations of flow, fluxes, storages, water balance
Parameter_ranges_for_model_1 (function)
Parameter_ranges_for_model_2 (function)
...
Parameter_ranges_for_model_m (function)
parameter_1 = [u,v]
Parameter_2 = [w,x]
...
parameter_o = [y,z]
Each model is accompanied by a file that specifies parameter ranges that have been standardized across all models (e.g. maximum interception depth is [0,5] mm in each model with interception). Use of these ranges is optional. The ranges can be used for parameter sampling or calibration, if they are combined with a sampling scheme (e.g. Monte Carlo) or optimization algorithm.

Figure 1: Schematic overview of the MARMMoT framework. MARRMoT provides 46 conceptual models implemented in a standardized way (part below the dotted line). Each model is a unique collection and arrangement of fluxes, but the code-wise setup

of each model is the same. Inputs required to run a model are time series of climate variables, values for the model parameters (which can optionally be sampled or optimized using provided, standardized ranges), and initial conditions for each model store. The model returns time series of simulated flow, fluxes and storages and a summary of the simulated water balance.



	Original model that is the basis of the MARRMoT implementation	Model ID	Snow store		Interception store		Surface depression store	Soil moisture store □ = deficit store					Groundwater store	Routing store				Routing function		Channel store		Number of parameters
1 store	Traditional bucket model	01	○	○	○	○	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	1
	Wetland, FLEX-Topo	02	○	○	○	○	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	4
	Unnamed	03	○	○	○	○	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	4
	Unnamed	04	○	○	○	○	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	6
	IHACRES	05	○	○	○	○	○	■	○	○	○	○	○	○	○	○	○	●	●	○	○	7
2 stores	Unnamed	06	●	○	○	○	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	4
	GR4J	07	○	○	●	○	○	●	○	○	○	○	○	●	○	○	○	●	●	○	○	4
	Unnamed	08	○	○	○	○	○	●	●	○	○	○	○	○	○	○	○	○	○	○	○	5
	Unnamed	09	○	○	○	○	○	●	○	○	○	○	●	○	○	○	○	○	○	○	○	6
	Unnamed	10	○	○	○	○	○	●	●	○	○	○	○	○	○	○	○	○	○	○	○	6
	Unnamed	11	○	○	○	○	○	●	○	○	○	○	●	○	○	○	○	○	○	○	○	6
	Unnamed	12	●	○	○	○	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	6
	Hillslope, FLEX-Topo	13	○	○	○	○	○	●	○	○	○	○	●	○	○	○	○	●	○	○	○	7
	TOPMODEL	14	○	○	○	○	○	●	●	○	○	○	○	○	○	○	○	○	○	○	○	7
	Plateau, FLEX-Topo	15	○	○	○	○	○	●	○	○	○	○	●	○	○	○	○	●	●	○	○	8
	Unnamed	16	○	○	●	○	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	8
3 stores	Penman drying curve	17	○	○	○	○	○	●	■	○	○	○	○	●	○	○	○	○	○	○	○	4
	SIMHYD	18	○	○	●	○	○	●	○	○	○	○	●	○	○	○	○	○	○	○	○	7
	Unnamed	19	○	○	○	○	○	●	●	○	○	○	●	○	○	○	○	○	○	○	○	8
	GSFB	20	○	○	○	○	○	●	●	○	○	○	●	○	○	○	○	○	○	○	○	8
	FLEX-B	21	○	○	○	○	○	●	○	○	○	○	○	●	●	○	○	●	○	○	○	9
	VIC	22	○	○	●	○	○	●	○	○	○	○	●	○	○	○	○	○	○	○	○	10
	LASCAM	23	○	○	○	○	○	●	●	○	○	○	●	○	○	○	○	○	○	○	○	24
4 stores	Unnamed	24	○	○	○	○	○	●	●	○	○	○	○	●	●	○	○	○	○	○	○	5
	TCM	25	○	○	○	○	○	●	■	●	●	○	○	○	○	○	○	○	○	○	○	6
	FLEX-I	26	○	○	●	○	○	●	○	○	○	○	○	●	●	○	○	●	●	○	○	10
	TANK model	27	○	○	○	○	○	●	●	●	●	○	○	○	○	○	○	○	○	○	○	12
	XINANJIANG	28	○	○	○	○	○	●	●	○	○	○	○	●	●	○	○	○	○	○	○	12
5 stores	HyMOD	29	○	○	○	○	○	●	○	○	○	○	○	●	●	●	●	○	○	○	○	5
	Unnamed	30	●	○	○	○	○	●	●	○	○	○	○	●	●	○	○	○	○	○	○	7
	Unnamed	31	●	○	○	○	○	●	●	○	○	○	○	●	●	○	○	○	○	○	○	8
	Unnamed	32	●	○	○	○	○	●	●	○	○	○	○	●	●	○	○	○	○	○	○	10
	SACRAMENTO	33	○	○	○	○	○	●	●	●	●	●	○	○	○	○	○	○	○	○	○	11
	FLEX-IS	34	●	○	●	○	○	●	○	○	○	○	○	●	●	○	○	●	●	○	○	12
	Unnamed	35	●	○	○	○	○	●	●	○	○	○	○	●	●	○	○	○	○	○	○	12
	MODHYDROLOG	36	○	○	●	○	●	●	○	○	○	○	■	○	○	○	○	○	○	●	○	15
	HBV-96	37	●	●	○	○	○	●	●	●	○	○	○	○	○	○	○	●	○	○	○	15
	TANK model - SMA	38	○	○	○	○	○	●	●	●	●	●	○	○	○	○	○	○	○	○	○	16
	MCRM	39	○	○	●	○	○	■	○	○	○	○	●	○	○	○	○	●	○	●	●	16
6 stores	SMAR	40	○	○	○	○	○	●	●	●	●	●	●	○	○	○	○	●	○	○	○	8
	NAM	41	●	○	○	○	○	●	●	○	○	○	○	●	●	●	○	○	○	○	○	10
	HYCYMODEL	42	○	○	●	●	○	●	○	○	○	○	○	●	●	●	○	○	○	○	○	12
	GSM-SOCONT	43	●	●	○	○	○	●	○	○	○	○	○	●	●	●	○	○	○	○	○	12
	ECHO	44	●	●	●	○	○	●	○	○	○	○	○	●	●	○	○	○	○	○	○	16
7	PRMS	45	●	○	●	○	●	●	●	○	○	○	●	●	○	○	○	○	○	○	○	18
8	CLASSIC	46	○	○	○	○	○	●	■	●	■	○	○	●	●	●	●	○	○	○	○	12

Figure 2: Overview of MARRMoT models. Models are sorted vertically by number of stores (1 at the top, 8 at the bottom). The columns show broad categories of hydrologic process that can be represented by a model. Coloured circles indicate the model has a store dedicated to the representation of this hydrological process (squares indicate a deficit store). The bar plot on the right shows each model's number of parameters. Colouring refers to the number of parameters.

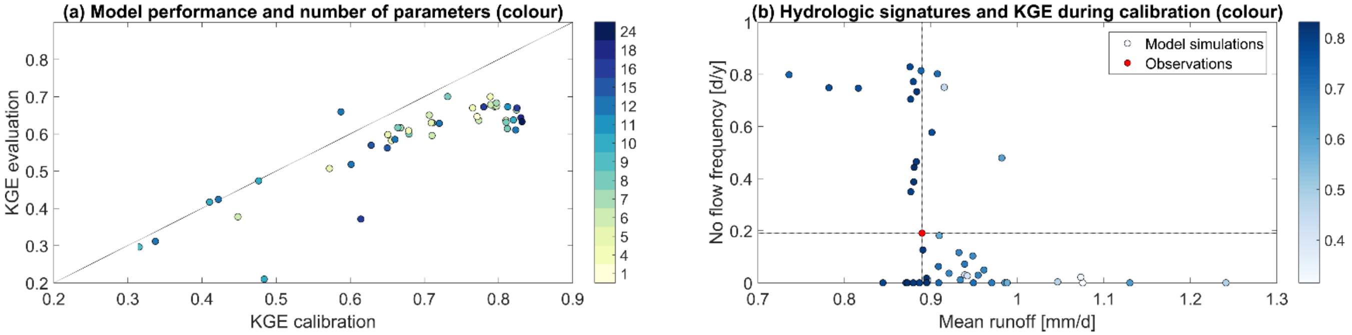


Figure 3: Example of MARRMoT application to Hickory Creek near Brownstown (USA). (a) model performance during calibration (1989-1998) and evaluation (1999-2009) periods. Each dot represents a single model and is coloured according to the model's number of calibrated parameters. (b) Comparison of simulated average flow and no-flow frequency signature values and observed values for those signatures (red dot bisected with lines). 5

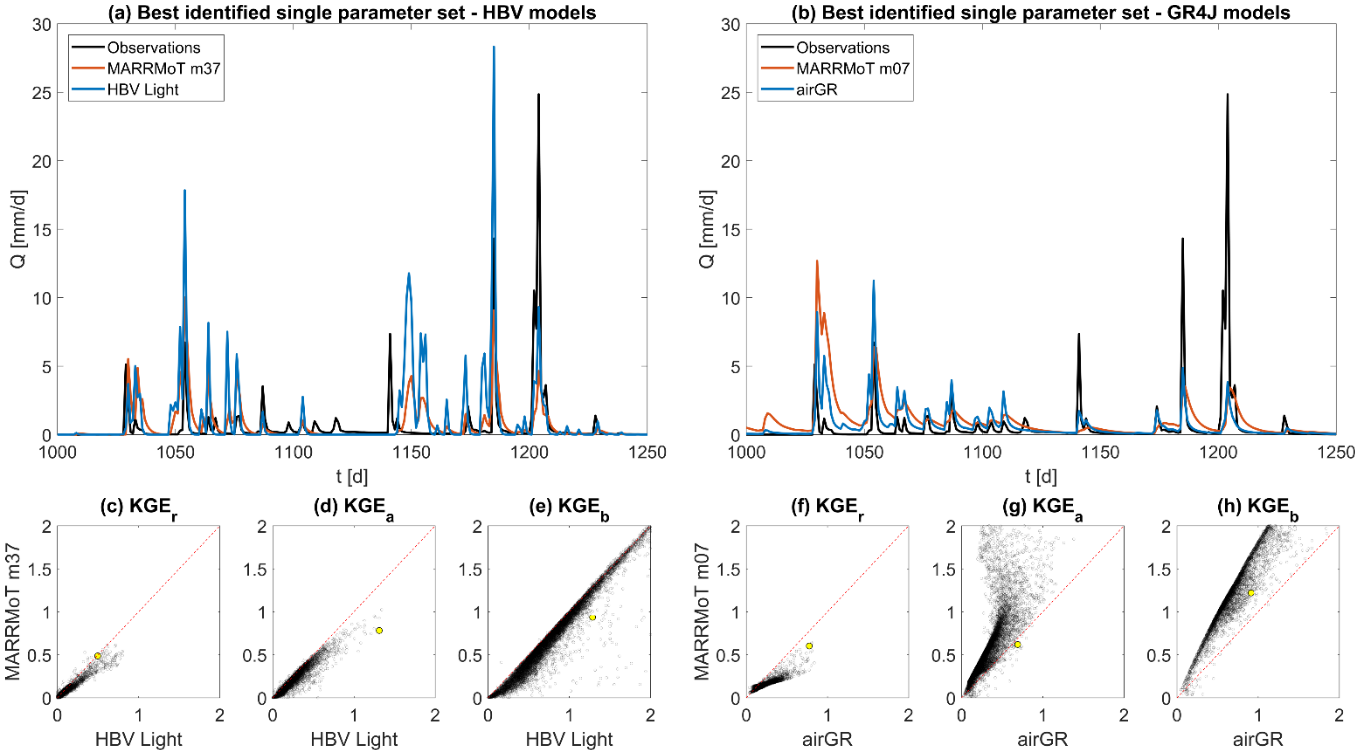


Figure 4: Comparison of two MARRMoT models and freely available model codes based on the same source material. (a) Close up of hydrographs generated by MARRMoT m37 and HBV Light using the same parameter values for their shared parameters. (b) Close up of hydrographs generated by MARRMoT m07 and airGR-GR4J using the same parameter values. (c-e) Constitutive components of the Kling-Gupta Efficiency (KGE) obtained by HBV Light and MARRMoT m37 for 10000 parameter sets in a single catchment. The yellow dot indicates the parameter set used to generate figure a. (f-h). Constitutive components of the KGE obtained by airGR-GR4J and MARRMoT m07 for 10000 parameter sets in a single catchment. The yellow dot indicates the parameter set used to generate figure b.

Table 1: MARRMoT models. Model IDs are used throughout this paper and the MARRMoT documentation. MARRMoT function names include a longer identifier that either refers to the name of the original model (e.g. m05_ihacres_7p_1s) or to the area of original application (e.g. m_01_collie1_1p_1s which was used in the Collie River basin). The column "Main changes" specifies structural changes between the MARRMoT model and the original model description (note that MARRMoT models are created solely based on the cited sources and not on any computer code). Not mentioned are cases where (i) model equations needed to be modified to account for the time step size at which the model is used; (ii) Ordinary Differential Equations were not given in the original source; (iii) cases where modelled processes were only described qualitatively in the original source, without equations; (iv) cases where model equations where smoothed in their MARRMoT implementations (these can be traced through the overview of flux equations in Supporting Materials S3).

ID	Original model name	Original time step	Main reference(s)	MARRMoT function	Main changes
01	Traditional bucket model	Annual	(Jothityangkoon et al., 2001)	m_01_collie1_1p_1s	-
02	Wetland, FLEX-Topo	Daily	(Savenije, 2010)	m_02_wetland_4p_1s	Model intended to be used with hillslope and plateau in spatially explicit fashion.
03	Unnamed	Monthly	(Jothityangkoon et al., 2001)	m_03_collie2_4p_1s	-
04	Unnamed	Daily	(Atkinson et al., 2002)	m_04_newzealand1_6p_1s	Separated constitutive functions from numerical approximation.
05	IHACRES	6 min to monthly	(Croke and Jakeman, 2004; Littlewood et al., 1997)	m_05_ihacres_7p_1s	Original can use temperature as proxy for evaporation; here PET is always used. Separated constitutive functions from numerical approximation.
06	Unnamed	Monthly	(Eder et al., 2003)	m_06_alpine1_4p_2s	Separated constitutive functions from numerical approximation.
07	GR4J	Daily	(Perrin et al., 2003; Santos et al., 2018)	m_07_gr4j_4p_2s	Combines equations from Santos et al. (2018) with Unit Hydrographs of Perrin et al. (2003).
08	Unnamed	Daily to annual	(Bai et al., 2009)	m_08_us1_5p_2s	Only 1 configuration from several different ones used here. This configuration shows a concept not seen in many other models. Separated constitutive functions from numerical approximation.
09	Unnamed	Daily to annual	(Son and Sivapalan, 2007)	m_09_susannah1_6p_2s	No spatial discretization through multiple buckets used here.
10	Unnamed	Daily to annual	(Son and Sivapalan, 2007)	m_10_susannah2_6p_2s	No spatial discretization through multiple buckets used here.
11	Unnamed	Daily	(Jothityangkoon et al., 2001)	m_11_collie3_6p_2s	-
12	Unnamed	Daily	(Eder et al., 2003)	m_12_alpine2_6p_2s	Separated constitutive functions from numerical approximation.
13	Hillslope, FLEX-Topo	Daily	(Savenije, 2010)	m_13_hillslope_7p_2s	Model intended to be used with wetland and plateau in spatially explicit fashion.
14	TOPMODEL		(Beven et al., 1995; Clark et al., 2008)	m_14_topmodel_7p_2s	No spatial discretization. Only 1 out of many possible configurations used. Not based on topographic index values.

15	Plateau, FLEX-Topo	Daily	(Savenije, 2010)	m_15_plateau_8p_2s	Model intended to be used with hillslope and wetland in spatially explicit fashion.
16	Unnamed	Hourly	(Atkinson et al., 2002, 2003)	m_16_newzealand2_8p_2s	Porosity and soil depth simplified to a single soil moisture storage parameter. Separated constitutive functions from numerical approximation.
17	Penman drying curve	Daily	(Penman, 1950; Wagener et al., 2002)	m_17_penman_4p_3s	-
18	SIMHYD	Daily	(Chiew et al., 2002)	m_18_simhyd_7p_3s	Interception and soil moisture excess flows expressed through different functions.
19	Unnamed	Daily	(Farmer et al., 2003)	m_19_australia_8p_3s	Porosity and soil depth simplified to a single soil moisture storage parameter. Evaporation equations simplified. Separated constitutive functions from numerical approximation.
20	GSFB	Daily, but meant for monthly yield	(Nathan and McMahon, 1990; Ye et al., 1997)	m_20_gsfb_8p_3s	-
21	FLEX-B	Hourly	(Fenicia et al., 2008b)	m_21_flexb_9p_3s	-
22	VIC	Daily	(Clark et al., 2008; Liang et al., 1994)	m_22_vic_10p_3s	No spatial discretization of land types. No use of sensible and latent heat fluxes. Leaf-Area-Index approximated with sinusoidal function and calibration parameters.
23	LASCAM	Daily	(Sivapalan et al., 1996)	m_23_lascam_24p_3s	-
24	Unnamed	Daily	(Ye et al., 2012)	m_24_mopex1_5p_4s	Different formulation for storage excess flows used here.
25	TCM	Daily and event (15 min)	(Moore and Bell, 2001)	m_25_tcm_6p_4s	No spatial discretization in different hydrologic zones.
26	FLEX-I	Hourly	(Fenicia et al., 2008b)	m_26_flexi_10p_4s	-
27	TANK model	Hourly to daily	(Sugawara, 1979, 1995)	m_27_tank_12p_4s	-
28	XINANJIANG	Daily	(Zhao, 1992)	m_28_xinanjiang_12p_4s	No spatial discretization. Tension water represented through double instead of single parabolic curve.
29	HyMOD	Daily	(Boyle, 2001; Wagener et al., 2001)	m_29_hymod_5p_5s	-
30	Unnamed	Daily	(Ye et al., 2012)	m_30_mopex2_7p_5s	Different formulation for storage excess flows used here.
31	Unnamed	Daily	(Ye et al., 2012)	m_31_mopex3_8p_5s	Different formulation for storage excess flows used here.
32	Unnamed	Daily	(Ye et al., 2012)	m_32_mopex4_10p_5s	Different formulation for storage excess flows used here. Leaf-Area-Index approximated with sinusoidal function with calibrated parameters.
33	SACRAMENTO	Daily	(Burnash, 1995; National Weather Service, 2005)	m_33_sacramento_11p_5s	Various equations in the lower zone were changed to allow simultaneous calculation of all fluxes instead of the original forced sequential calculation.

34	FLEX-IS	Daily	(Fenicia et al., 2008b; Nijzink et al., 2016)	m_34_flexis_12p_5s	Different formulation of storage excess flows. Separated constitutive functions from numerical approximation.
35	Unnamed	Daily	(Ye et al., 2012)	m_35_mopex5_12p_5s	Different formulation for storage excess flows used here. Leaf-Area-Index approximated with sinusoidal function with calibrated parameters.
36	MODHYDROLOG	Daily	(Chiew, 1990; Chiew and McMahon, 1994)	m_36_modhydrolog_15p_5s	No spatial routing scheme.
37	HBV-96	Daily	(Lindström et al., 1997)	m_37_hbv_15p_5s	No spatial discretization. No precipitation and evaporation from lakes. No correction factors for climate inputs.
38	TANK model - SMA	Hourly to daily	(Sugawara, 1979, 1995)	m_38_tank2_16p_5s	-
39	MCRM	Daily	(Moore and Bell, 2001)	m_39_mcrm_16p_5s	Simplified evaporation and routing procedures.
40	SMAR	Hourly to daily	(O’Connell et al., 1970; Tan and O’Connor, 1996)	m_40_smar_8p_6s	Fixed number of upper stores instead of treating this as a calibration parameter.
41	NAM	Daily	(Nielsen and Hansen, 1973)	m_41_nam_10p_6s	Linear reservoirs used instead of routing functions.
42	HYCYMODEL	Hourly to daily	(Fukushima, 1988)	m_42_hycymodel_12p_6s	Assumption made about evaporation equation. Separated model equations from numerical approximation.
43	GSM-SOCONT	Daily	(Schaefli et al., 2005)	m_43_gsmsocont_12p_6s	No spatial discretization. No annual glacier calculations.
44	ECHO	Hourly to daily	(Schaefli et al., 2014)	m_44_echo_16p_6s	No spatial discretization. Soil moisture storage given in absolute terms instead of fractional terms.
45	PRMS	1 min to daily	(Leavesley et al., 1983; Markstrom et al., 2015)	m_45_prms_18p_7s	PET is a model input instead of calculated within the model. Simplified interception and snow modules. No spatial discretization.
46	CLASSIC	Daily	(Crooks and Naden, 2007)	m_46_classic_12p_8s	No spatial discretization. No arable soil component. Separated model equations from numerical approximation.