ENHANCING HYDROLOGICAL RAINFALL-RUNOFF SIMULATION USING MACHINE LEARNING METHODS

EDUARDO JOSÉ ACUÑA ESPINOZA

ENHANCING HYDROLOGICAL RAINFALL-RUNOFF SIMULATION USING MACHINE LEARNING METHODS

Zur Erlangung des akademischen Grades eines

DOKTORS DER INGENIEURWISSENSCHAFTEN
(Dr.-Ing)

von der KIT-Fakultät für
Bauingenieur-, Geo- und Umweltwissenschaften
des Karlsruher Instituts für Technologie (KIT)
genehmigte

DISSERTATION

von
M. Sc. Eduardo José Acuña Espinoza
aus Costa Rica

Tag der mündlichen Prüfung:
06. Juni 2025

REFERENT: PD Dr.-Ing. Uwe Ehret
KORREFERENT: Prof. Dr. Nicole Bäuerle
KORREFERENT: Prof. Dr. Jan Cermak

Karlsruhe 2025

Dedicated to my mom and dad, for all the opportunities they gave me,

and to Maria Fernanda, whose support made this possible.

"Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful." – George Box

ACKNOWLEDGMENTS

Special thanks to my main supervisor, Uwe Ehret, for his technical guidance and continuous support over the last 3-years.

I would like to thank Manuel Álvarez, Ralf Loritz and Philipp Reiser for the many hours spent discussing this research. Furthermore, to Frederik Kratzert, Daniel Klotz and Martin Gauch, for their technical support and continuous feedback.

Many thanks to other scientists including Prof. Nicole Bäuerle, Prof. Hoshin Gupta and Prof. Andras Bardossy for taking the time to exchange ideas. Moreover, I would like to thank Anneli Guthke for including me in her group and allowing me to work in their office.

I would like to express my gratitude to the Karlsruhe Institute of Technology and its Open Access Publishing Fund for supporting the publication of my research articles. Additionally, I sincerely thank the Google Cloud Program (GCP) team for awarding me credits to facilitate this research, as well as the state of Baden-Württemberg for providing access to bwHPC.

ABSTRACT

Hydrological rainfall-runoff modeling plays an important role in several water resource applications, including flood forecasting, hydroelectric power generation, and water supply planning. Moreover, accurate and reliable predictions contribute to better decision-making in these areas. While machine learning (ML) techniques have significantly advanced these models, challenges remain in improving interpretability, generalization to unseen conditions, and efficient handling of high-resolution data. This thesis advances the state-of-the-art by addressing these challenges.

Chapters 2 and 3 focus on the hybrid modeling paradigm, where data-driven techniques—more specifically, long-short term memory networks (LSTMs)—are integrated with conceptual hydrological models. Through a series of experiments, the hybrid models' performance, interpretability, and generalization capabilities are assessed. The results demonstrate that hybrid models achieve state-of-the-art performance, comparable to stand-alone data-driven techniques, and surpassing traditional conceptual models. However, the experiments in Chapter 2 also reveal that, when coupled, the data-driven approach can compensate for structural deficiencies in the conceptual components. This suggests that relying solely on performance metrics for model selection in hybrid frameworks may be misleading. While hybrid models offer access to unobserved variables compared to stand-alone data-driven techniques, and provide some degree of interpretability, it is important to note that the interpretability derived from simplified basin-average conceptual models is more associative than grounded in strict physical principles. Therefore, this interpretability should be taken as such.

Chapter 3 explores the generalization capabilities of hybrid models, particularly their ability to predict extreme discharges under out-of-sample conditions. Findings show that hybrid models generally perform similarly to stand-alone LSTM networks. However, stand-alone LSTMs excel in areas where the conceptual component of the hybrid model struggles with runoff generation assumptions. At the same time, hybrid models produce higher discharges for the most extreme cases of the dataset, where LSTMs are constrained by their theoretical saturation limit, defined during the training process.

While chapters 2 and 3 focus on interpretability and generalization, chapter 4 addresses the challenge of applying data-driven methods to sub-daily predictions, where computational costs remain a challenge. This issue is especially relevant, as most hydrological studies using LSTMs focus on daily-scale predictions, whereas applications like flood forecasting would benefit from higher temporal resolution to more accurately capture the dynamics of hydrographs. To overcome this limitation, chapter 4 introduces a technique that processes data at multiple temporal frequencies within a single LSTM cell. This approach enhances model generality and simplicity while maintaining state-of-the-art performance.

Overall, this thesis contributes to enhance hydrological rainfall-runoff simulation through machine learning methods by (1) evaluating the integration of machine learn-

ing into conceptual hydrological modeling and (2) advancing purely data-driven approaches.

ZUSAMMENFASSUNG

Die hydrologische Niederschlags-Abfluss-Modellierung spielt eine wichtige Rolle in mehreren wasserwirtschaftlichen Anwendungen, einschließlich Hochwasservorhersage, Wasserkraftproduktion und Wasserressourcenplanung. Zudem tragen genaue und zuverlässige Vorhersagen zu einer besseren Entscheidungsfindung in diesen Bereichen bei. Während maschinelle Lernverfahren (ML) diese Modelle erheblich verbessert haben, bestehen weiterhin Herausforderungen bei der Verbesserung der Interpretierbarkeit, der Generalisierung auf unbekannte Bedingungen und der effizienten Verarbeitung hochauflösender Daten. Diese Arbeit trägt zur Weiterentwicklung des aktuellen Stands der Technik bei, indem sie diese Herausforderungen adressiert.

Kapitel 2 und 3 konzentrieren sich auf das hybride Modellierungsparadigma, bei dem datengestützte Techniken – insbesondere Long-Short-Term-Memory-Netzwerke (LSTMs) – mit konzeptionellen hydrologischen Modellen integriert werden. Durch eine Reihe von Experimenten werden die Leistung, Interpretierbarkeit und Generalisierungsfähigkeiten der Hybridmodelle bewertet. Die Ergebnisse zeigen, dass Hybridmodelle eine Leistung auf dem neuesten Stand der Technik erreichen, die mit rein datengestützten Techniken vergleichbar ist und traditionelle konzeptionelle Modelle übertrifft. Die Experimente in Kapitel 2 zeigen jedoch auch, dass der datengetriebene Ansatz strukturelle Defizite in den prozessbasierten Komponenten ausgleichen kann, wenn beide kombiniert werden. Dies deutet darauf hin, dass die ausschließliche Verwendung von Leistungsmetriken zur Modellauswahl in hybriden Frameworks irreführend sein kann. Während Hybridmodelle im Vergleich zu rein datengestützten Techniken Zugang zu untrainierte Variablen bieten und ein gewisses Maß an Interpretierbarkeit ermöglichen, sollte beachtet werden, dass die Interpretierbarkeit, ddie sich aus vereinfachten, Einzugsgebietsgemittelten Prozessmodellen Modellen ergibt, eher assoziativ als auf strengen physikalischen Prinzipien beruhend ist. Daher sollte diese Interpretierbarkeit auch entsprechend betrachtet werden.

Kapitel 3 untersucht die Generalisierungsfähigkeiten von Hybridmodellen, insbesondere ihre Fähigkeit, extreme Abflüsse unter Bedingungen außerhalb der Trainingsdaten vorherzusagen. Die Ergebnisse zeigen, dass Hybridmodelle im Allgemeinen ähnlich wie rein LSTM-Netzwerke abschneiden. Allerdings schneiden reine LSTMs in Bereichen besser ab, in denen die konzeptionelle Komponente des Hybridmodells Schwierigkeiten bei den Annahmen zur Abflussgenerierung hat. Gleichzeitig erzeugen Hybridmodelle höhere Abflüsse in den extremsten Fällen des Datensatzes, in denen LSTMs durch ihre theoretische Sättigungsgrenze, die während des Trainingsprozesses definiert wurde, eingeschränkt sind.

Während Kapitel 2 und 3 den Fokus auf Interpretierbarkeit und Generalisierung legen, befasst sich Kapitel 4 mit der Herausforderung, datengestützte Methoden auf Sub-Tages-Vorhersagen anzuwenden, bei denen die Rechenkosten eine Herausforderung darstellen. Dieses Problem ist besonders relevant, da die meisten hydrologischen Studien, die LSTMs verwenden, sich auf tägliche Vorhersagen konzentrieren, während Anwendungen wie die Hochwasserprognose von einer höheren zeitlichen Auflösung profitieren würden, um die Dynamik von Hydrographen genauer zu erfassen. Um

diese Einschränkung zu überwinden, führt Kapitel 4 eine Technik ein, die Daten mit mehreren zeitlichen Frequenzen innerhalb einer einzelnen LSTM-Zelle verarbeitet. Dieser Ansatz verbessert die Generalität und Einfachheit des Modells, während die ßtate-of-the-art"beibehalten wird., wodurch hochauflösende Vorhersagen für praktische Anwendungen realistischer werden.

Insgesamt trägt diese Dissertation dazu bei, die hydrologische Niederschlags-Abfluss-Simulation durch maschinelle Lernmethoden zu verbessern, indem sie (1) die Integration von maschinellem Lernen in konzeptionelle hydrologische Modelle bewertet und (2) rein datengestützte Ansätze weiterentwickelt.

CONTENTS

iv An approach for handling multiple temporal frequencies with different input dimensions using a single LSTM cell

v Conclusions

vi Appendix

LIST OF FIGURES

LIST OF TABLES

Part I

INTRODUCTION

1 INTRODUCTION

1.1 MOTIVATION AND OVERVIEW

The following work explores and develops techniques for improving the answer to the question: How much water will be in the river tomorrow? This answer has significant societal implications, as it directly impacts flood prediction, hydroelectric power generation, water supply for human consumption and agriculture, navigation, tourism, among other fields.

To answer this question, we build models. We express our system understanding through a series of mathematical equations, that give us an estimate of our variable of interest. Specifically, we express our model as $y = f(x)$, where y is the target variable, f is the model, and x are the model inputs. Given that the main driver to explain discharge in a river is precipitation, the area of hydrology that deals with this question is called rainfall-runoff modeling.

Rainfall-runoff modeling traces back to 1851, when James Mulvaney introduced what is now known as the rational method (Beven, 2012; Mulvaney, 1850). Since then, modeling approaches have continuously evolved, integrating deeper process understanding into their mathematical formulations. This evolution has led to the development of physically-based models that, with advances in computational power, numerical solvers, and data availability, can solve the shallow-water equations at the catchment scale (Bladé et al., 2014; Caviedes-Voullième et al., 2023; Li et al., 2025). However, these models are computationally demanding, particularly for large-scale simulations across multiple basins at the national level. As a result, simplified methods are still used for the day-to-day operations of forecasting agencies (LARSIM-Entwicklergemeinschaft, 2022). Given these constraints, this study focuses on conceptual and data-driven modeling approaches, as they fit better for operational applications at scale.

1.1.1 *Conceptual hydrological models*

Conceptual hydrological models encode our understanding of the different processes in a simplified structure, where storage units (commonly referred to as buckets) are interconnected by a network of fluxes. More formally, the models describe how different state variables interact through a set of equations that define how they evolve over time. Moreover, the equations rely on a set of parameters to control the model's behavior. Figure 1.1a shows an example of a conceptual model´s structure where discharge is predicted using 3 meteorological variables (precipitation, temperature and evapotranspiration), 5 state variables (buckets) and 8 parameters. As one can see, each bucket is associated with a different process, and the parameters define how the water is routed through the system. The association of the buckets and fluxes with physical processes, domains and states (e.g., snow, soil-moisture, percolation), provides a sense of interpretability. Examples of commonly used conceptual models are described in Chapter 2.

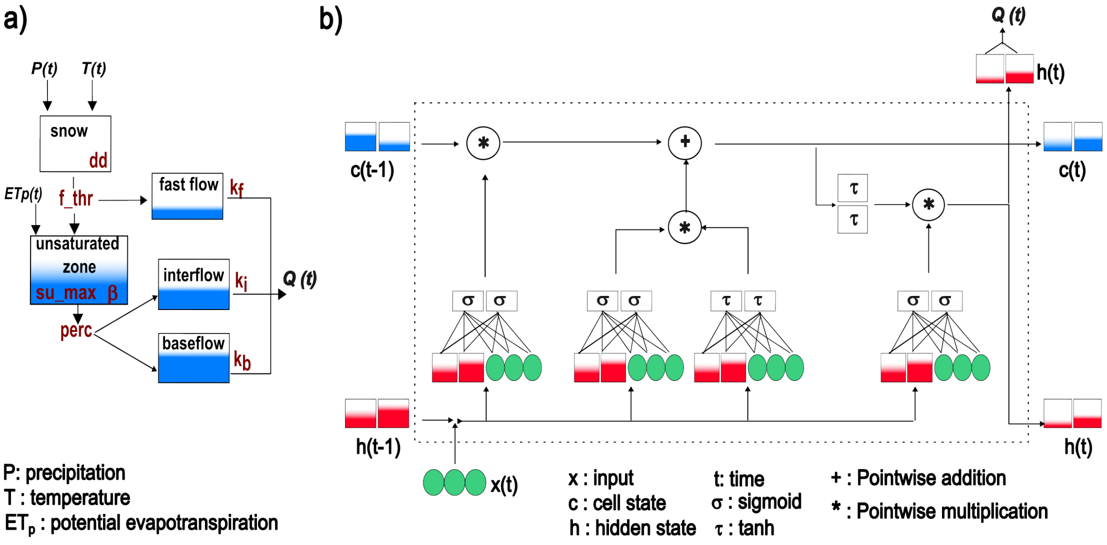

Figure 1.1: a) Example of a conceptual hydrological model structure. b) Example of an LSTM cell´s structure, interpreting the cell and hidden states as storages.

Despite their simplified structures and reliance on empirical equations, conceptual models have proven effective and are widely used in practice. As indicated by the British statistician George Box *"remember that all models are wrong; the practical question is how wrong do they have to be to not be useful"*. Nevertheless, to gain a better understanding of the model's capacity and to compare it with other methodologies, it is useful to consider the underlying assumptions made when constructing such models, and the associated limitations.

Most conceptual models assume that discharge can be reproduced using a predefined number of storage units (buckets), define a priori how these units interact, specify how the input variables influence the output signal and enforce mass conservation. Additionally, threshold parameters allow the model to activate different components depending on prevailing conditions (e.g., snow accumulates when the temperature is below zero, and the fast-flow reservoir fills only when precipitation exceeds a certain threshold). In most cases, the parameters governing fluxes between buckets remain constant over time, and discharge is expressed as a function of the bucket´s storage.

On the one hand, these assumptions provide a prior understanding of the system's functioning, which helps generalize to unseen conditions and requires less calibration data. Moreover, they offer a certain degree of interpretability—though this interpretability is often limited to association—which becomes useful when communicating with stakeholders. Furthermore, the modeler can recover unobserved variables (e.g., soil moisture) that can be used for further purposes.

On the other hand, these assumptions also cause model limitations, associated with overconstraining the model. In many cases, the imposed prior may be biased or overly simplistic. For instance, while mass conservation serves as a useful regularization constraint in a numerical scheme's control volume, its direct application to a large area may not always hold due to uncertainties in measurements and interpolation of the observed quantities. In operational settings, observed values are often adjusted in a preprocessing step using empirical relationships to compensate for these discrepancies (LARSIM-Entwicklergemeinschaft, 2022). Additionally, predefined interactions between storage units may be incomplete or incorrect, forcing the parameter calibration process to compensate for structural deficiencies in the model. As a result, different combinations of model structures and parameter sets can produce similar output signals, leading to ambiguity in model selection and interpretation (Beven, 2012; Spieler et al., 2020).

1.1.2 *Long-short term memory networks: LSTM*

Long Short-Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997) are a type of data-driven model that has demonstrated the potential to outperform conceptual models in hydrological modeling. Their application to rainfall-runoff simulation was introduced by Kratzert et al. (2018), and subsequent studies have corroborated and expanded upon these findings, establishing LSTMs as state-of-the-art methodologies for rainfall-runoff simulation (Kratzert et al., 2019b; Lees et al., 2021; Loritz et al., 2024) and operational flood forecasting (Nearing et al., 2024). In this section, a brief overview of their structure is provided, highlighting both the similarities and key differences between LSTMs and conceptual models. For a detailed explanation of LSTM's internal mechanisms, see Kratzert et al. (2018) and Starmer (2022).

LSTMs belong to the family of recurrent neural networks (RNNs), and, like conceptual models, are designed to process sequential data one step at a time. Moreover, they also store information using internal states, which are updated from one time step to the next based on previous values and new inputs. Consequently, both approaches rely on state variables to capture temporal dependencies. This structural similarity makes LSTMs particularly well-suited for hydrological modeling (Kratzert et al., 2018). Nevertheless, despite these shared characteristics, key differences arise in how each model processes and retains information.

Unlike conceptual models, which typically use a single set of storage units (buckets), LSTMs maintain two distinct types of memory: cell states and hidden states (see Fig. 1.1b). This design also differentiates them from standard recurrent neural networks (RNNs), which rely solely on hidden states for memory. The introduction of cell states helps LSTMs address a well-known limitation of standard RNNs—their difficulty in learning long-term dependencies due to instabilities in gradient propagation during training (Bengio et al., 1994; Pascanu et al., 2013). This limitation is particularly relevant in rainfall-runoff simulation, where discharge at a given time may be influenced by delayed hydrological processes such as snowmelt or slow groundwater routing, where the respective inputs enter the system days or even months in advance. By incorporating cell states that interact with other components of the LSTM cell through simple linear operations (see Fig. 1.1b), LSTMs facilitate more stable gradient propagation. This mechanism enables them to capture long-term dependencies more effectively, ultimately improving performance when processing extended hydrological time series.

Additionally, unlike conceptual models, LSTMs do not enforce mass conservation within their storage units. Moreover, while conceptual models impose predefined connections between storage components, LSTMs dynamically learn, during optimization, both the interactions between states and the influence of the different inputs. This flexibility allows us to incorporate multiple input variables and, during optimization, determine their relevance and impact, whereas, in conceptual models, these relationships are previously defined, making it more challenging to integrate new information. Furthermore, rather than modifying storage through a fixed set of parameters, LSTMs employ dynamic gating mechanisms to regulate the storages based on context, allowing the model to adapt its behavior under varying conditions.

The inherent flexibility of LSTMs, in contrast to the rigid structures of conceptual models, has enabled them to achieve higher performance. However, the fact that LSTMs do not inherently associate a meaning with their parameters and states, has raised concerns about interpretability (Reichstein et al., 2019). Moreover, purely data-driven models struggle with generalization beyond the training range, as they rely on observed patterns, and extreme events—by definition—are rare and often underrepresented in training data. Lastly, while most hydrological studies using LSTMs focus on daily-scale predictions, applications such as flood forecasting would benefit from sub-daily (e.g., hourly) predictions, which come with a significantly higher computational cost for both training and evaluation (Gauch et al., 2021).

1.1.3 *Hybrid models*

Combining data-driven and conceptual models into so-called hybrid models has been proposed as a strategy for enhancing hydrological modeling (Reichstein et al., 2019; Shen et al., 2023). The key idea is to leverage the predictive power and flexibility of data-driven methods while incorporating the regularization of a conceptual structure, embedding prior knowledge into the learning process. This integration aims to create models that are both accurate and physically consistent.

Various strategies have been proposed to integrate these methodologies. Data-driven techniques have been used to replace entire sub-modules of conceptual models (Hoge et al., 2022; Li et al., 2023) and to post-process their outputs (Frame et al., 2021). Additionally, some approaches impose conservation constraints directly within the data-driven architecture (Frame et al., 2023). Another common strategy involves using data-driven methods to parameterize physical models, incorporating additional dynamics and context (Feng et al., 2022; Kraft et al., 2022). This last approach, which I will explore through various experiments, will be examined in detail, along with its benefits and limitations, in Chapters 2 and 3.

Herath et al. (2021) and Reichstein et al. (2019) have suggested the potential of hybrid models to improve interpretability and enhanced generalization. By introducing conceptual regularization, hybrid models could provide a structured representation of physical processes while preserving the predictive capabilities of data-driven methods. Regarding generalization, by incorporating physical constraints through conceptual components, hybrid models may improve their ability to generalize to unseen conditions, enhancing robustness in hydrological predictions under extreme or out-of-sample scenarios.

1.2 OBJECTIVES

This work explores techniques to overcome limitations in the current state-of-the-art. Building on the hybrid model paradigm, I first evaluate whether integrating data-driven techniques with conceptual hydrological models enhances model interpretability and generalization. Additionally, I investigate methods for handling high-resolution data while maintaining a moderate computational cost. The following objectives are proposed to guide this research.

General objective

- Enhance hydrological rainfall-runoff simulations using machine learning methods.

Specific objectives

- Investigate whether the flexibility of the data-driven component in hybrid models affects their physical interpretability.
- Analyze the ability of hybrid hydrological models to generalize beyond training conditions, particularly in predicting extreme hydrological events.
- Develop an LSTM-based architecture to efficiently handle high-resolution hydrological data while maintaining a moderate computational cost.

1.3 CONTRIBUTION AND DOCUMENT STRUCTURE

Having introduced the key modeling approaches, I now turn to the structure of this document. The subsequent chapters explore how modifications and combinations of conceptual and data-driven models can enhance the accuracy of our simulations.

Chapter 2 and Chapter 3 adopt the hybrid modeling approach to investigate the effects of combining data-driven and conceptual models. Through a series of experiments, I assess whether the potential advantages highlighted in the literature can be supported. Specifically, I focus on evaluating the interpretability and generalization capabilities of these models, with particular emphasis on their ability to simulate conditions beyond the training range.

Chapter 4 addresses the challenges faced by data-driven methods when handling long sequences of high-resolution data, particularly in scenarios such as hourly discharge predictions. A method to address these challenges is proposed, improving both generalization and computational efficiency.

Chapter 5 summarizes the key findings and presents the conclusions drawn from the studies.

Part II

TO BUCKET OR NOT TO BUCKET? ANALYZING THE PERFORMANCE AND INTERPRETABILITY OF HYBRID HYDROLOGICAL MODELS WITH DYNAMIC PARAMETERIZATION

This study is published in the scientific journal Hydrology and Earth System Science (HESS). The remainder of part II is a reprint of:

Acuña Espinoza, E., Loritz, R., Álvarez Chaves, M., Bäuerle, N., and Ehret, U.: To bucket or not to bucket? Analyzing the performance and interpretability of hybrid hydrological models with dynamic parameterization, Hydrol. Earth Syst. Sci., 28, 2705–2719, https://doi.org/10.5194/hess-28-2705-2024, *2024.*

2

TO BUCKET OR NOT TO BUCKET? ANALYZING THE PERFORMANCE AND INTERPRETABILITY OF HYBRID HYDROLOGICAL MODELS WITH DYNAMIC PARAMETERIZATION

ABSTRACT

Hydrological hybrid models have been proposed as an option to combine the enhanced performance of deep learning methods with the interpretability of process-based models. Among the various hybrid methods available, the dynamic parameterization of conceptual models using long short-term memory (LSTM) networks has shown high potential. We explored this method further to evaluate specifically if the flexibility given by the dynamic parameterization overwrites the physical interpretability of the process-based part. We conducted our study using a subset of the CAMELS-GB dataset. First, we show that the hybrid model can reach state-of-the-art performance, comparable with LSTM, and surpassing the performance of conceptual models in the same area. We then modified the conceptual model structure to assess if the dynamic parameterization can compensate for structural deficiencies of the model. Our results demonstrated that the deep learning method can effectively compensate for these deficiencies. A model selection technique based purely on the performance to predict streamflow, for this type of hybrid model, is hence not advisable. In a second experiment, we demonstrated that if a well-tested model architecture is combined with an LSTM, the deep learning model can learn to operate the process-based model in a consistent manner, and untrained variables can be recovered. In conclusion, for our case study, we show that hybrid models cannot surpass the performance of data-driven methods, and the remaining advantage of such models is the access to untrained variables.

2.1 INTRODUCTION

Rainfall–runoff models are useful tools to support decision-making processes related to water resources management and flood protection. Over the past decades, hydrological conceptual models have emerged as important tools for these purposes, finding widespread usage in academia, industry, and national weather services (Boughton and Droop, 2003). These models, known for their simplicity, computational efficiency, and ability to generalize, encode our understanding of hydrological processes within a fixed model structure. By connecting the various macroscopic storages (also known as buckets) through a network of fluxes, conceptual models try to emulate the internal processes occurring within a catchment. The accurate representation of these processes relies on calibrated parameters, which are adjusted to achieve consistency with observed data. Examples of widely used conceptual models include Hydrologiska Byråns Vattenavdelning (HBV) (Bergström, 1992), Sacramento (Burnash et al., 1973), GR4J (Perrin et al., 2003)), Precipitation-Runoff Modeling System (PRMS) (Leavesley et al., 1983), and TOPMODEL (Beven and Kirkby, 1979), to name a few. Additionally, there are software tools available, such as Raven (Craig et al., 2020) and Superflex (Dal Molin et al., 2021) which facilitate the creation of customized models tailored to specific basin characteristics and key hydrological processes.

Despite the widespread use of conceptual models, data-driven techniques, particularly long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) networks, have recently shown the potential to outperform conceptual models, particularly in large sample model comparison studies (Feng et al., 2020; Kratzert et al., 2019b; Lees et al., 2021). The improvement in performance can be attributed, partly, to the inherent flexibility of LSTM networks (LSTMs hereafter), which surpasses the constraints imposed by fixed model structures by effectively mapping connections and patterns through optimization techniques. However, the characteristic that allows LSTMs to excel in performance has also sparked criticism regarding their interpretability (Reichstein et al., 2019), owing to the fact that weights and biases in LSTMs lack clear semantic meaning, making it challenging to discern the underlying reasons for their decision and predictions. In recent years, notable advancements in linking hydrological concepts to the internal states of LSTMs have been made (Kratzert et al., 2019a; Lees et al., 2022), and we seek to further contribute in this research direction.

Reichstein et al. (2019) and Shen et al. (2023) indicate that combining process-based environmental models with machine learning (ML) approaches, into so-called hybrid models, can harness the strengths of both methodologies, leveraging the improved performance of data-driven techniques while retaining the interpretability and consistency offered by physical models. Among the various approaches proposed by the authors, one method involves the parameterization of physical models using data-driven techniques. Kraft et al. (2022) applied this method, with the idea that replacing poorly understood or challenging-to-parameterize processes with ML models can effectively reduce model biases and enhance local adaptivity. Moreover, their study demonstrated that the hybrid approach achieved comparable performance to process-based models. Feng et al. (2022) followed a similar procedure, in which the parameters of an HBV model were dynamically estimated using an LSTM network. Their study convincingly demonstrates the effectiveness of this approach, revealing its ability to achieve state-of-the-art performance that directly rivals purely data-driven methods when applied to the Catchment Attributes and Meteorology for Large-sample Studies (CAMELS) dataset (Addor et al., 2017) in the United States (CAMELS-US). In their study, Feng et al. (2022) implemented both static and dynamic parameterization techniques and observed that the latter led to slightly improved performance.

The dynamic parameterization of a process-based model is not a new idea. Lan et al. (2020) indicate that, historically, the most common approach to accomplish this is the calibration for different sub-periods. They support this statement by referencing over 20 studies on this subject published in the last 15 years. According to those authors, this method divides the data into sub-periods, considering seasonal characteristics or clustering approaches and proposing a set of parameters for each sub-period. The idea is to capture the temporal variations of the catchment characteristics.

Using an LSTM to give a dynamic parameterization of a process-based model may be seen as a generalization of this process. Specifically, one uses a recurrent neural network that analyzes a given sequence length, so the proposed parameters are context informed and reflect the current state of the system. The main difference is that the data-driven parameterization is much more flexible, as a custom parameterization can be proposed for each prediction, and it is not constrained to a typical small set of predefined sub-periods. Also, one can include as input to the LSTM any information that is considered useful to make an informed parameter inference, even if this is not used later in the conceptual part of the model.

However, it is important to note that Feng et al. (2022) warn about the flexibility of LSTM networks when used for dynamic parameterizations. They posed the hypothesis that, while applying dynamic parameterization increases the likelihood of achieving high performance, there is a risk of compromising the physical significance of the model, potentially resulting in the system behaving more like an LSTM variant rather than a hydrologically meaningful model. In other words, model deficits and ill-defined process descriptions might be compensated by the LSTM. Moreover, Frame et al. (2022) argue that adding any type of constraint, physically based or otherwise, to a data-driven model is only beneficial when such constraints contribute to the optimization process.

Motivated by the outcomes achieved in the aforementioned articles, our study aims to dig deeper into the coupling of LSTM and conceptual models. We believe that dynamic parameters provided by an LSTM allow the conceptual model to not only adapt to changes in the hydrological regime, which is physically reasonable (Loritz et al., 2018), but also to compensate for inherent deficiencies or oversimplifications within the model structure. More specifically, and guided by the warning given by Feng et al. (2022) and Frame et al. (2022), our study aims to address the following research questions:

- 1. Do conceptual models serve as an effective regularization mechanism for the dynamic parameterization of LSTMs?
- 2. Does the data-driven dynamic parameterization compromise the physical interpretability of the conceptual model?

To address the research questions at hand, we have structured our article as follows. In Sect. 2.2 we describe the structure and training process of the conceptual, data-driven, and hybrid models employed in this study. Additionally, we outline the details of the dataset used to train and test the rainfall–runoff models. In Sect. 2.3, after proving that the hybrid model performance is comparable with the LSTM, we conduct experiments to answer our first research question. By systematically modifying the conceptual model, we assess how different forms of regularization affect the overall performance of the hybrid model. This will allow us to better understand the effect of different conceptual models as regularization and the interaction between the data-driven and conceptual components. Furthermore, to address the second research question, we analyze the internal states of the conceptual model to evaluate how much physical interpretability the different variants of our conceptual model are keeping. Finally, we summarize our key findings in Sect. 2.4.

2.2 DATA AND METHODS

To answer the research questions stated in the previous section, we compared three types of models: purely data-driven (LSTM), stand-alone process-based models, and the hybrid approach. The first two types served as baselines in the different experiments we performed for the latter.

The first subsections of this section present an overview of the dataset utilized for training and testing our models. The second subsection describes the dataset used to evaluate the internal states of our process-based models. The last three segments explain the structures of the different models.

2.2.1 *Dataset*

To train and test our rainfall–runoff models, we used the CAMELS-GB dataset (Coxon et al., 2020a). This dataset contains information about river discharge, catchment attributes, and meteorological time series for 671 catchments in Great Britain. To facilitate the comparison of our results with the studies of Lees et al. (2021, 2022), we maintained the periods for training (1 October 1980–31 December 1997), validation (1 October 1975–30 September 1980), and testing (1 January 1998–31 December 2008) from their studies.

2.2.2 *ERA5-LAND*

As outlined in the introduction, one of the primary objectives of this study is to assess the physical consistency of our hybrid model. To achieve this, we conducted several tests, one of which involved comparing the unsaturated zone reservoir of the conceptual model with soil moisture estimates (details in Sect. 2.3.5). Following the procedure proposed by Lees et al. (2022), we compared our model's results with data from ERA5-LAND (Sabater et al., 2021). This dataset, based on a 9 km x 9 km gridded format, is a land component reanalysis of the ERA5 dataset (Hersbach et al., 2020). According to Lees et al. (2022), reanalysis data offer several advantages, including longer time series availability, easy transferability to basin-average quantities (consistent with the CAMELS-GB process) due to the gridded format, and global coverage, enabling its application in various locations. As our study region and testing period aligns with that of Lees et al. (2022), we utilized the NetCDF file provided by the authors, which was publicly accessible. We then extracted the values and normalized the data to a [0–1] range for comparative purposes. This normalization approach is consistent with

Ehret et al. (2020), where they followed a similar process to assess the realism of the unsaturated zone soil moisture dynamics of a conceptual hydrological model.

ERA5-Land contains information about soil water volume at four different levels. Level 1 (swvl1) provides information at a depth of 0 to 7 cm, level 2 (swvl2) from 7 to 28 cm, level 3 (swvl3) from 28 to 100 cm, and level 4 (swvl4) from 100 to 289 cm. When using this information to evaluate our models, we consistently found higher correlations for all cases when compared against swvl3. Therefore, the results reported in Sect. 2.3.5 are associated with that depth.

2.2.3 *Conceptual hydrological model*

In this study, we employ a conceptual model named the Simple Hydrological Model (SHM) (Ehret et al., 2020) that is in its essence a slightly altered HBV model. A description of the model architecture and its internal working can be found in Appendix A.1. We used the SHM both as a stand-alone benchmark and as an integral component of the hybrid model.

To establish a benchmark for comparing our data-driven and hybrid methods, we performed individual calibrations of the SHM for each specific basin of interest. This approach is in line with Kratzert et al. (2024, 2019b) and Nearing et al. (2021), who indicate that conceptual models generally perform better when calibrated at the individual basin level rather than using a regional calibration approach. To ensure a fair comparison and mitigate potential calibration biases that may favor our hybrid model, we employed two established calibration methods and selected the one that yielded the best performance for each basin. We used "shuffled complex evolution" (SCE-UA) (Duan et al., 1994) and "differential evolution adaptive metropolis" (DREAM) (Vrugt, 2016), both implemented within the SPOTPY (Statistical Parameter Optimization Tool for Python) library (Houska et al., 2015).

2.2.4 *LSTM*

As mentioned in previous sections, we incorporated an LSTM model as a benchmark for our comparison. For a comprehensive understanding of the internal workings of LSTM networks, we refer to the work by Kratzert et al. (2018). In this subsection, we will provide an overview of the key aspects required to comprehend the training process. Our data-driven model was implemented using the PyTorch library (Paszke et al., 2019), and the corresponding code can be found in the repository accompanying this paper.

The model architecture and hyperparameters align with Lees et al. (2021) and Lees et al. (2022). We used a single LSTM layer with 64 hidden states, a dropout rate of 0.4, an initial learning rate of 1 x 10^{-3}, and a sequence length of 365 d. The batch size was set to 256, and the initial bias of the forget gate was set to 3. Additionally, the Adam algorithm (Kingma and Ba, 2014) was used for the optimization.

We also maintained as input three dynamic forcing variables (precipitation, potential evapotranspiration, and temperature), along with the same 22 static attributes proposed in the original studies, which encode key characteristics of the catchments. The model output was compared against the observed specific discharge. Following common ML practices, both the input and output data were standardized using the global mean and standard deviation of the training dataset.

To train the LSTM, we used the basin-averaged Nash–Sutcliffe efficiency (NSE^*) loss function proposed by Kratzert et al. (2019b). This function divides the squared error between the modeled and observed output by the variance of the specific discharge series associated with each respective basin in the training period. As described by the authors, (NSE^*) provides an objective function that reduces bias towards large humid basins during the optimization process, avoiding the underperformance of the regional model in catchments with lower discharges. Given that we are training our regional model for a batch size $N = 256$, the training loss was calculated according to Eq. (2.1):

$$NSE^* = \frac{1}{N} \cdot \sum_{i=1}^{N} \frac{(y_i^{obs} - y_i^{sim})^2}{(s_i + \epsilon)^2}, \tag{2.1}$$

where y_i^{obs} is the observed discharged (standardized), y_i^{sim} the simulated discharged (standardized), s_i the standard deviation of the flow series (in training period) for the basin associated with element i, and ϵ is a numerical stabilizer ($\epsilon = 0.1$) so that the loss function remains stable even when basins with low-flow standard deviations are considered.

2.2.5 *Hybrid model: LSTM+SHM*

Our hybrid model was created by combining an LSTM network, with the same architecture as the one from the previous section, with the SHM. The LSTM network predicts a set of values that serve as parameters for the SHM for each simulation time step. These parameter values are then utilized by the SHM to simulate the discharge, as indicated in Fig. 2.1.

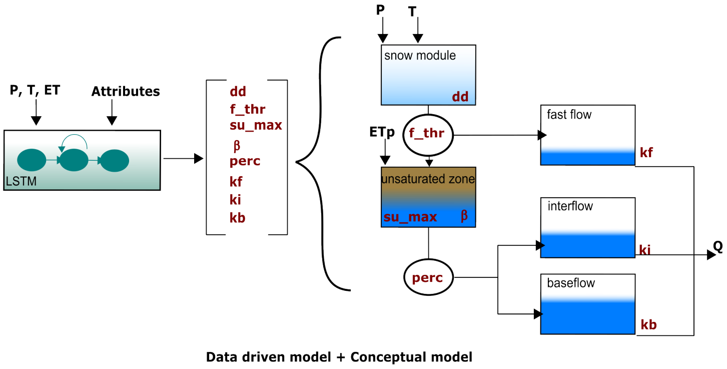

Figure 2.1: Structure of the hybrid hydrological model: LSTM+SHM

An alternative way to interpret the hybrid model is to see the conceptual model as a head layer on the LSTM. As we see in Table 2.1, in our stand-alone LSTM we require a dense layer (e.g., fully connected linear layer) to translate the information contained in the hidden states into a single output signal. In the LSTM+SHM case, a dense layer is still used to convert the hidden states into as many output signals as parameters in the conceptual model. However, these signals are further processed using the conceptual model to obtain the final discharge. One of the hypotheses that will be tested in the following sections is if further processing of the signals through the conceptual structure allows us to recover information about non-target variables, e.g., soil moisture.

Table 2.1: Visualization of hybrid models as head layers of data-driven methods

Model	**Neural Network**	**Head Layer**	**Output**
LSTM	LSTM	Dense	Q
LSTM+SHM	LSTM	Dense+SHM	Q
LSTM+Bucket	LSTM	Dense+Bucket	Q
LSTM+NonSense	LSTM	Dense+NonSense	Q

Appendix A.2 provides a comprehensive explanation of the training process for the hybrid models, emphasizing the distinctions from the training approach utilized for the LSTM models.

2.3 RESULTS AND DISCUSSION

2.3.1 *Benchmarking our LSTM model*

Kratzert et al. (2024) explains the importance of using community benchmarks to test if new ML pipelines are configured appropriately. They suggest that in the case that the researchers decide to use their own models or different setups, they should first recreate standard benchmarks to make sure that their model is up to date with the current state of the art and then make the respective changes.

Considering that the stand-alone LSTM model was going to be used as a baseline for all our experiments, we trained our LSTM model architecture on the benchmark established by Lees et al. (2021) for CAMELS-GB and compared its performance against their model. To further validate our architecture, we also trained on the benchmark established by Kratzert et al. (2019b) for CAMELS-US and again compared the performance of our model against theirs.

As we can see in Fig. 2.2, the cumulative distribution functions (CDFs) for the Nash–Sutcliffe efficiency (NSE) of the different basins are very much alike. For the case of Great Britain, Lees' model achieved a median NSE of 0.88, while ours reached 0.87. In the case of the USA, Kratzert's benchmark reported a median NSE of 0.759, while our model got 0.74. The small differences can be explained by the fact that both benchmark studies make the calculation based on an ensemble of various LSTM models, while we only presented the results for a single run. However, the overall agreement validates our model's pipeline and increases the confidence in the results.

2.3.2 *LSTM vs LSTM+SHM*

Once we tested our stand-alone LSTM pipeline, the next task was to develop our hybrid model (LSTM+SHM). By following the data and methods outlined in Sect. 2.2, we achieved a performance comparable to that of an LSTM. When evaluated across the 669 basins in the testing period, the LSTM reported a median NSE of 0.87, while the LSTM+SHM yielded a value of 0.84. Figure 2.3a displays a CDF–NSE curve, clearly

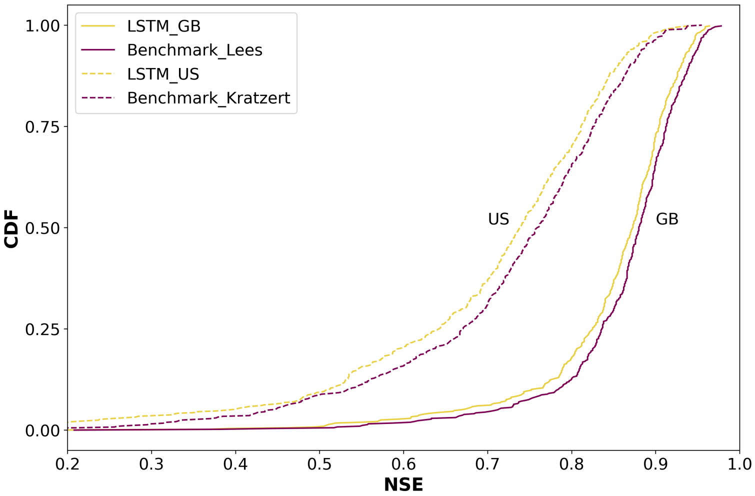

Figure 2.2: Cumulative density functions of the NSE comparing our LSTM model with current state-of-the-art benchmarks

demonstrating the close performance between both models. In the same figure, we can see that both models outperformed the basin-wide calibration of the SHM model, which achieved a median NSE of 0.76.

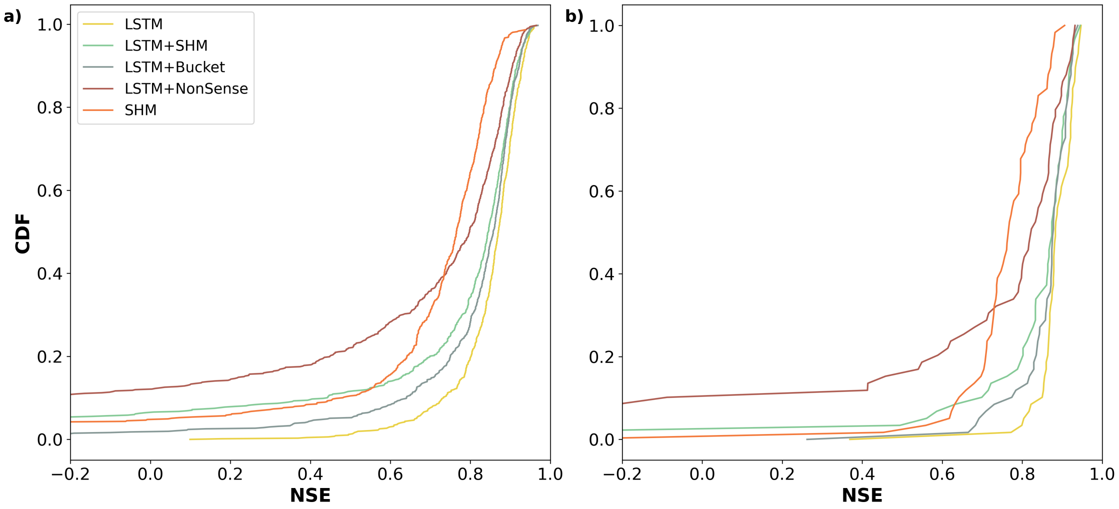

Figure 2.3: Cumulative density functions of the NSE for the different models. a) CDF was calculated using all 669 basins b) CDF was calculated using a subset of 60 basins

One point worth explaining is the decreased performance of our hybrid model compared to the LSTM in the low performing basins. More specifically, the LSTM reported only 3 basins with NSE lower than zero, while the hybrid reported 44. Of these 44 basins, 37 were also the lowest performing basins in the stand-alone SHM model, which suggests a problem with the input data. The LSTM network can account for biases in the forcing variables (e.g., precipitation or evapotranspiration) because mass conservation is not enforced (Frame et al., 2023). However, both the conceptual and the hybrid approaches have a mass conservative structure, so the input quantities cannot be adjusted. This problem was also reported by Feng et al. (2022) when applied to certain basins in CAMELS-US. It is important to highlight that this issue was observed in under 7% (44/669) of the data, and in most cases, the performance of the hybrid approach is fully comparable with the LSTM.

In summary, we observed comparable performance between the LSTM and LSTM+SHM models. Moreover, both models outperformed the SHM-only model, which indicates that the dynamic parameterization given by the LSTM is able to improve the predictive capability of the model. This finding aligns with Feng et al. (2022), where they reached a similar conclusion despite using a different conceptual model and applying it to a different dataset. However, as described in the Introduction, we are interested in looking at the LSTM– SHM interaction to evaluate if the good performance of the hybrid model is due to the right reasons (Kirchner, 2006) and based on a consistent interaction between the two model approaches, or if the LSTM network is overwriting the conceptual element. This will be explored in the following section.

2.3.3 *Effect of different regularizations*

The first step to answer the aforementioned research question was to evaluate if the dynamic parameterization given by the LSTM can overcome the regularization imposed by the conceptual model. For this, we conducted two experiments, in which the structure of the conceptual model was modified. In the first experiment (see Fig. 2.4a), we substituted the SHM with a single linear reservoir, leading to the removal of most hydrological processes typically represented in a conceptual model through different reservoirs and interconnecting fluxes. A single bucket model only assures mass conservation and a dissipative effect in which the input is lagged based on the recession coefficient in combination with a macroscopic storage. As observed in Fig. 2.4a, the model involves two calibration parameters: the recession parameter k and a factor for the evapotranspiration term (α). Similar to the previous cases, we defined predefined ranges in which the parameters were allowed to vary, with k in the range [1–500] and α in the range [0–1.5]. Our initial expectation was that if our head layer (a) restricts the flexibility of the LSTM because the output of the LSTM (after our dense layer) is further passed through a one-process (single bucket) layer and (b) the one-process layer encodes almost no hydrological process understanding, then the performance of the model would drop. However this was not the case. The performance of this hybrid model (LSTM+Bucket) is fully comparable to that of the LSTM and LSTM+SHM, achieving a median NSE of 0.86 (see Fig. 2.3a). For reference, when calibrated without dynamic parameterization, the median NSE of the stand-alone bucket model drops to 0.59. This finding indicates that the LSTM's dynamic parameterization effectively compensates for the missing processes, and the regularization provided by the single bucket is insufficient to impact the model's performance.

Given the insights gained from the LSTM+Bucket experiment, we conducted a second experiment introducing an intentionally implausible structure in the conceptual model, referred to as LSTM+NonSense. As shown in Fig. 2.4b, we removed the fast-flow reservoir, creating a single flow path comprising the baseflow, interflow, and unsaturated zone, in that specific order. We also maintained the parameter ranges specified in Table A.1, which restrict the baseflow reservoir to have smaller recession times than the interflow. Then, only after the water has been routed through these two reservoirs can

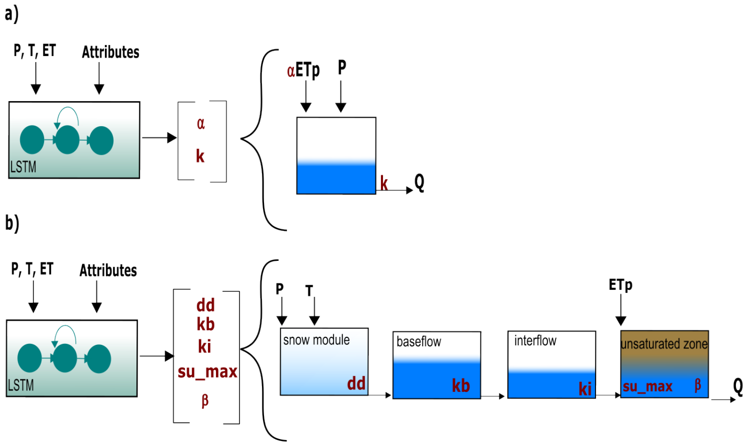

Figure 2.4: Structure of the different regularization: a) LSTM+Bucket b) LSTM+NonSense

it enter the unsaturated zone, where the outflow is no longer controlled by a recession parameter but by an exponential relationship depending on *su_max* and β. The stand-alone NonSense model yielded a median NSE of 0.51. However, after applying dynamic parameterization, the LSTM+NonSense achieved a median NSE of 0.80 (see Fig. 2.3a), improving the stand-alone NonSense by over 50% and surpassing the SHM model in 60% of the basins. During the test, we observed that the optimization routines tried to reduce the recession parameter of the baseflow and interflow to avoid the initial lagging. This caused the optimized parameters to reach the lower limits, which might have limited an additional performance increase. Expanding the parameter ranges might lead to a further performance gain; however, this would come at the cost of reducing the differences between the reservoirs, which contradicts the objective of the experiment. Taken together, these experiments provided valuable insights into addressing the first research question posed in the Introduction: can conceptual models effectively serve as a regularization mechanism for the dynamic parameterization given by the LSTMs? Based on our results, we observed that the regularization offered by the conceptual model is not strong enough to reduce the hybrid model performance, and the dynamic parameterization given by the LSTM can even compensate for missing processes and implausible structures. Figure 2.5 highlights that, for some basins, we obtained a similar hydrograph for all models used. Therefore, we recommend being careful about using this hybrid scheme for comparing different types of conceptual models or multiple working hypotheses (Clark et al., 2011), especially if we are evaluating model adequacy by performance alone, as the overall performance can be adjusted by the data-driven part.

2.3.4 *Testing on a subset of basins*

In the above sections, we showed that both LSTMs and hybrid models outperformed stand-alone conceptual models. In this subsequent section, we replicate the experiments focusing on a subset of basins. This subset responded to an inherent limitation of conceptual models, which in principle does not affect data-driven techniques. Unlike LSTM networks, which learn directly from the data without a predefined structure, con-

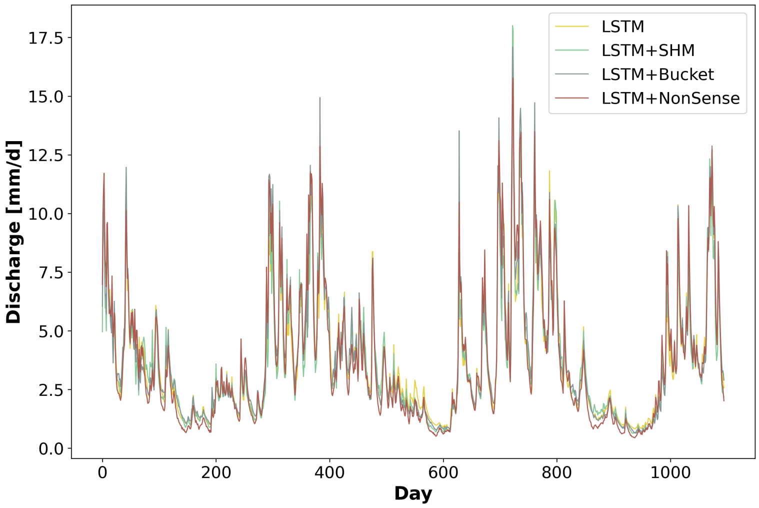

Figure 2.5: Specific discharge series in the testing period for basin ID 15006, simulated by the different models

ceptual models have fixed model architectures designed to represent specific processes. This means that anthropogenic impacts such as reservoir operations, withdrawals, or transfers may not be adequately captured by conceptual models unless they are directly accounted for. While this limitation is a clear advantage of data-driven techniques, we wanted to make a comparison on a level playing field. Therefore, as a first filter, we selected only basins with the label "benchmark_catch=TRUE", which, according to Coxon et al. (2020a), can be treated as "near-natural", i.e., catchments in which the human influence in flow regimes is modest and where natural processes predominantly drive the flow regimes. As a second filter, we considered the temporal resolution of the data and the size of the catchment. The CAMELS-GB dataset contains data with a daily resolution. Consequently, we need to consider catchments with a sufficient size such that discharge variations are resolved by daily data. After applying the aforementioned filters, we identified 60 basins that passed both criteria. For detailed information on the specific basin IDs, please refer to the Supplement of this article.

Figure 2.3b shows the results when the models are tested on the subset of 60 basins. We can see that the LSTM (median NSE = 0.88), LSTM+SHM (0.87), LSTM+Bucket (0.88), and LSTM+NonSense (0.82) continue to outperform the stand-alone SHM (0.76) in a setting designed to account for the limitation of the latter. This result reaffirms the findings highlighted in the preceding sections.

2.3.5 *Analysis of LSTM+SHM*

To tackle our second research question and assess the interpretability of the conceptual part of our LSTM+SHM model, we conducted several tests. Hybrid models, as highlighted by Feng et al. (2022), Kraft et al. (2022) and Hoge et al. (2022), offer the advantage of providing access to untrained variables as the model's states and fluxes have dimensions and semantic meaning. As such, our first test was a model intercomparison. Specifically, we evaluated the filling level of the unsaturated zone reservoir, representing soil moisture in our model (LSTM+SHM), against ERA5-LAND soil water volume information. The process of utilizing reanalysis data and the necessary data processing steps for this comparison are detailed in Sect. 2.2.2.

Across the 669 basins in the testing period, the LSTM+SHM model demonstrated a median correlation of 0.86 when compared against the soil moisture simulation provided by ERA5-LAND. This result indicates that the unsaturated zone dynamics are well represented in our model and that the hybrid approach allows us to recover this variable without including any soil moisture information in our training.

Lees et al. (2022) present an alternative method to extract non-target variables from data-driven techniques. They train a model (which they call a probe) to map the information contained in the cell states of the LSTM to a given variable. Specifically, they trained an LSTM using CAMELS-GB, mapped the soil moisture information using a probe, and evaluated their outcome against ERA5-LAND data. Because our testing period was aligned with their experiment, we were able to directly compare their results to ours.

Lees' probe method reported a median correlation against ERA5-LAND data of 0.9, surpassing our value of 0.86. Figure 2.6 shows a more detailed comparison. Both methods got similar results in basins with high correlation; however, Lees' method was more robust in low-performing basins. This effect can be directly linked to the explanation we gave above when a similar behavior was observed when predicting discharge. The LSTM network can account for biases in the forcing variables because mass conservation is not enforced, while our hybrid approach is limited by their mass conservative structure.

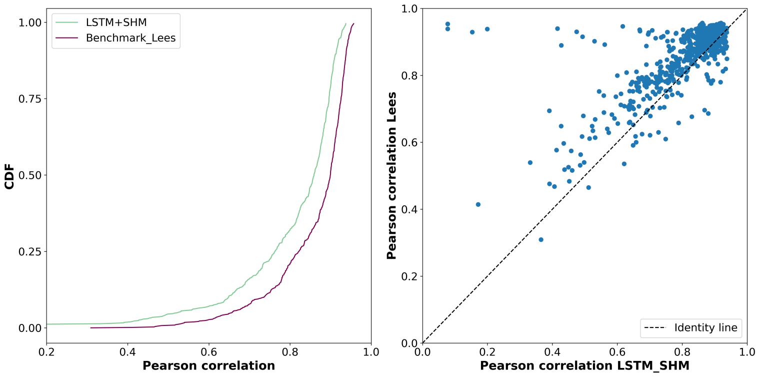

Figure 2.6: Comparison of soil moisture estimates using our hybrid approach method and Lees et al. (2022) approach. The simulated results of both approaches were compared against ERA5-Land data. Left: CDF for the correlation coefficient obtained by both models when applied to the testing dataset. Right: Comparison of correlations provided by both models.

The main difference with Lees et al. (2022) is that their probe to extract non-target variables needs to be trained, while in our hybrid approach no extra training needs to be done. The fact that in Lees et al. (2022) the probe can be as simple as a linear model, which requires few points to train, is not being argued, and in many cases, this will reduce the advantage given by our method.

Lastly, we would like to point out that the correlation obtained by the LSTM+Bucket (0.82) and LSTM+NonSense (0.85) models is still high, which can be attributed to the strong dependence of soil moisture with the precipitation and evapotranspiration series, both of which serve as boundary conditions for all models. This point also highlights our previously stated concern about using hybrid models for comparing different types of conceptual models or multiple working hypotheses.

In addition to the comparison with external data, we also examined the correlation between soil moisture estimates produced by the LSTM+SHM model and the standalone SHM. The median correlation value of 0.96 further confirms that the unsaturated zone within our hybrid model operates under our initial expectations. Figure 2.7 exemplifies this agreement for basin 42010, where the modelled (LSTM+SHM) and ERA5-LAND series exhibit a correlation of 0.86, equivalent to the median correlation observed across all 669 basins. For reference, the median correlation of the stand-alone SHM over all basins was also 0.86.

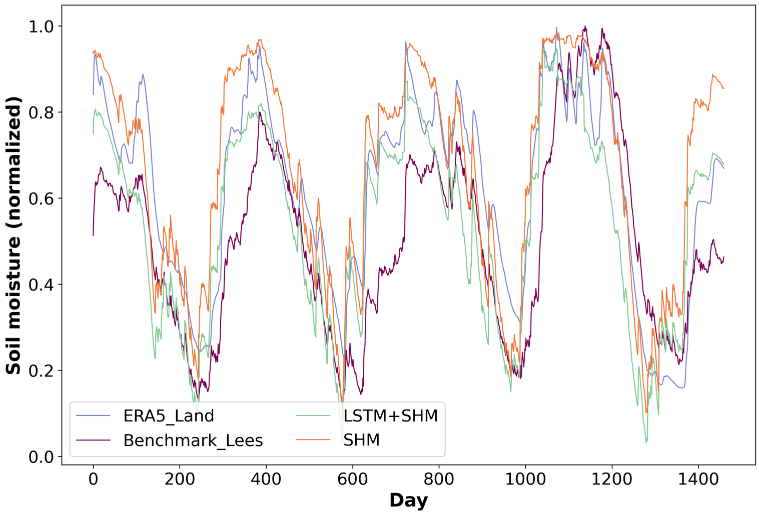

Figure 2.7: Soil moisture time series comparison during the testing period for basin ID 42010

The last experiment to further evaluate the consistency of our hybrid model was to analyze the parameter variation over time. Figure 2.8 presents the results for four calibration parameters: *su_max*, β, *kb*, and *ki*, across two different basins (reasons for choosing these two basins are explained below). We begin by examining the behavior of the first two parameters.

The purpose of *su_max* and β is to control the water transfer from the unsaturated zone reservoir to the interflow and baseflow, following Eq:(2.2):

$$qu\_out = qu\_in \cdot \left(\frac{su}{su\_max}\right)^{\beta}, \tag{2.2}$$

where *qu_out* represents the water going out of the unsaturated zone, *qu_in* represents the water entering the unsaturated zone from precipitation and snowmelt, *su* refers to the unsaturated zone storage or soil moisture, and *su_max* and β are the calibration parameters. It is important to note that the value of *su* cannot exceed *su_max*, which forces their quotient to be less or equal to one. Consequently, a larger value of *su_max* and/or β, leads to a decrease in the unsaturated zone outflow.

The parameter variation in basin ID 15016 presents clear seasonal patterns. During low-flow periods, both parameters increase, resulting in reduced water availability for the remaining two reservoirs and, consequently, a decrease in the total outflow. On the other hand, during high-flow periods, the opposite happens. As both parameters decrease, there is an increase in water availability, resulting in higher outflows.

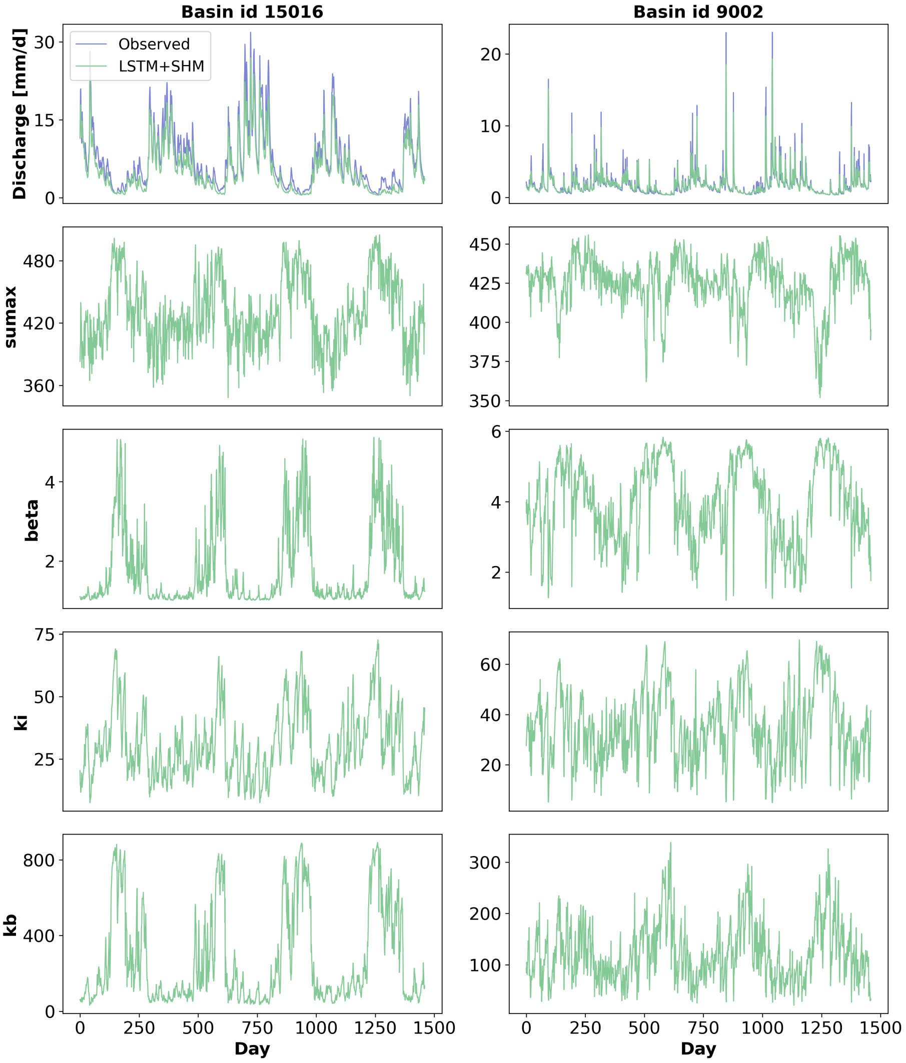

Figure 2.8: Time variation of parameters for basins 15016 (left column) and 9002 (right column). It should be noted that the Y-Axis ranges of the two basins differ.

Regarding the other two parameters, *kb*, and *ki* have a linear relationship with their respective outflows, acting as the denominator of their storage units ($q_{[i,b]} = \frac{s_{[i,b]}}{k_{[i,b]}}$). For *kb*, we can observe seasonal patterns, which allows the model to further increase the baseflow in wet periods and reduce it during dry seasons. This also aligns with our knowledge that hydraulic conductivity is lower when the soil is drier. On the contrary, *ki* displays faster variations.

Basin 9002 shows high-frequency variability for most of the parameters but still a good agreement between the observed and simulated discharge. This basin also exhibits a large increase in performance when the dynamic parameterization is applied, achieving an NSE of 0.90 for the LSTM+SHM against an NSE of 0.73 for the stand-alone SHM. This boost in performance when the dynamic parameterization is applied is 4 times as high as the equivalent scenario for basin 15016. We argue that this is the basis for explaining the high-frequency variability we see in the figure.

In a hypothetical case in which we have a perfect conceptual model that considers all the processes happening in the basin, the predicted parameters will be constant. However, due to structural limitations of the conceptual architecture, the LSTM does take part in the predicting. Because of how the LSTM and the conceptual model are connected, the only way for the former to pass predicting information to the latter is through the parameters. Some deficiencies can be compensated with a more seasonal pattern, while others need faster pulses.

An alternative approach is to increase the complexity of our process-based part, reducing the necessity of the data-driven method to compensate for structural deficiencies. For example, Feng et al. (2022) represent the catchment processes using 16 HBV models acting in parallel, which are parameterized through an LSTM. In their case, the recession parameters were predicted as constant in time, and the necessary flexibility to get state-of-the-art performance and account for missing sub-processes was considered by the semidistributed format.

It is worth considering that the bigger and more complex a process-based model is, the more similar it can become to an LSTM. More buckets generate higher flexibility, which can account for more complex process representation, such as different processes, multiple residence times, and mass transfers. Moreover, these buckets are normally updated considering some input and output fluxes and some losses. In an LSTM each cell state can be interpreted as a bucket, which can be modified by a forget gate, input gate, and (indirectly) output gate. The main difference is that the gates in the LSTM depend on the input and the previous hidden states, and in conceptual models, for simplicity, we usually take these as constants. However, this can be modified by making the gates of the buckets context dependent, and then both models would be alike. Therefore, regularizing a hybrid model with a complex conceptual model will probably reduce the work that needs to be done by the LSTM, but the final product will not be that different from having just a stand-alone LSTM.

2.4 SUMMARY AND CONCLUSIONS

In recent years, the idea of creating hybrid models by combining data-driven techniques and conceptual models has gained popularity, aiming to combine the improved performance of the former with the interpretability of the latter. Following this line of thought, Feng et al. (2022) used as a hybrid approach the parameterization of a process-based hydrological model by an LSTM network. The authors demonstrated the potential of the technique to achieve comparable performance as purely data-driven techniques and to outperform stand-alone conceptual models. Kraft et al. (2022) also achieved promising results following a similar process.

Motivated by this outcome, our article dug into the effect of dynamic parameterization in our conceptual model and the consequences this might have on the interpretability of the model. More specifically, we tried to answer the following questions. (1) Do conceptual models effectively serve as a regularization mechanism for the dynamic parameterization given by the LSTMs? (2) Does the dynamic parameterization of the data-driven component overwrite the physical interpretability of the conceptual model?

The first step towards answering these questions was to create a hybrid model. We coupled an LSTM network with a conceptual hydrological model (SHM), using the former as a dynamic parameterization of the latter. In our study, we demonstrated that our hybrid approach (LSTM+SHM) was able to achieve state-of-the-art performance, comparable to purely data-driven techniques (LSTM). Both models were trained in a regional context, using the CAMELS-GB dataset. The median NSE of 0.87 and 0.84 for the LSTM and LSTM+SHM, respectively, outperform the basin-wise calibrated conceptual model, which served as the baseline and achieved a median NSE of 0.76. These findings align with existing literature. For instance, Feng et al. (2022) reached similar conclusions when applying a hybrid model to the CAMELS-US dataset.

Having accomplished a well-performing hybrid model, we addressed the first research question. By modifying the regularization given by the conceptual model, we tested to which degree the dynamic parameterization given by the LSTM has the potential to compensate for missing processes. We proved that a hybrid model composed of an LSTM plus a single bucket (LSTM+Bucket) was able to achieve a similar performance as the LSTM+SHM and LSTM-only models. This indicates that the regularization given by the conceptual model is not strong enough to drop the predictive capability of the hybrid model, and missing processes are outsourced to the data-driven part. We also demonstrated that if we use an intentionally implausible structure (LSTM+NonSense), the LSTM also has the flexibility to artificially increase performance.

However, the fact that the data-driven component possesses this capability does not necessarily imply that a well-structured conceptual model cannot be consistently utilized by the LSTM. Therefore, we further analyzed the internal functioning of our LSTM+SHM model to answer our second research question. We compared the soil moisture predicted by our hybrid model with data from ERA5-LAND. This test addressed one of the main benefits of hybrid models over purely data-driven ones, which is their ability to predict untrained states. Across our testing set, comprised of 669 basins, we obtained a median correlation of 0.86 between our simulated soil moisture and the ERA5-LAND data. This result indicates that our hybrid model was able to produce coherent temporal patterns of the untrained state variables, without having access to the corresponding data during the training period. We also compared the unsaturated zone reservoir of the LSTM+SHM against the unsaturated zone of the stand-alone SHM, which reported a median correlation of 0.96. These results indicate that the dynamic parameterization was operating the unsaturated zone reservoir consistently and according to our initial expectations. The last section of the study presented the results of the dynamic parameterization for two basins, where we showed that the high-frequency variations of the parameter's time series are caused by the LSTM trying to compensate for structural deficiencies in our process-based model.

We summarize the key findings of our study as follows:

- 1. Do conceptual models serve as an effective regularization mechanism for the dynamic parameterization of LSTMs?

  No. Our initial expectation was that if (a) our head layer restricts the flexibility of the LSTM and (b) the process layer encodes almost no hydrological understanding, then the performance of the model would drop; however, this was not the case. This indicates that structural deficiencies in the architectures can be compensated by the data-driven part. Therefore, we recommend being careful about using this hybrid scheme for comparing different types of process-based models, especially if we are evaluating model adequacy by performance alone, as the overall performance can be adjusted by the data-driven part.

- 2. Does the data-driven dynamic parameterization compromise the physical interpretability of the conceptual model?

  Partially. We showed that a well-structured conceptual model maintains certain interpretability and even gives us access to untrained variables. However, we also showed that even with a well-structured conceptual model, the LSTM is going to compensate for missing processes and structural limitations, especially when the architecture of the process is not well suited for a specific case. Increasing the complexity of the process-based model would result in less intervention of the data-driven part; however, we argue that the more complex a process-based model is, the more similar it will be to an LSTM network.

- To bucket or not to bucket?

  In our experiments, we were not able to increase the performance of the data-driven models by adding a conceptual head layer, and even though the mean performance of the different models was the same, purely data-driven methods showed better results in low-performing basins. Therefore, until this point, the remaining advantage is the access to non-target variables, which other authors have accomplished with the use of probes. In future research, we will conduct other experiments to evaluate the performance of hybrid models under different conditions, but until this point, we do not have evidence that adding buckets gives a considerable advantage over purely data-driven techniques.

Part III

ANALYZING THE GENERALIZATION CAPABILITIES OF A HYBRID HYDROLOGICAL MODEL FOR EXTRAPOLATION TO EXTREME EVENTS

This study is published in the scientific journal Hydrology and Earth System Science (HESS). The remainder of part III is a reprint of:

Acuña Espinoza, E., Loritz, R., Kratzert, F., Klotz, D., Gauch, M., Álvarez Chaves, M., and Ehret, U.: Analyzing the generalization capabilities of a hybrid hydrological model for extrapolation to extreme events. Hydrology and Earth System Sciences, 29(5), 1277–1294. https://doi.org/10.5194/hess-29-1277-2025*, 2025.*

3

ANALYZING THE GENERALIZATION CAPABILITIES OF A HYBRID HYDROLOGICAL MODEL FOR EXTRAPOLATION TO EXTREME EVENTS

ABSTRACT

Data-driven techniques have shown the potential to outperform process-based models in rainfall–runoff simulation. Recently, hybrid models, which combine data-driven methods with process-based approaches, have been proposed to leverage the strengths of both methodologies, aiming to enhance simulation accuracy while maintaining a certain interpretability. Expanding the set of test cases to evaluate hybrid models under different conditions, we test their generalization capabilities for extreme hydrological events, comparing their performance against long short-term memory (LSTM) networks and process-based models. Our results indicate that hybrid models show performance similar to that of the LSTM network for most cases. However, hybrid models reported slightly lower errors in the most extreme cases and were able to produce higher peak discharges.

3.1 INTRODUCTION

Data-driven techniques have demonstrated the potential to outperform process-based models in rainfall–runoff simulation (Feng et al., 2020; Kratzert et al., 2019b; Lees et al., 2021). Moreover, Frame et al. (2022) addressed concerns about the generalization capability of data-driven methods for extrapolation to extreme events, demonstrating that long short-term memory (LSTM) networks (Hochreiter and Schmidhuber, 1997) outperformed process-based models in such scenarios.

Recently, hybrid models that combine process-based models with data-driven approaches have been proposed (Reichstein et al., 2019; Shen et al., 2023). The idea behind hybrid models is that they integrate the strengths of both process-based and data-driven approaches to improve simulation accuracy while maintaining a notion of interpretability (Hoge et al., 2022; Jiang et al., 2020). Among the various approaches available to combine these methodologies, the parameterization of process-based models using data-driven techniques has shown promising results (Tsai et al., 2021). One way to interpret this technique is that it involves integrating a neural network with a process-based model in an end-to-end pipeline, where the neural network handles the parameterization of the process-based model. Alternatively, this can be viewed as a neural network with a process-based head layer, which not only compresses the information into a target signal but has a certain structure that allows for the recovery of untrained variables. Kraft et al. (2022) applied this method, demonstrating that substituting poorly understood or challenging-to-parameterize processes with machine learning (ML) models can effectively reduce model biases and enhance local adaptivity. Similarly, Feng et al. (2022) and Acuña Espinoza et al. (2024b) employed LSTM networks to estimate the parameters of process-based models, achieving state-of-the-art performance comparable with LSTMs and outperforming stand-alone conceptual models.

In a previous study, Acuña Espinoza et al. (2024b) tested the performance and interpretability of hybrid models, with the overall goal of looking at the advantages provided by adding a process-based head layer to a data-driven method. They show that hybrid models can achieve comparable performance with LSTM networks while maintaining a notion of interpretability. In this type of hybrid model, the interpretability arises from association. Specifically, the authors map the parameters and components of process-based models to predefined processes, domains, and states (e.g., baseflow, interflow, snow accumulation). While this approach does provide interpretability, it is important to clarify that this interpretability is, indeed, limited to associations and may lack rigorous physical principles, especially when one uses models such as the simple hydrological model (SHM) (Ehret et al., 2020) or the Hydrologiska Byråns Vattenbalansavdelning (HBV) (Bergström, 1992), which present significant simplifications of the underlying physical process. Moreover, Acuña Espinoza et al. (2024b) also warn about the possibility that the data-driven section of the hybrid model compensates for structural deficiencies in the conceptual layer.

Building on this research line and expanding the set of test cases to evaluate hybrid models under different conditions, this study follows the procedure proposed by Frame et al. (2022) to investigate the ability of different models to predict out-of-sample conditions, focusing on their capability to generalize to extreme events. We compare the performance of hybrid models against both traditional process-based models and stand-alone data-driven models. We aim to determine which model demonstrates higher predictive accuracy, particularly in simulating extreme hydrological events. We thereby address the following two research questions:

- How does a hybrid model compare to a process-based model and a stand-alone data-driven model in the simulation of extreme hydrological events?
- Does hybrid modeling offer a higher performance than stand-alone data-driven approaches?

To achieve this objective, we have structured this article as follows: Sect. 3.2 describes the training data–test data split and gives an overview of the different models. In Sect. 3.3, we compare the results of various tests that assess the generalization capabilities of data-driven, hybrid, and conceptual models. Lastly, Sect. 3.4 summarizes the key findings of the experiments and presents the conclusions of the study

3.2 DATA AND METHODS

Donoho (2017) emphasize the importance of community benchmarks in driving model improvement. In the hydrological community, this practice has also been suggested to enable a fair comparison between new and existing methods (Kratzert et al., 2024; Nearing et al., 2021; Shen et al., 2018). Consequently, we built our experiments considering two existing studies. First, we used the procedure proposed by Frame et al. (2022) to evaluate the generalization capability of different models (see Sect. 3.2.1). In accordance with this study, the experiments were conducted using the CAMELS-US dataset (Addor et al., 2017; Newman et al., 2015) in the same subset of 531 basins. Second, we used the hybrid model architecture $\delta_n(\gamma^t, \beta^t)$, further explained in Sect. 3.2.2.2, as proposed by Feng et al. (2022). This architecture demonstrated competitive performance compared to LSTM networks in their original experiments, which also used the CAMELS-US dataset.

3.2.1 *Data handling: training/test split*

To produce an out-of-sample test dataset and to evaluate the generalization capability of the different models for extreme streamflow events, we split the training and test sets by years based on the return period of the maximum annual discharge event. Closely following the procedure recommended by Bulletin 17C (England Jr et al., 2019), we fitted a Pearson III distribution to the annual maxima series of each basin, which we extracted from the observed CAMELS-US discharge records. We then calculated the magnitude of the discharge associated with different probabilities of exceedance. Using the discharge associated with the 5-year return period as a threshold, we classified the water years (here, a water year is defined as the period of time between 1 October and 30 September of the following year) into training or test sets. Figure 3.1a shows an example of the training–test split for basin no. 01054200 in the northeast of the United States (US). The water years which contained only discharge records smaller than the associated 5-year threshold were used for training, while cases in which this threshold was exceeded were used for testing. It is important to note that there was a 365 d buffer between each training and testing period. The value of 365 d corresponds to the sequence length used by the LSTM model, and the buffer period avoids the leaking of test information during training. The results of the frequency analysis and the training data–test data split for each basin can be found in Acuña Espinoza (2024). The original dataset contained 531 basins, each with 34 years of data (from 1980 to 2014), for a total of 18 054 years of data. After the data split process, 9489 years were used for training, 3429 were used for testing, and 5136 years were buffers. Excluding the buffer data, 73% of the data were used for training and 27% were for testing. This distribution is consistent with the 80%–20% theoretical split associated with the 5-year return period.

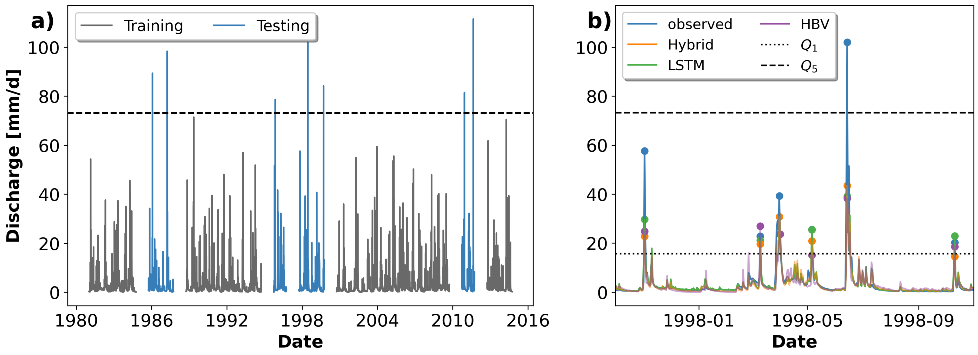

Figure 3.1: (a) Observed discharge (mmd^{-1}) from 1984 to 2016 for catchment ID no. 01054200. Gray lines represent training data, including discharge below the 5-year return period threshold, marked by the dashed line (Q_5). Blue lines indicate test data for discharge exceeding this threshold. This training–test split, based on discharge exceedance probability, is designed to assess model performance under extreme hydrological conditions. (b) Example of peak identification for basin no. 01054200 for 1 year within the test period. Q_1 represent the 1-year return period threshold, which was used to identify peak events. Q_5 represent the 5-year return period threshold, which was used for the training data–test data split.

It should be noted that the results from the training data–test data split differed from the ones proposed by Frame et al. (2022). In their study, the frequency analysis was done with instantaneous peak flow observations taken from the USGS NWIS (US Geological Service, 2016), and a maximum cap of 13 water years was used to train each basin. Instead, we used the observed daily data from the CAMELS-US dataset and did not impose restrictions on the maximum number of training years. We would also like to re-emphasize that training the model exclusively on water years containing events smaller than a 5-year return period and testing it on water years with events larger than a 5-year return period was meant as a form of stress test to get a sense of the model behavior with regard to extreme-streamflow events. In practical applications, one would not choose to use this type of setup, but one should use all available information about this kind of event for model training.

3.2.2 *Data-driven, hybrid and conceptual models*

The experiments in this study were conducted using three models: a stand-alone LSTM, the HBV model as a stand-alone conceptual model, and a hybrid approach. Both the LSTM and the hybrid model were trained using the NeuralHydrology (NH) package (Kratzert et al., 2022), while the optimization of the stand-alone conceptual model used the SPOTPY library (Houska et al., 2015). Consistently with previous studies, the LSTM and hybrid models were trained regionally using the information from all 531 basins at the same time, while the stand-alone conceptual model was trained basin-wise (locally). In other words, in this study, we compare the model results of an LSTM network, a hybrid model, and 531 individually trained conceptual models.

3.2.2.1 *LSTM*

The hyper-parameters for the stand-alone LSTM were taken from Frame et al. (2022). We used a single-layer LSTM with 128 hidden states, a sequence length of 365 d, a batch size of 256, and a dropout rate of 0.4. The optimization was done using the Adam algorithm (Kingma and Ba, 2014). An initial learning rate of 10^{-3} was selected, which was decreased to $5x10^{-4}$ and 10^{-4} after 10 and 20 epochs, respectively. The basin-averaged Nash–Sutcliffe efficiency (NSE) loss function proposed by Kratzert et al. (2019b) was used for the optimization. In a slight deviation from Frame et al. (2022), we trained our model for 20 epochs instead of 30. We trained our models using five dynamic inputs from the Daymet forcing – precipitation (mmd^{-1}), average shortwave radiation (Wm^{-2}), maximum temperature (°C), minimum temperature (°C), and vapor pressure (Pa) – and 27 static attributes, listed in Table A1 in the Appendix of Kratzert et al. (2019), describing the climatic, topographic, vegetation, and soil characteristics of the different basins.

We used an ensemble of five LSTM networks to produce the final simulated discharge. In other words, we trained five individual LSTM models with the architecture described above but initialized each one using a different random seed. After training, we ran each model, individually, to retrieve the simulated discharges, and we took the median value as the final discharge signal, which we used in the analysis. Using an ensemble of LSTM networks allows us to produce more robust results (Gauch et al., 2021; Kratzert et al., 2019b; Lees et al., 2021) and reduces the effects associated with the random initialization of the models.

3.2.2.2 *Hybrid model: LSTM+HBV*

For the hybrid model architecture, we used the $\delta_n(\gamma^t, \beta^t)$ model proposed by Feng et al. (2022). In this architecture, an ensemble of 16 HBV models acting in parallel was parameterized by a single LSTM network. Each of the 16 ensemble members contained an HBV model with four buckets, whose flows were controlled by 11 static and 2 time-varying parameters. The discharge of the ensemble was calculated as the mean discharge of the 16 members. Moreover, to produce the final outflow, the ensemble discharge was routed using a two-parameter unit hydrograph. In total, the LSTM produced 210 parameters (16 ensemble members, each with 13 HBV parameters and 2 routing parameters) which were used to control the ensemble of conceptual models, as well as the routing scheme. The model was trained end to end. Figure B.1 in Appendix B.1 shows a scheme of the hybrid model structure

During training, each batch contained 256 samples, each with a sequence of 730 d. The first 365 d were used as a warmup period to stabilize the internal states (buckets) of the HBV and to reduce the effect of the initial conditions. These 365 values did not contribute to the loss function. The remaining 365 time steps were used to calculate the loss, back-propagate the gradients, and update the model's weights and biases. Further details on the model implementation can be found in the *hybrid_extrapolation_seed#.yml* files of the Supplement. To validate our pipeline, we benchmarked our hybrid model implementation using the experiments proposed by Feng et al. (2022). These results are shown in Appendix B.1, where we show a similar performance between our implementation and their results. Only after validating our pipeline did we run the extrapolation experiments.

3.2.2.3 *Stand-alone conceptual model: HBV*

To have a full comparison of the model spectrum, we also included a stand-alone conceptual model. We used a single HBV model plus a unit hydrograph routing routine, resulting in a model with 14 calibration parameters (12 from the HBV and 2 from the routing). Note that this HBV instance has one less parameter (12 instead of 13) than the version used in the hybrid model. This one-parameter difference is to maintain consistency with Feng et al. (2022), where the authors used the 13-parameter HBV only when dynamic parameterization was included and the 12-parameter model for the static version. Similarly to Acuña Espinoza et al. (2024b), the stand-alone conceptual models were trained basin-wise using shuffled complex evolution (SCE-UA) (Duan et al., 1994) and DiffeRential Evolution Adaptive Metropolis (DREAM) (Vrugt, 2016), both implemented in the SPOTPY library (Houska et al., 2015). We then selected, for each basin, the calibration parameters that yielded better results.

3.2.3 *Performance metrics*

To evaluate the overall performance of the model we used Nash–Sutcliffe efficiency (NSE) (Nash and Sutcliffe, 1970). However, the main objective of this study is to evaluate the ability of the models to predict high-flow scenarios. Therefore the majority of the analyses were done using only peak flows.

Given the amount of data comprised in the test period (3429 years over the 531 basins), the peak identification was done automatically. For this task, we used the find_peaks function of the signal module in the SciPy library (Virtanen et al., 2020), defining a 7 d window as the criterion for independent events. Moreover, we selected only the peaks above the 1-year return period threshold to have a better representation of high-flow scenarios. After we identified the peaks in the observed discharge series, we extracted the associated values from the simulated series of the different models. Figure 3.1b exemplifies this process for basin no. 01054200 for 1 year within the test period, where each dot represents an identified peak. Once the peaks were identified, we calculated the absolute percentage error (APE) as a metric for model performance:

$$APE = \frac{|y_{obs} - y_{sim}|}{y_{obs}}, \tag{3.1}$$

where y_{obs} and y_{sim} are the observed and simulated discharge, respectively.

3.3 RESULTS AND DISCUSSION

After training the models, we performed multiple analyses to evaluate their generalization capabilities to extreme events. The results of these analyses are presented in this section. All the results discussed here are for the test period.

3.3.1 *Model performance comparison for whole test period*

Figure 3.2 shows the cumulative distribution functions (CDFs) for the NSE reported for each model over the whole test period. We can see that the LSTM outperforms the hybrid model, with median NSE values of 0.75 and 0.71, respectively. Moreover, both models outperform the stand-alone HBV model, which has a median NSE value of 0.64. The result that both the LSTM and the hybrid model outperform the stand-alone HBV is not surprising, and similar results have been reported in the literature (Acuña Espinoza et al., 2024b; Feng et al., 2022; Kratzert et al., 2019b; Lees et al., 2021; Loritz et al., 2024). This can be attributed to the fact that conceptual models present a relatively simple structure that, in a lot of cases, oversimplifies the actual physical processes. For example, the HBV model assumes that all flows have a linear relationship with the storage; that the storage and/or discharge rate does not change over time; and that snow melting is a linear process, proportional to the difference between a threshold temperature and the air temperature. Both the LSTM and hybrid model have more flexible frameworks that allow them to increase their performance. Moreover, we show that, even with a different training–test split compared to the usual temporally contiguous subsets, our results are consistent with the ones reported by Feng et al. (2022) and Acuña Espinoza et al. (2024b), where the same model ranking was observed.

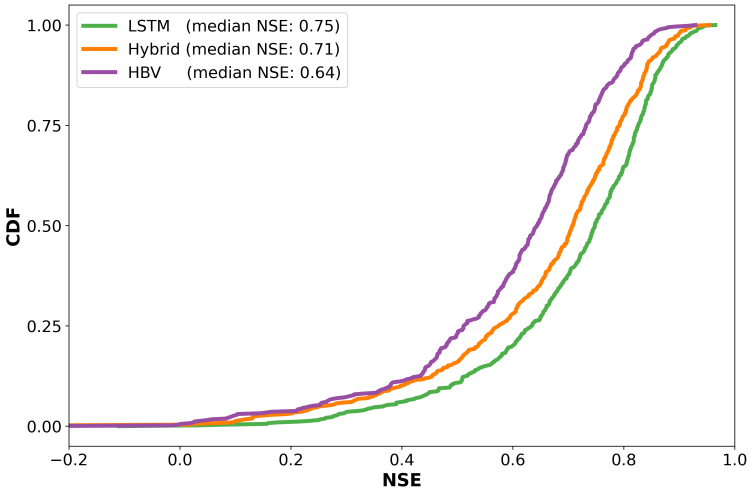

Figure 3.2: Cumulative density function (CDF) of the Nash–Sutcliffe efficiency (NSE) for the different models, generated using 531 basins of the CAMELS-US dataset. The NSE was calculated over the whole test period of each basin.

3.3.2 *Model performance comparison for peak flows*

The metrics shown in Fig. 3.2 were calculated using the whole test period. Consequently, they summarize the overall performance of the three models. However, the main objective of this study is to evaluate the ability of the models to predict high-flow scenarios. For this, we used the APE metric, which we calculated following the procedure described in section 3.2.3. Figure 3.3a presents, for each model, the distribution of the APE for all of the peak flows. This figure shows a similar distribution for the three models, with the LSTM presenting a slightly lower median error than the hybrid and stand-alone HBV. Lower values are better for the APE metric. The finding that LSTMs outperform process-based models aligns with Frame et al. (2022) and helps to challenge the notion that data-driven methods are less capable of extrapolation (Reichstein et al., 2019; Slater et al., 2023). In the case of the hybrid model, although the LSTM exhibits a slightly lower median error, the error distributions of both models are similar. Therefore, we do not find strong evidence suggesting that one architecture is significantly better than the other in this scenario, leaving it to the reader's discretion to choose the model that best suits their needs.

3.3.3 *Model performance comparison for out-of-sample peak flows*

Figure 3.3a allowed us to evaluate the performance of the models only in peak discharges. However, this still did not give us a performance metric for values exclusively outside of the training range. More specifically, the results presented in Fig. 3.3a evaluated the error in 17 580 observed events. Considering the fact that the test period contained 3429 years, we got an average of five peaks per year. However, as shown in Fig. 3.1b, these peaks were not necessarily larger than the 5-year return period thresholds used during training. Figure 3.3b shows the same error metric but classifies the peaks based on their return period. The four categories to the right of the vertical dashed line present the errors associated with discharges beyond the 5-year return period threshold, giving a strict evaluation of the generalization capabilities of the models. We can see that the LSTM outperformed the hybrid and HBV models slightly in the 1–5-year and 5–25-year return periods. In the remaining three intervals, the performance of the LSTM and hybrid are comparable, with the HBV also showing similar behavior for the last one. Figure B.3 in Appendix B.2 shows the variation in the APE due to five random initializations of the models. We can see that, in the last three categories, the LSTM performed better in three cases, the hybrid performed better in seven cases, and they both reported the same median value in four cases. Moreover, the differences in the median values between the LSTM and the hybrid model are smaller than the metric variation due to the random initialization, supporting our hypothesis that, for higher return periods, all models perform similarly.

In most cases, the errors increased for higher return periods. This was expected as models were trying to generalize to flows farther away from their training range. On the other hand, the peaks of the return period of 100+ years presented similar or slightly lower errors than the ones in the 50–100-year category. At this point, the reported errors were close to 60%, which indicated that no model could satisfactorily reproduce the observed peaks. Moreover, because of the characteristic of the metric (see Eq. 3.1), the error was scaled by the magnitude of the observation, which would explain why the 50–100 and 100+ presented similar errors.

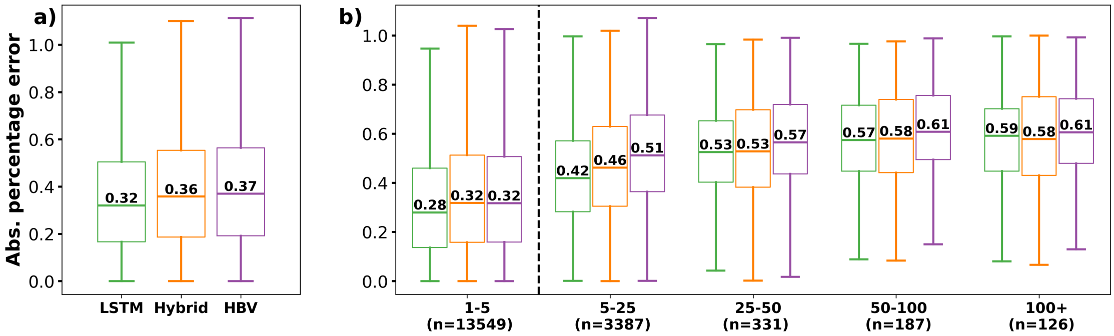

Figure 3.3: a) Absolute percentage error (APE) between the observed peak discharge and the associated simulation value for the different models. The results show the error distribution, from all 531 basins, calculated only for the peak flows of their test period (total of 17580 values) b) APE, classified by the return period of the observed peaks. The four categories to the right of the dashed vertical line present the errors associated with observed discharge above the 5-year return period threshold, evaluating the out-of-sample capabilities of the models. The n-value below each category indicates the amount of data used to produce the box-plot.

3.3.4 *Spatial analysis of model performance*

To further understand the capacity of the different models, we evaluated their performance in space, examining how predictive accuracy varies across regions and identifying differences between the models. As shown in Fig. B.4, in Appendix B.3, all models exhibited lower performance in the basins located in the Great Plains (center) and in the southwest of the US. This pattern, previously documented in the scientific literature for both data-driven methods (Gauch et al., 2021) and process-based models (Newman et al., 2015), is also observed in the hybrid architecture, as demonstrated here. Martinez and Gupta (2010) identified the aridity (PET/P) and runoff ratio (Q/P) as good predictors of model performance. The comparison between Figs. B.4 and B.6 supports this hypothesis, revealing an association between lower performance and regions characterized by high aridity and low runoff ratios.

Given the shared behavior between the models, and in line with the objective of this comparison study, we further evaluate the differences in their performance. Figure 3.4 shows the difference between the hybrid and the LSTM model, calculated as $\text{APE}_{\text{hybrid}}$-$\text{APE}_{\text{LSTM}}$. Consequently, negative (blue) values indicate that the hybrid model performed better (i.e., presented lower error), while positive (red) values favor the LSTM.

The first spatial tendency that we can notice is that, along the central part of the US, the LSTM tends to perform better. As previously discussed, this area is characterized by arid basins. In arid basins, surface runoff can be associated with high-intensity events over a short time period. Moreover, the HBV model generates runoff under the assumption that the discharge is a function of the basin storage and lacks a direct channel to transform precipitation into surface runoff (e.g., the water always routes through a linear reservoir). This assumption might not hold for the runoff-generating process in arid basins. Considering the fact that the hybrid model is regularized by an HBV layer, this structural deficiency would explain why the LSTM outperforms the hybrid model in this area.

The second spatial tendency that we can observe is that, for the map of the return period of 100+ years, there is a cluster over the Pacific Northwest in which the hybrid model outperforms the LSTM. These basins are characterized by high precipitation values (see Fig. B.6c) and high discharges. This is a challenge for the LSTM architecture due to saturation problems, which will be explained in detail in the next section.

Figure B.4 further presents the differences between the HBV and LSTM models, as well as between the hybrid model and HBV. In the first case, the LSTM outperforms the HBV in most cases, with the same exception of the northwestern coast for the return period of 100+ years. In the second case, the hybrid model presents an overall better performance.

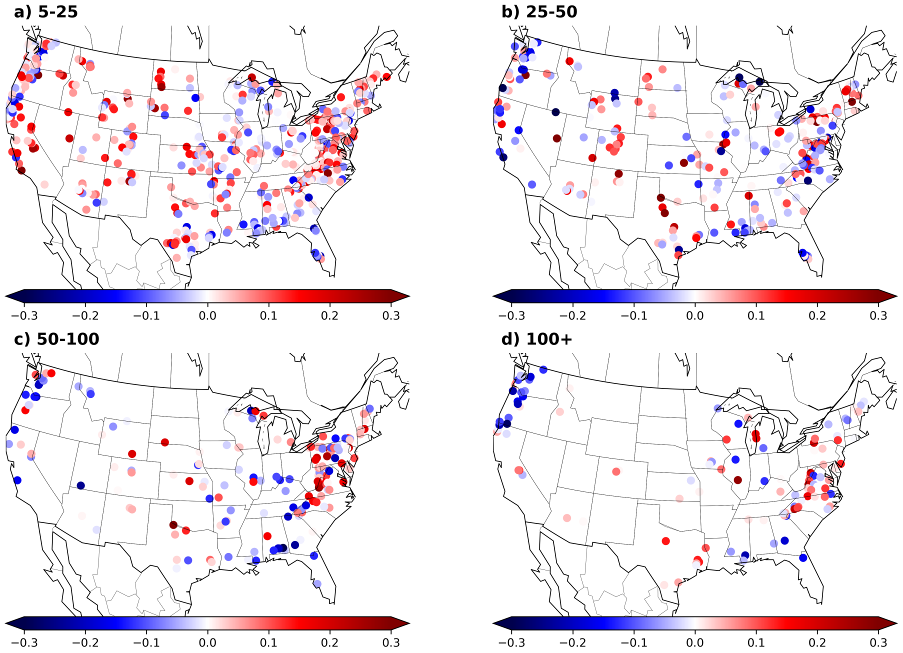

Figure 3.4: Spatial visualization of the difference between the absolute percentage error (APE) for the hybrid and LSTM models. Each point corresponds to one basin. The four maps are associated with the different return periods. The color scale indicates the difference between the median APE for the different models. The difference is calculated as $\text{APE}_{\text{hybrid}}$-$\text{APE}_{\text{LSTM}}$; therefore, negative (blue) values indicate that the hybrid model performs better, and positive (red) values indicate that the LSTM performs better.

3.3.5 *Saturation analysis: behavior of the models during extreme flow scenarios*

The saturation problem in LSTM models with a single linear head layer (as described by Kratzert et al., 2024) arises due to the inherent limitations of the model architecture, resulting in a theoretical prediction limit (see Eq. B2 of Kratzert et al., 2024). In other words, independently of the input series, the associated prediction cannot go above the theoretical limit. This limit is a function of the weights and biases of the head linear layer and varies for each trained model instance. From our experiments, where the model is trained only on discharge values below a 5-year return period, we observed that the maximum value predicted by the LSTM was 78.9mmd^{-1}, which is close to the calculated theoretical prediction limit of 83.9mmd^{-1}. This saturation limit could explain the LSTM's low performance for the cluster of basins in the Pacific Northwest of the US for the return period of 100+ years. Out of the 10 basins with the highest errors within this cluster, 7 of them presented discharges for the return period of 100+ years that were above the saturation limit of the LSTM. Therefore, independently of the forcing series, the model was not able to reach such values. Figure 3.5a shows an example in which this saturation problem is particularly pronounced. On the other hand, both the hybrid and the stand-alone HBV model do not have such a theoretical limit. The conceptual model architecture is defined with an unlimited capacity in the buckets, and, due to mass conservation, the water received by the models after evapotranspiration and other abstractions must be accounted for within the system.

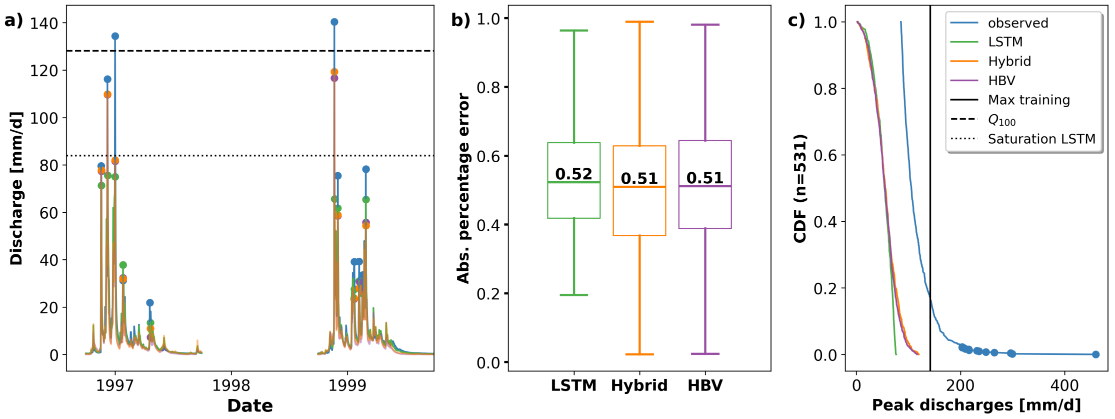

Figure 3.5: (a) Observed and simulated discharges for 2 years of the testing period for basin no. 11532500. The dashed and dotted lines indicate the 100-year return period discharge and the theoretical saturation limit of the LSTM model, respectively. (b) Absolute percentage error (APE) of the 531 highest discharges for the different models. (c) Cumulative density function (CDF) of the 531 observed highest discharge values across all basins and their respective simulated values. The blue dots help visualize the fact that less than 3% of the events have values between 200 and 400 mmd^{-1}.

The theoretical prediction limit in the LSTM is a function of the weights and biases of the head layer, which are a result of the training process. In our experiment, we artificially restricted the training data to discharges smaller than the 5-year return period thresholds, reducing the support of the data space the model was fitted to. Consequently, this setup directly intensified the saturation problem. In practical applications where the model is trained on all available data, the saturation issue would tend to decrease in relevance. Moreover, it would only affect the few gauges in which the extreme discharges are above the saturation limit. However, a theoretical saturation limit remains, which is an undesirable property in a hydrological model, especially in cases where we are designing infrastructure for extreme events outside of any training data (e.g., 1000-year flood). Further research should be invested in overcoming this problem.

Apart from the statistical artifacts introduced by our selection procedure, we found two potential issues that might lead to the peak underestimation of the hybrid models.

Figure 3.6 shows the precipitation and observed discharge, together with the accumulated value and the simulated discharge series, for four of the largest events in the dataset.

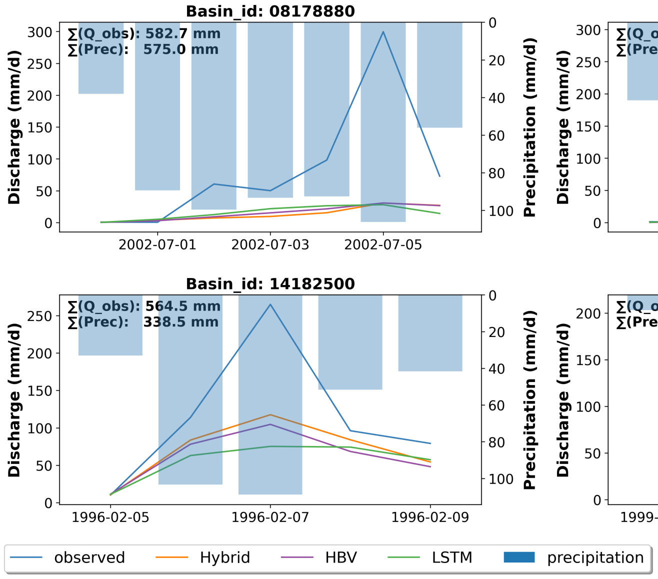

Figure 3.6: Example of 4 of the most extreme events presented in the dataset. The subplots show the precipitation series and the observed and simulated hydrographs. ∑Q_obs and ∑Prec indicate the cumulative sum of the discharge and precipitation series. Basins 08178880 and 08200000 have similar precipitation and discharge volumes while basins 14182500 and 14138800 have a precipitation volume smaller than the discharge volume.

For basin nos. 08178880 and 08200000, the accumulated precipitation of the event is similar to or larger than the accumulated discharge; however, the simulated series strongly underestimates the discharge. This behavior can arise due to structural limitations in the hydrological model. For example, given the lack of a fast response channel, a high precipitation pulse can be divided and routed through several linear reservoirs, attenuating the respective discharge peak. This effect could have been strengthened by our training–test split, given that the optimization parameters, which control the interaction between the buckets, were learned for certain conditions, which were inadequate for other out-of-sample hydrological events.

In this regard, the hybrid model presents a theoretical advantage over the HBV model through the possibility of dynamic parameterization that adapts the model behavior to current conditions. The $\delta_n(\gamma^t, \beta^t)$ hybrid model uses a dynamic β coefficient to control the recharge rate at which precipitation was transferred to the other buckets. However, as discussed in the previous paragraph and as shown in Fig. B.7, during high-intensity events, β reached the limits of its predefined interval, which limited the model in further adapting its behavior.

The second issue that we found is a possible bias in the input data. For basin nos. 14182500 and 14138800, the accumulated precipitation is smaller than the accumulated discharge, which would explain the underestimation of the simulated values. Westerberg and McMillan (2015) indicate that bias in precipitation measurements can be caused by point uncertainty, interpolation uncertainty, and equipment malfunction. Moreover, Bárdossy and Anwar (2023) indicate that catchment-averaged precipitation values present a higher bias during extreme events. This bias poses a challenge for the HBV and hybrid models, which rely on a mass-conservative structure. Without sufficient water input into the system, the model inherently cannot replicate the observed discharges.

An alternative hypothesis to precipitation bias is that the high discharges are caused by snowmelt or glacier melt. In this case, the accumulated precipitation during the event can be smaller than the accumulated discharge since part of the discharge would be caused by melting water that entered the system weeks or months before. Both basins are located in the state of Oregon (northwestern USA), and, accounting for the dates of both events, there is a possibility for snowmelt-induced discharge. Nevertheless, the snow module of HBV does not reproduce this behavior, which would point towards a structural deficiency in the model.

3.3.6 *Limitations and uncertainties*

In this study, we focus our results and analysis on model performance as we tackled our research questions from a practitioner's point of view. However, it should be noted that other criteria, such as model interpretability, also play an important role in an integral model evaluation as these allow us to assess whether the generated results align with expected domain-specific behaviors. Appendix B.4 briefly discusses the temporal variation in the dynamic parameters for the hybrid model in this context. We further refer to studies such as those of Acuña Espinoza et al. (2024b), Kraft et al. (2022), Hoge et al. (2022), and Feng et al. (2022) for a deeper analysis of model interpretability for hybrid models.

Following the procedure proposed by Frame et al. (2022), we split the training and test periods by years based on the return period of the maximum annual discharge. Both periods used information from the previously selected 531 basins. In future work, one could use different subsets of basins during training and testing to further compare the models based on an ungauged basin scenario and evaluate their spatial generalization capability

The comparison presented here was done using the hybrid architecture proposed by Feng et al. (2022). This architecture was chosen because it gave a competitive performance compared to LSTMs in their original experiment and because the code was open source. However, other architectures might lead to different results. From the analysis presented above, we hypothesized that a process-based layer that includes a fast-flow channel might improve performance. Moreover, expanding the hybrid model's parameter range or the number of dynamic parameters might be beneficial. However, the added flexibility might come at the expense of model interpretability. Additionally, considering the fact that catchment-averaged precipitation values present a higher bias during extreme events, strategies to overcome this limitation should be tested. Given the non-mass-conservative structure of the LSTM, systematic input biases can be accounted for. However, similar strategies should be evaluated for the hybrid case (e.g., considering a dynamic parameter that factorizes the precipitation input). Furthermore, other strategies to create a hybrid model, such as component replacement, should also be tested (Hoge et al., 2022; Kraft et al., 2022). We encourage the hydrological community to expand the test cases presented here.

The training–test split applied in this study was intended as a form of stress-testing to get a sense of a model's capacity to generalize to unseen events. For the reasons stated in previous sections, this stress-testing method directly affects the saturation problem in the LSTM and the parameter optimization for the hybrid and conceptual models. In a practical case, one should not remove low-probability events from the training data. Furthermore, in an operational setting, all the data should be included during model training to increase the performance of the models (Nevo et al., 2022).

Lastly, differences between simulated and observed values, especially in extreme events, can also be attributed to higher uncertainty in the observed quantities, including discharge and precipitation (Bárdossy and Anwar, 2023; Di Baldassarre and Montanari, 2009; Westerberg and McMillan, 2015). We did not consider this type of uncertainty in our analysis as this would be outside of the scope of the paper.

3.4 SUMMARY AND CONCLUSIONS

In this study, we evaluated the generalization capabilities of data-driven, hybrid, and conceptual models for predicting extreme hydrological events. Following the methodology proposed by Frame et al. (2022), we partitioned our data based on occurrence frequencies using the 5-year return period discharge as a threshold. We trained our models using information from water years with discharges strictly lower than the threshold and tested their performance on low-probability data. This setting was meant as a form of stress-test to get a sense of the model behavior with regard to extreme-streamflow events. Our findings indicated that the LSTM outperforms the hybrid and HBV models slightly for 1–5-year and 5–25-year return periods, and all models show similar performance for higher discharges.

The spatial analysis of the models' performance revealed that all three models exhibited higher errors in more arid basins, consistently with findings reported in the literature (Gauch et al., 2021; Martinez and Gupta, 2010; Newman et al., 2022). Regarding the differences between the models, the LSTM presented lower errors than the hybrid model in more arid basins, particularly for events in the 5–25-year and 25–50-year return period categories. This disparity can be attributed to the structural limitations of the HBV model, which assumes that discharge is a function of basin storage, a premise that may not align with runoff-generating processes in arid basins. Given that the hybrid model is regularized by an HBV layer, this structural deficiency would explain why the LSTM outperforms the hybrid model in this area. On the contrary, for the events of the return period of 100+ years, the hybrid model outperformed the LSTM in a cluster of basins located on the northwestern coast of the US. This behavior can be linked to the LSTM's theoretical discharge saturation limit, which is determined during training. In the northwestern cluster, the event's magnitude exceeded the saturation limit of the LSTM, preventing the LSTM from simulating the observed discharges. As discussed before, the training–test split used in this study artificially increased the saturation problem as it constricted the data space the model was fitted to. In practice, this problem can be attenuated by also considering low-probability events during training. Nevertheless, a theoretical limit will still exist, which is not a desirable property for a hydrological model. Additional research to overcome this limitation should be encouraged.

Expanding on the analysis to include the most extreme scenarios across the whole dataset, it can be concluded that all models underestimated the scenarios with the most extreme flow. However, the hybrid model and HBV were able to simulate higher discharges than the LSTM and presented an error distribution with longer tails toward smaller values. Upon further investigation, we noticed that the reasons for underestimating the extreme-flow scenarios were different. For the LSTM, its saturation limit was reached. On the other hand, the hybrid and HBV models underestimated the discharge due to structural deficiencies and possible bias in the input data. The dynamic parameterization of hybrid models might help reduce the former by changing the model response based on current conditions. This idea is conceptually similar to how an LSTM operates, with the gate structures operating based on current and past conditions.

Overall, in most of the experiments performed here, we did not find strong evidence suggesting that there is a significant difference between the extrapolation capabilities of LSTM networks and hybrid models. However, hybrid models did report slightly lower errors in the most extreme cases and were able to produce higher peak discharges. We leave it to the reader's discretion to choose the model that best suits their needs.

Part IV

AN APPROACH FOR HANDLING MULTIPLE TEMPORAL FREQUENCIES WITH DIFFERENT INPUT DIMENSIONS USING A SINGLE LSTM CELL

This study is published in the scientific journal Hydrology and Earth System Science (HESS). The remainder of part IV is a reprint of:

Acuña Espinoza, E., Kratzert, F., Klotz, D., Gauch, M., Álvarez Chaves, M., Loritz, R., and Ehret, U. Technical note: An approach for handling multiple temporal frequencies with different input dimensions using a single LSTM cell. Hydrology and Earth System Sciences, 29(6), 1749–1758. https://doi.org/10.5194/hess-29-1749-2025*, 2025*

4 TECHNICAL NOTE: AN APPROACH FOR HANDLING MULTIPLE TEMPORAL FREQUENCIES WITH DIFFERENT INPUT DIMENSIONS USING A SINGLE LSTM CELL

ABSTRACT

Long short-term memory (LSTM) networks have demonstrated state-of-the-art performance for rainfall-runoff hydrological modelling. However, most studies focus on predictions at a daily scale, limiting the benefits of subdaily (e.g. hourly) predictions in applications like flood forecasting. Moreover, training an LSTM network exclusively on sub-daily data is computationally expensive and may lead to model learning difficulties due to the extended sequence lengths. In this study, we introduce a new architecture, multi-frequency LSTM (MF-LSTM), designed to use input of various temporal frequencies to produce sub-daily (e.g. hourly) predictions at a moderate computational cost. Building on two existing methods previously proposed by the co-authors of this study, MF-LSTM processes older inputs at coarser temporal resolutions than more recent ones. MFLSTM gives the possibility of handling different temporal frequencies, with different numbers of input dimensions, in a single LSTM cell, enhancing the generality and simplicity of use. Our experiments, conducted on 516 basins from the CAMELS-US dataset, demonstrate that MF-LSTM retains state-of-the-art performance while offering a simpler design. Moreover, the MF-LSTM architecture reported a 5 times reduction in processing time compared to models trained exclusively on hourly data.

4.1 INTRODUCTION

Data-driven methods, particularly long short-term memory (LSTM) networks (Hochreiter and Schmidhuber, 1997), have demonstrated state-of-the-art performance in rainfall-runoff hydrological modelling (Kratzert et al., 2019b; Lees et al., 2021; Loritz et al., 2024). Currently, most studies primarily focus on predictions at a daily scale. However, certain applications, such as flood forecasting, can benefit from subdaily scale predictions, especially in small fast-responding catchments. These higher temporal resolutions allow the model to better capture an event's magnitude and avoid artificial attenuation or dampening caused by daily aggregation. In addition, they allow the model to reproduce more accurately the temporal dynamics of the hydrograph and open up the possibility of capturing flash floods. For this reason, many operational flood forecasting services, including the National Water Prediction Service of the National Oceanic and Atmospheric Administration (NOAA) in the USA and the Flood Forecasting Centre of Baden-Württemberg (HVZ) in Germany, produce forecasts at a sub-daily resolution for their operational services.

One major drawback of running LSTM models at exclusively hourly resolution is the significant increase in computational cost for both model training and evaluation. For instance, studies using LSTM models at daily resolution typically employ a sequence length of 365 d for predictions (Klotz et al., 2022; Kratzert et al., 2019b; Lees et al., 2021; Loritz et al., 2024). By spanning a full year of data, this approach allows the LSTM model to capture long-term seasonal processes, such as snowmelt (Kratzert et al., 2019a). However, for hourly data, the equivalent sequence length increases to 365 x 24 = 8760 time steps, leading to a substantial increase in the computational resources required. Moreover, LSTM models have shown difficulties in learning information over long sequence lengths (Chien et al., 2021; Zhang and You, 2020), which would create a direct limitation when working exclusively with high-frequency data, such as hourly or 15 min resolutions.

A potential strategy for tackling this problem is to reduce the sequence length. However, this comes at the cost of excluding long-term processes. For example, if a sequence length of 365 time steps is maintained when working with hourly data, the look-back period would only cover 2 weeks, as opposed to a full year. Consequently, the model might not account for important long-term dynamics.

Another possible solution is the concept of ordinary differential equation LSTM (ODE-LSTM) models proposed by Lechner and Hasani (2020). The authors handle nonuniformly sampled data through the use of a continuous time state representation of recurrent neural networks. Gauch et al. (2021) carried out experiments exploring the potential of ODE-LSTM models in rainfall-runoff modelling. However, they indicated that this method achieved lower performance at a higher computational time than their proposed alternative.

Gauch et al. (2021) proposed the idea of processing older inputs at coarser temporal resolutions compared to more recent data. This approach is based on the fact that, for a dissipative system like a catchment, the importance of the temporal distribution of inputs diminishes the further back in time we look (Loritz et al., 2021). For instance, in cases where discharge during spring is driven by snowmelt, the exact hour in which snow accumulated 2 months earlier is unlikely to affect the hydrograph. Similarly, when modelling a storm, the basin's response will vary depending on the soil saturation. If the soil is saturated due to heavy rain over the past month, the precise timing of a peak in rainfall 3 weeks before becomes irrelevant. Thus, this approach to handling inputs at different temporal resolutions allows the model to capture long-term processes without the computational burden of processing all data at high frequency. In the following, we use a concrete example to both better illustrate the ideas proposed by Gauch et al. (2021) and make the connection with our method. For this, we will use 1 year of data to make a prediction, but only the most recent 2 weeks (14 x 24 = 336 time steps) will be processed at hourly resolution, while the rest will be processed at daily resolution. The number of time steps processed at each frequency is a model hyperparameter.

The first architecture proposed by Gauch et al. (2021), referred to as shared multi-timescale LSTM (sMTS-LSTM), begins with a forward pass at daily resolution (e.g. 365 time steps). The LSTM network's hidden and cell states from 2 weeks prior to the final time step are then retrieved, and the LSTM network is re-initialized with these states. Then, a second forward pass is performed using hourly data for the last 2 weeks. Moreover, since both daily (from the first forward pass) and hourly (from the second forward pass) predictions are available for the last 2 weeks, the authors proposed a regularization technique in which an extra term is incorporated into the loss function to induce consistency between the daily and hourly predictions.

One limitation of this architecture, highlighted by the authors, is that, because the same LSTM cell processes both daily and hourly data, the input at both timescales must include the same number of variables. As they mentioned, this can be problematic in operational settings, where different temporal resolutions often have different available variables. To address this, the authors proposed a more general architecture called multi-timescale LSTM (MTS-LSTM). In this variant, the hidden and cell states retrieved from 2 weeks prior are passed through a transfer function, and the result is used to initialize a second LSTM cell, which processes the hourly data. The advantage of this approach is that, with separate LSTM cells for each temporal frequency, different sets of input variables can be used at each resolution. We refer the reader to Fig. C.1 for a graphic visualization of these ideas.

Building on the work of Gauch et al. (2021), we propose a new methodology that combines the strengths of both models. We refer to it as multi-frequency LSTM (MF-LSTM). On the one hand, this new methodology retains the simplicity and elegance of the sMTS-LSTM model by using a single LSTM cell to process data at multiple temporal frequencies. On the other hand, we keep the ability of the MTS-LSTM model to handle different numbers of input variables at each frequency, which we accomplish through the use of embedding layers. Moreover, and as explained in detail in the following sections, we make predictions only at the highest frequency (e.g. hourly), and the remaining frequencies are recovered by aggregation, which guarantees cross-timescale consistency without the use of additional regularization.

The remainder of the paper is structured as follows. Section 4.2 details the MF-LSTM architecture and the experimental setup, including the datasets used and the benchmark comparisons. In Sect. 4.3, we present and analyse the results of these experiments. Finally, Sect. 4.4 summarizes the key findings and offers the study's conclusions.

4.2 DATA AND METHODS

4.2.1 *Data and benchmarking*

To ensure consistency with Gauch et al. (2021) and to enable a direct comparison, we followed their experimental setup. Keeping the same experimental setup allowed us to compare their results against our proposed method without having to rerun their experiments. The importance of driving model improvement through community benchmarks has been discussed previously in the machine learning and hydrological communities (Donoho, 2017; Klotz et al., 2022; Kratzert et al., 2024; Nearing et al., 2021).

The experiments were conducted in 516 basins located across the contiguous United States, all of which are part of the CAMELS-US dataset (Addor et al., 2017). From this dataset, we extracted 26 static attributes (see Table C.3). The hourly input data (see Table C.1) were extracted from North American Land Data Assimilation System (NLDAS-2) hourly products (Xia et al., 2012), while the target discharge data were retrieved from the USGS Water Information System (US Geological Service, 2016). Following standard machine learning practices, the data were divided into three subsets. The training period was from 1 October 1990 to 30 September 2003, the validation period was from 1 October 2003 to 30 September 2008, and the testing period was from 1 October 2008 to 30 September 2018.

4.2.2 *MF-LSTM*

The concept of MF-LSTM comes from the principle that an LSTM cell has no inherent limitation in processing data at different temporal frequencies. In contrast to process-based hydrological models, where one would not update a storage (say interflow) with a 5mmh^{-1} flux (say evapotranspiration) in one time step and then with an 8mmd^{-1} flux in another, an LSTM cell can accommodate an equivalent updating scheme. To process an input, an LSTM cell always processes a sequence one step at a time. However, there is no explicit assumption about the progress of time within one such step. Due to its time-dependent gating mechanisms, the LSTM cell can learn to modulate how the cell states are updated, regardless of the temporal resolution of the inputs. Consequently, we can leverage this property to handle multiple temporal frequencies within a single LSTM cell, processing older inputs at coarser resolutions and more recent data at higher resolutions.

A concrete example of this approach is illustrated in Fig. 4.1a using daily and hourly frequencies. In this example, our goal is to predict hourly discharge. To capture long-term processes, we initially input a full year of high-resolution hourly data (e.g. 365 x 24 = 8760 hourly time steps). To avoid the computational burden and learning difficulties associated with processing long sequences, the first n time steps were processed at a coarser resolution, reducing the length of the input sequence entering the LSTM cell. The number of time steps processed at each resolution is a model hyperparameter and can be determined through hyperparameter tuning.

The example in Fig. 4.1a shows the case where the first n = 351 x 24 = 8424 time steps were aggregated into 351 blocks, each containing an average of 24 hourly measurements. Given the normalization of the input and target data and the non-mass-conservative structure of the LSTM model, both the average and the sum of the hourly measurements can be used. The remaining m = 14 x 24 = 336 time steps were then processed at the original hourly resolution. By applying this method, we reduced the original sequence length from 8760 to 351+336 = 687 time steps, decreasing the amount of data to be processed by a factor of 12.8.

To inform the LSTM cell about the frequency it should be operating at, we added a flag channel. This has a value of zero for the first 351 time steps and a value of one for the remaining 336. Adding a flag to help the model distinguish between different types of conditions is a common practice in machine learning, as it provides the model with additional context (Nearing et al., 2022). For the LSTM cell specifically, the flag channel acts as an additional bias that further regulates the gating mechanisms. Figure 4.1b shows the inclusion of the flag channel for the different frequencies.

Note that, in the previous paragraph, we used pre-defined values to simplify the explanation and clarify the concept. However, the method is by no means restricted to this setup, and its flexibility allows it to adjust the number of time steps processed at each resolution and the order in which the different frequencies are applied. Additionally, the composition of the time series can be alternated from batch to batch during training or inference. Moreover, the method is not restricted to using only two frequencies and, as we show in the next section, a weekly–daily–hourly frequency scheme can be handled without any additional burden.

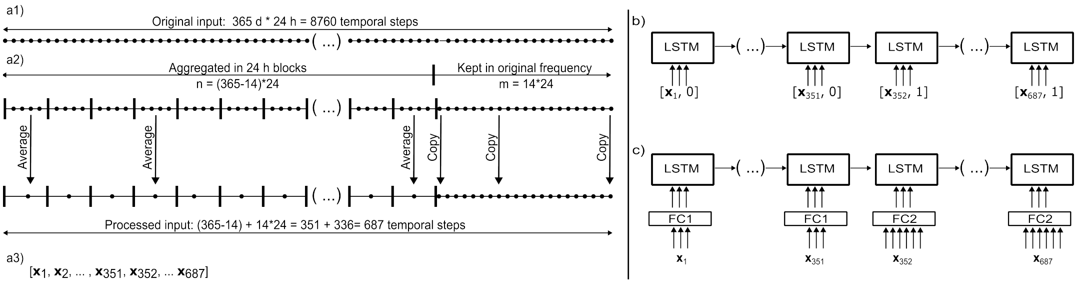

Figure 4.1: Data-handling structure for MF-LSTM. **(a1)** The original sequence length consists of 1 year of hourly data: 8760 temporal steps. **(a2)** The first (365 − 14) x 24 = 351 x 24 = 8424 time steps are aggregated into 351 blocks, while the remaining 14 x 24 = 336 time steps are processed at their original hourly frequency. **(a3)** The final input series that will be processed by the LSTM cell consists of 351 + 336 = 687 time steps. **(b)** In the case where the same number of inputs for each frequency are available, we add a flag as an additional channel to help the LSTM cell to identify each frequency. **(c)** In the case where different numbers of inputs for each frequency are available, a fully connected (FC) linear layer can be used to map the variable number of inputs of each frequency to a pre-defined number of channels.

One of the main advantages stated by Gauch et al. (2021) about the MTS-LSTM architecture is its ability to handle a variable number of inputs for each frequency, because different LSTM cells are used for each temporal frequency (see Fig. C.1). We propose the use of embedding networks as an alternative solution. By using one embedding network for each temporal frequency, we can map different numbers of inputs to a shared dimension. This strategy allows us to separate the steps of our pipeline. We use the LSTM cell for sequence processing only, and we use the embedding networks to prepare the original information in the format or type that the LSTM cell requires. In the simplest case, the embedding networks could even be a single linear layer, as will be used for the rest of this paper (see Fig. 4.1c). We evaluated the embedding network with and without the flag channel and observed comparable performance in both cases. This result indicates that the embedding network can internally identify frequency information without the need for an additional flag channel. Therefore, we opted for the simpler approach and excluded the extra channel.

In summary, the main distinction between MF-LSTM and sMTS-LSTM lies in MF-LSTM's ability to handle a different number of inputs for each temporal frequency, which gives an advantage in operational settings where different temporal resolutions often have different available variables. Moreover, the primary difference between MF-LSTM and MTS-LSTM lies in the simpler architecture of the former, which uses a single LSTM cell, in contrast to one cell per frequency. This results in a more parsimonious model that aligns closely in structure and usage with traditional single frequency LSTM models.

4.3 RESULTS AND DISCUSSION

4.3.1 *Performance comparison*

Our long-term goal, which goes beyond the scope of this study, is to implement an operational hourly hydrological forecasting system using machine learning methods. The MF-LSTM method is a step towards achieving this, as it enables computationally efficient simulation of hourly discharges while allowing us to handle a variable number of inputs at each temporal resolution – both requirements for our broader objective. Consequently, the results reported in this section will focus on two aspects: the ability of MF-LSTM to produce hourly discharge and the ability to handle a variable number of inputs while doing so.

Gauch et al. (2021) presented two experimental setups that address these aspects. Therefore, we ran these experiments as a benchmark to evaluate the performance of our method against their results. Both experiments evaluate the case in which one is interested in simulating hourly discharges, and they do so by processing part of the information at daily frequency and part of it at hourly frequency. More specifically, data for 1 year are processed, but only the last 14 d (336 h) are processed at hourly frequency. The value of 336 h was identified in the original study by hyperparameter tuning. In all of the cases, the results are reported using an ensemble of 10 independent LSTM models that were initialized using different random seeds. The final simulation value is taken as the median streamflow across the 10 models for each time step.

The first experiment evaluated the scenario where the same number of inputs (see Table C.1) was used for both daily and hourly processing. In this case, we can directly compare our model's performance with the results reported by Gauch et al. (2021) for MTS-LSTM and sMTS-LSTM, together with what they refer to as the naive approach. The naive approach involves running a standard LSTM model exclusively on hourly data with a sequence length of 4320 h (6 months). We can see from Fig. 4.2a that all the models present the same performance up to the second decimal place, with a median Nash–Sutcliffe efficiency (NSE) of 0.75.

The second experiment evaluated the scenario in which different numbers of inputs were used for the daily and hourly steps (see Table C.2). Specifically, the daily frequency

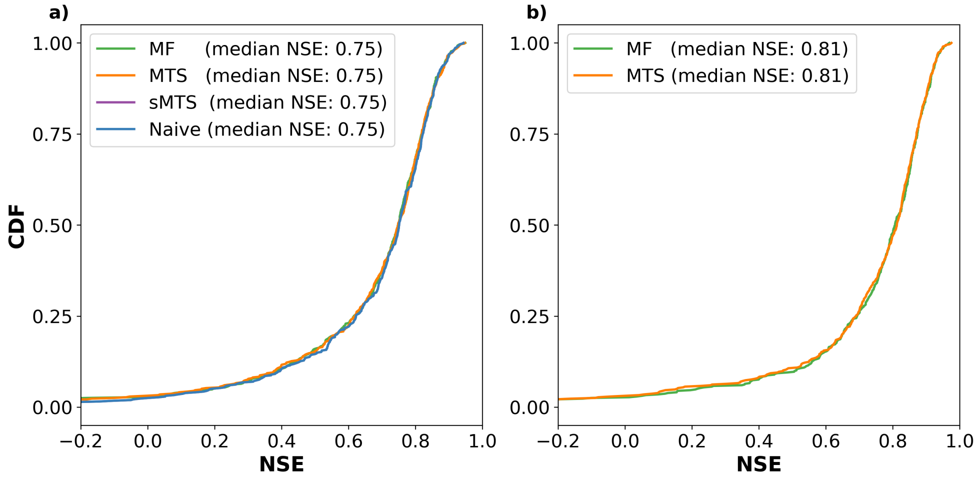

Figure 4.2: Cumulative NSE distribution of the different models, evaluating the prediction accuracy for hourly discharges along 516 basins in the USA. **(a)** Case where the same number of variables is used during daily and hourly processing (see Table C.1). **(b)** Case where different numbers of variables are used during daily and hourly processing (see Table C.2).

incorporated 10 dynamic variables from the Daymet and Maurer forcing datasets, while the hourly steps included 21 variables. Eleven of these variables were sourced from the NLDAS-2 forcing at hourly resolution, and the remaining 10 were a low-resolution re-discretization of the 10 daily variables into an hourly frequency (i.e. the daily value was repeated 24 times). Consistent with Gauch et al. (2021), since the Maurer forcings go until 2008, the results of this experiment are reported for the validation period. As shown in Fig. 4.2b, MF-LSTM achieves a performance comparable to that of MTS-LSTM, with both reporting a median NSE of 0.81. Comparisons with the sMTS and naive models are not possible for this experiment, as previously explained, because these models cannot accommodate different numbers of variables for different frequencies.

The previous experiments showed that the MF-LSTM architecture can achieve a state-of-the-art performance that is fully comparable with the MTS-LSTM and sMTS-LSTM architectures. The results show that a single LSTM cell can handle multiple temporal frequencies at the same time. Moreover, the second experiment confirms that a simple fully connected linear layer can successfully encode different numbers of input variables in a pre-defined number of channels.

We also ran an additional experiment to evaluate the capacity of the model to handle more than two frequencies. Specifically, we used a weekly–daily–hourly scheme. The first half of the year (182 d) was handled using a weekly aggregation. The next 5.5 months (169 d) were at daily resolution, and the remaining 14 d used an hourly frequency. Our results showed that MF-LSTM is capable of handling this case, presenting a similar performance (see Fig. C.2) and reducing the sequence length from 687 to 531.

4.3.2 *Computational efficiency*

As shown in Fig. 4.1a, one key advantage of processing older inputs at coarser temporal resolution than more recent ones is the reduced computational cost, particularly when compared with feeding in the whole sequence length at a finer resolution (e.g. hourly).

This reduction in computational cost impacts not only the training time, but also the memory usage. With long sequence lengths, one might run out of GPU memory or be forced to use alternative strategies such as reducing the batch size during training and evaluation.

However, the total training time is influenced by external factors, such as differences in hidden size or batch size, which are not directly related to the methods themselves. To minimize these external effects, we conducted an additional experiment where we standardized the hidden size and batch size across all the models and compared the average time needed to process a batch. The results showed that MF-LSTM, MTS-LSTM, and sMTS-LSTM exhibited nearly identical efficiency, while the naive approach was approximately 5 times slower. For reference, training MF-LSTM on a Tesla V100 GPU took around 7 h.

4.4 CONCLUSIONS

In this study, we introduced the MF-LSTM architecture, designed to produce sub-daily (e.g. hourly) predictions at a moderate computational cost while giving the model access to long sequences of input data. Building on Gauch et al. (2021), our method processes the input's temporal sequence using different aggregations. Hence, it accounts for the fact that the effect of the input's temporal distribution diminishes the further we look back in time. This allows MF-LSTM to predict hourly discharges without the overhead of handling the entire sequence at a fine temporal scale.

The ability of the LSTM model to maintain performance while handling data from the past at lower resolutions highlights how the LSTM cell acts similarly to a process-based hydrological model, with dissipative behaviour when it comes to the memory of past forcings. This is a step towards understanding LSTM-based predictions better as they are gaining popularity for applications in hydrology.

As high-resolution data become increasingly available in the environmental sciences, traditional LSTM models will continue to face challenges when trying to learn from these long sequence lengths. The approach we present here, with its simplicity and computational efficiency, offers a practical solution. Areas like weather forecasting, where data at resolutions of minutes are not uncommon, might benefit from this type of model. Moreover, the possibility of combining multiple frequencies, like our weekly–daily–hourly scheme, enables modellers to extend look-back periods. This may also be beneficial in other domains such as groundwater, where long-term historical data are required to capture slow dynamic processes.

Our proposed embedding strategy opens up the possibility of mapping different numbers of inputs to a shared dimension. This flexibility not only simplifies the model architecture by allowing a single LSTM cell to handle diverse input configurations but also enhances the model's adaptability in operational settings, where the availability of input data may vary across timescales. This overcomes the limitation previously stated in sMTS-LSTM.

Furthermore, we demonstrate that a single LSTM cell can effectively manage processes operating at different timescales (eliminating the need for separate LSTM cells for each timescale) and transfer functions between their hidden states. This results in a more parsimonious model that aligns more closely in structure and usage with traditional single-frequency LSTM models, making the transition from single-frequency to multi-frequency LSTM models more intuitive for users.

Through experiments on 516 basins from the CAMELS-US dataset, the MF-LSTM model demonstrated the same performance as the MTS-LSTM and sMTS-LSTM models, indicating that the added simplicity and generality do not come at the expense of predictive capability. Moreover, the new architecture presents a similar computational cost to the two previous options and reduces the training time by a factor of 5 when compared to the naive approach.

The fact that a single LSTM cell allows us to handle multiple frequencies could be due to the close similarities between processes at different timescales (e.g. daily and hourly). The LSTM architecture takes advantage of these similarities, along with its ability to regulate gates based on the current context, enabling it to effectively process multiple frequencies. By using a single LSTM cell, we can leverage the additional information content encoded in our data.

The hyperparameters of the model were adopted from Gauch et al. (2021), who conducted hyperparameter tuning. We acknowledge that transferring these parameters across different architectures may not guarantee optimal model performance. However, the primary objective of this technical note is to introduce the new architecture. Furthermore, we demonstrate that, even with the given hyperparameters, the proposed model achieves a performance comparable to the current state of the art.

Part V

CONCLUSIONS

5 CONCLUSIONS

5.1 SUMMARY AND CONCLUSIONS

To improve the ability to answer the question: How much water will be in the river tomorrow? this thesis explored and developed techniques to enhance hydrological rainfall-runoff simulations using machine learning methods. Chapters 2 and 3 focused on hybrid modeling, combining data-driven techniques with conceptual hydrological models to assess their performance, interpretability and generalization capabilities. Chapter 4 addressed the challenges faced by data-driven methods when handling long sequences of high-resolution data, and introduced the MF-LSTM architecture, which enables handling multi-frequency data within a single LSTM cell, improving computational efficiency for hourly discharge simulations.

In chapters 2 and 3, I constructed hybrid models by integrating a conceptual head layer with an LSTM model. The experiments demonstrated that hybrid models can achieve state-of-the-art performance, comparable to LSTMs and surpassing stand-alone conceptual models. Nonetheless, one should be careful in interpreting the role of the conceptual head layer. Findings in chapter 2 revealed that, given enough flexibility, the LSTM can compensate for structural deficiencies in the conceptual component. As a result, hybrid models can maintain high predictive performance even when the conceptual model is inadequate. This also implies that evaluating conceptual models solely based on performance metrics, within a hybrid framework, can be misleading, as the LSTM can compensate for deficiencies in the underlying conceptual structure.

Chapter 2 also demonstrated that incorporating a conceptual head layer provides access to unobserved variables, as evidenced by the model's ability to capture soil moisture dynamics without direct soil moisture inputs during training. This access, along with the predefined associations between buckets, fluxes, and parameters with specific states and processes, offers a degree of interpretability. However, when relying on simplified conceptual models with basin-averaged characteristics, this interpretability is primarily associative rather than grounded in strict physical principles. Therefore, its practical usefulness should not be overstated.

Chapter 3 explored the generalization capabilities of hybrid, LSTM, and conceptual models in predicting extreme hydrological events. While all models exhibited similar performance in most cases, certain patterns emerged. On the one hand, LSTMs performed better in arid basins, where the conceptual model struggled with runoff generation assumptions. On the other hand, for the most extreme events, the hybrid model produced higher peak discharge estimates, while LSTM predictions were limited by their saturation threshold.

The training/testing split used in chapter 3 was designed to assess the generalization capabilities of the models, by exposing them to out-of-sample conditions, different from those seen during training. However, this setup also artificially amplified the LSTM's saturation issue by restricting the range of discharges it was trained on. In practical applications, this limitation can be mitigated by incorporating and emphasizing low-probability extreme events during training and leveraging extensive datasets to broaden the model's learned distribution.

Chapters 2 and 3 showed that hybrid models represent a viable approach, capable of achieving state-of-the-art performance. However, the results also indicate that stand-alone LSTM networks are stronger candidates for practical hydrological forecasting applications. While the additional interpretability provided by hybrid models is conceptually appealing, its practical utility remains debatable, especially if one is interested exclusively in a good predicting tool that maps inputs to outputs. Moreover, in most extrapolation scenarios, the predictive performance of stand-alone LSTMs and hybrid models was comparable, and alternative techniques can be employed to mitigate the saturation issue in LSTMs. Furthermore, purely data-driven approaches offer greater flexibility, allowing the integration of multiple data sources and variable types into the forecasting pipeline, without the constraints of modifying specific components of a process-based model. This adaptability extends to incorporating the target variable when available, facilitating a smooth transition between hindcast (past) and forecast (future) periods, while eliminating the need for additional data assimilation modules. During the hindcast period, the model can leverage both meteorological observations and past observed discharge, whereas the forecast period can rely exclusively on meteorological predictions. Given these advantages, the fourth chapter of this thesis addressed a current challenge faced by data-driven models: the computational cost associated with handling high resolution data.

Chapter 4 introduced the MF-LSTM architecture, designed to efficiently handle high-resolution data across multiple timescales while maintaining predictive accuracy. This method processes temporal sequences at different resolutions, based on the principle that the effect of the input's temporal distribution diminishes the further one looks back in time. By doing so, MF-LSTM generates hourly discharge predictions without the computational burden of processing the entire sequence at the finest temporal scale.

A key advantage of MF-LSTM is its ability to handle multiple timescales within a single LSTM cell, eliminating the need for separate cells and transfer functions. This results in a more parsimonious model. Furthermore, the model structure aligns closely with traditional single-frequency LSTMs, making the transition to multi-frequency modeling more intuitive for users. Additionally, its ability to embed different numbers of input variables at each time resolution into a shared latent space enhances adaptability in operational settings.

Overall, the objectives of the thesis were addressed. The integration of machine learning techniques into hydrological modeling was explored, assessing the potential benefits of hybrid models while advancing purely data-driven approaches. The interpretability and generalization capabilities of hybrid models were examined, and a new multi-frequency LSTM architecture was proposed. These contributions provide valuable insight to improve the answer to the question: how much water will be in the river tomorrow?.

5.2 OUTLOOK

The approaches explored in this thesis adopt a lumped framework, treating entire catchments as single entities with averaged properties. While effective, this methodology does not take full advantage of the now available spatial data, including high-resolution meteorological and geographical datasets. Future research should focus on incorporating spatially distributed information into hydrological models. Integrating spatial information within data-driven models in an end-to-end trainable pipeline has the potential to enhance predictive performance, particularly in large basins where lumped representations may be too coarse, or in flash-flood forecasting, where the precise location of a storm within the basin significantly influences runoff response. Additionally, incorporating river network structure could further improve model accuracy by capturing flow connectivity and spatial dependencies within catchments.

Part VI

APPENDIX

A APPENDIX TO CHAPTER II

A.1 SHM MODEL

In this section, we offer a concise overview of the conceptual model's key features. For a fully detailed explanation, we refer to Ehret et al. (2020). In a slight variation from the original paper, we included a snow module, and the potential evapotranspiration is read directly from the CAMELS-GB dataset.

Figure A.1 illustrates the overall structure of the model. The model input consists of three forcing variables: Precipitation (P) $[mm \cdot d^{-1}]$, temperature (T) $[^{\circ}C]$, and potential evapotranspiration (ETp) $[mm \cdot d^{-1}]$. These three quantities were read directly from the CAMELS-GB dataset (Coxon et al., 2020a). To emulate the hydrological processes occurring in the basin, the model uses five storage components, namely, snow module, unsaturated zone, fast flow, interflow, and baseflow. Overall, to regulate the fluxes between components, eight parameters need to be calibrated: *dd*, *f_thr*, *su_max*, β, *perc*, *kf*, *ki* and *kb* (see units in Table A.1).

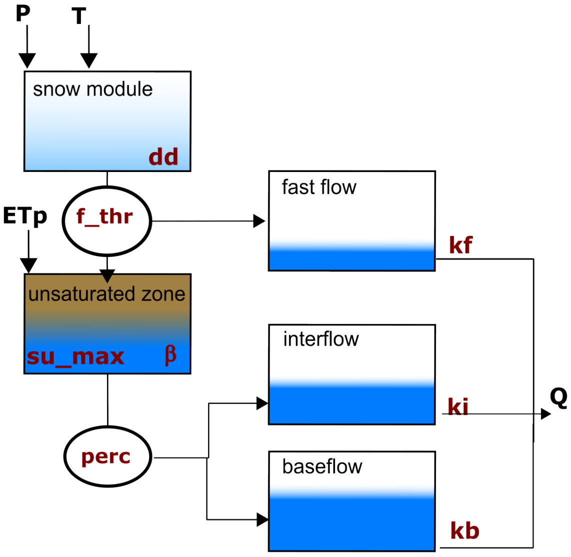

Figure A.1: Structure of SHM hydrological model used for rainfall-runoff prediction

The snow module receives P and T as inputs. Based on the temperature, precipitation is either stored as snow or moves forward together with additional discharge from snowmelt (if any). Snowmelt is calculated using the degree-day method in which the parameter *dd* relates to the volume of snowmelt at a given temperature. If the outflow of the snow module exceeds a threshold (*f_thr*), the excess is directed to the fastflow reservoir, while the remaining portion enters the unsaturated zone bucket. On the other hand, if the snow storage outflow is smaller than *f_thr*, all water enters the unsaturated zone as input. Within the unsaturated zone, several processes occur. First, evapotranspiration causes water loss. The potential evapotranspiration (ETp) is provided as a forcing variable but is adjusted to reflect the actual evapotranspiration considering water availability. Additionally, there is an outflow from the unsaturated zone, determined by a power relationship involving the parameters *su_max* and β. This outflow is then divided by the *perc* parameter, allocating portions to the inflows of the interflow and baseflow storages. Finally, the total discharge of the basin is computed as the sum of the outflows from the fast-flow, interflow, and baseflow storages. Each outflow is a linear function of its corresponding storage and the recession parameters *kf*, *ki*, and *kb*, respectively.

A.2 TRAINING PROCESS COMPARISON BETWEEN HYBRID AND LSTM MODELS

While the coupling of the data-driven and conceptual models may appear straightforward from a general perspective, it is important to highlight several details. First, as one can notice from Fig. 2.1 in the main paper, the forcing variables (P, T , ET) are used as inputs for both the LSTM and the SHM. The forcing variables and static attributes used as inputs for the LSTM are standardized using the method described in the previous section. However, due to the mass-conservative structure of the SHM, their input variables (P, T , ET) are used in the original scale. Second, it is important to consider that the SHM parameters have certain feasible ranges. While the LSTM could theoretically learn these ranges, the optimization process becomes highly challenging due to the immense search space involved. We found that without constraining the parameter ranges, the LSTM was not able to identify parameters that yield a functional hydrological model. Hence, we predefined ranges within which the parameters can vary (see Table A.1). These ranges were defined considering the findings in Beck et al. (2020) and Beck et al. (2016), which provide valuable insights into the appropriate parameter values of conceptual models. By defining these ranges, we not only reduced the computational costs of the optimization but also ensured consistency with the methodology employed by Feng et al. (2022).

To map the output of the LSTM network to the predefined ranges, the j outputs ($j = 8$, one per parameter) are passed through a sigmoid layer to transform the values to a [0, 1] interval. Then the transformed values are mapped to the predefined ranges through a min–max transformation, as exemplified in Eq. (A.1):

$$\theta_j = x_j^{min} + \text{sigmoid}(o_j) \cdot (x_j^{max} - x_j^{min}), \tag{A.1}$$

where θ_j is each of the values passed as parameters to the SHM, o_j are the original outputs of the LSTM network, and x_j^{min} and x_j^{max} correspond to the minimum and maximum values of the predefined ranges (see Table A.1) in which each parameter can vary, respectively.

Lastly, there is a difference in how we trained our LSTM and hybrid model. The first one was trained using a seq2one approach, while the second one used a seq2seq methodology. Furthermore, even though both models used a spin-up period (e.g., sequence length), the spin-up period of the hybrid model also considered a time to stabilize the internal states of the conceptual model.

Table A.1: Search range for SHM parameters during hybrid model optimization

Parameter	Minimum value (x_j^{min})	Maximum value (x_j^{max})	Unit
dd	0.0	10.0	$mm°C^{-1}d^{-1}$
f_thr	10.0	60.0	mm
su_max	20.0	700.0	mm
β	1.0	6.0	−
$perc$	0.0	1.0	%
k_f	1.0	20.0	d
k_i	1.0	100.0	d
k_b	10.0	1000.0	d

mm : millimeters, °C : degree celsius, d : days

To facilitate the understanding of the previous concepts, let us create an example. Let us assume that both the LSTM and the hybrid model were trained on 60 basins and 10 years ($\sim$ 3652 d) of data, even though we know from before that this was not the case.

For training the LSTM, we use a batch size of $N = 256$ and a seq2one method. Therefore, to construct each batch we randomly select, without replacement, 256 data points from the total pool of $3652 \cdot 60 = 219\ 120$ training points. For each of the 256 points of our batch, we run our LSTM for a given sequence length (e.g., 365 time steps) and extract the last simulated value. We then calculate our loss function with a metric that quantifies the difference between the observed and simulated values of the 256 training points and backpropagate this loss to update the network weights and biases. To complete an epoch, we iterate through the $\frac{219120}{256} = 855$ batches.

In the case of the hybrid model, we train it as a seq2seq approach; therefore, to calculate the loss function we do not just use the last element of our simulated sequence but a part of that sequence (e.g., 365 time steps). Therefore, for the same scenario of 60 basins and 10 years of data, we have $\frac{60 \cdot 10 \cdot 365}{365} = 600$ training vectors, each with 365 elements that are used to calculate the loss. If we use a batch size of $N = 8$, each batch will contain 8 randomly selected vectors with 365 sequential training points each, so the loss function will quantify the difference of $365 \cdot 8 = 2920$ simulated and observed values. To complete an epoch, we iterate through $\frac{600}{8} = 75$ batches.

The other small difference while training the models is the length of the spin-up period. For the LSTM, we use a sequence length of 365 d, which means that we run a sequence of 365 d and extract the last value of this sequence to calculate the loss. The first 364 d help us consider the historical information to make a good prediction and to avoid bias due to the initialization of the cell states (usually zero). In our hybrid model, the LSTM uses a sequence length of 180 d, which means that only after 180 d do we start to retrieve the parameters that go into the conceptual model. The purpose of these 180 d is the same as before, consider the historical information to make a context-informed parameter estimation and avoid the bias due to the initialization of the cell states. However, we also need a warm-up period to avoid a biased result due to the initialization of the different storages of the conceptual part. Therefore, for each instance of the batch, we ran our conceptual model for a 2-year period. The initial year serves solely as a warm-up period (excluded from the loss function), while the second year's data are utilized for actual training. Figure A.2 illustrates the data handling while training both models.

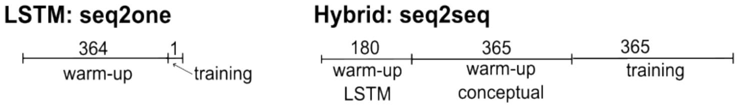

Figure A.2: Training scheme comparison for LSTM and hybrid model. The former one uses a seq2one approach. The latter uses a seq2seq approach and the total spin-up period consists of a sequence length for the data-driven part plus a warm-up period for the states of the conceptual part.

APPENDIX TO CHAPTER III

B

B.1 BENCHMARKING HYBRID MODEL

For our hybrid model we used the $\delta_n(\gamma^t, \beta^t)$ architecture proposed by Feng et al. (2022). A scheme of this architecture is shown in Fig. B.1. Given that our experiment pipeline was executed in the NeuralHydrology package, we first had to benchmark our model implementation against the original case. Figure B.2 shows that our model implementation produced similar results to the one reported by Feng et al. (2022).

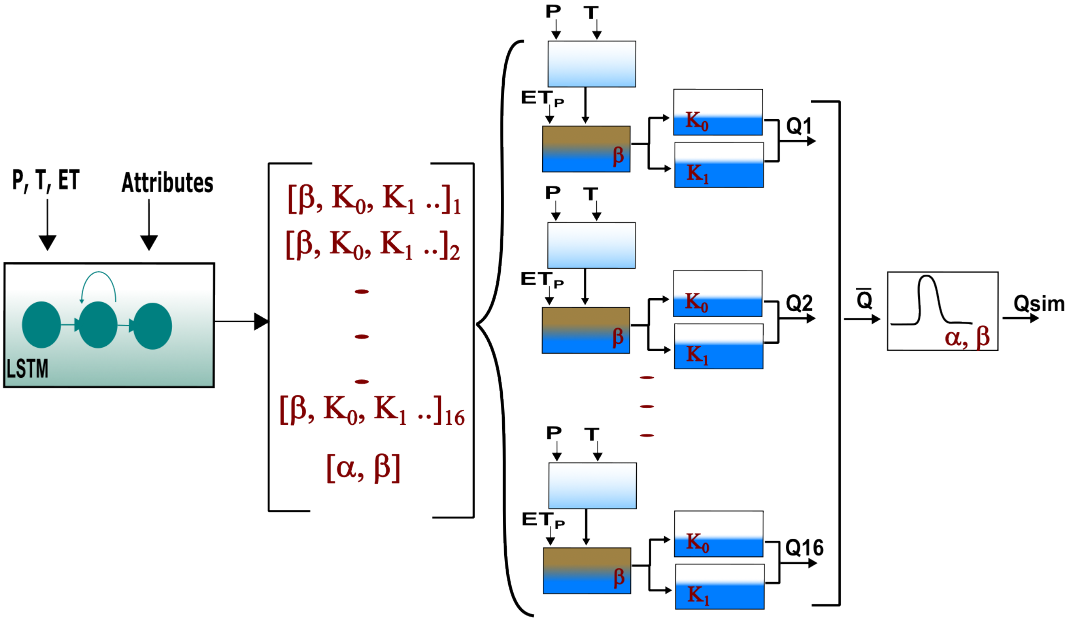

Figure B.1: Scheme of the hybrid model structure. The LSTM predicts the parameters used to parameterize the different hydrological modules. In total, the LSTM produced 210 parameters (16 HBV members each with 13 parameters, and 2 routing parameters). The 16 hydrological models are run in parallel. The produced discharges are averaged, and the averaged signal is further routed using a unit hydrograph based on the gamma function. The output of the routing module is retrieved as the final simulated discharge.

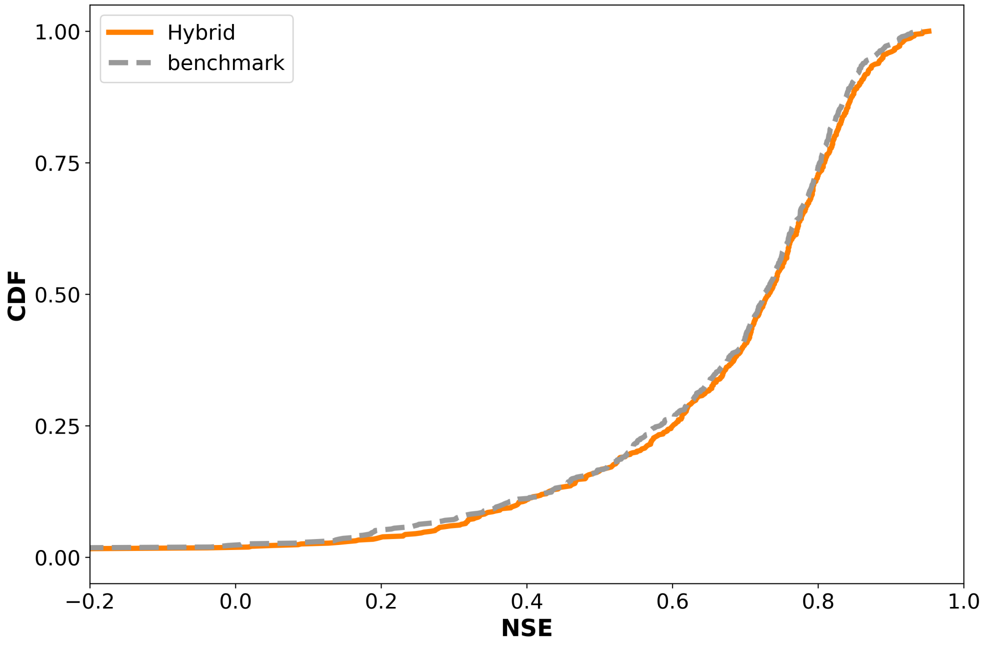

Figure B.2: Cumulative density function (CDF) of the Nash–Sutcliffe efficiency (NSE) for different models, generated using 671 basins of the CAMELS-US dataset.

B.2 Impact of random model initialization on absolute percentage error metric

Figure B.3 shows the effect of different model initialization on the APE metric. The ranking of the models in the last three categories (25-50 years, 50-100 years, 100+ years) varies depending on the model initialization. This indicates that the differences in the median values are within the statistical noise, and we cannot conclude that one model is better than the other.

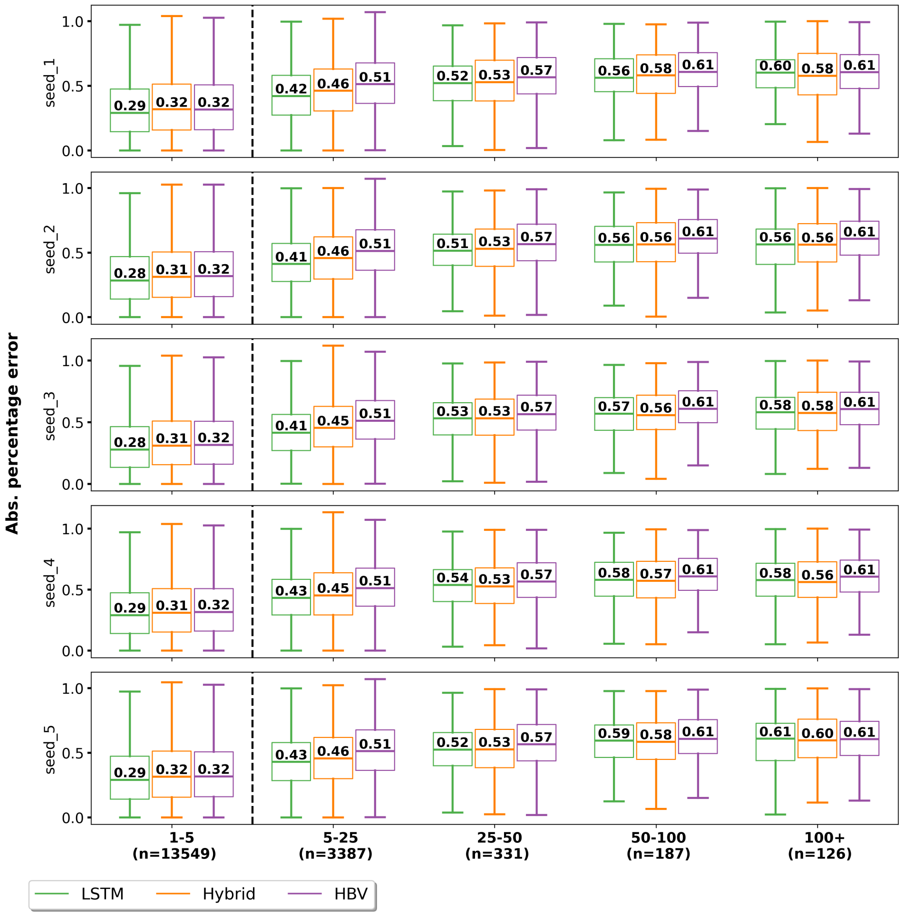

Figure B.3: Variation in absolute percentage error (APE) due to random initialization of the LSTM and hybrid models.

B.3 SPATIAL VISUALIZATION AND COMPARISON OF MODEL PERFORMANCE

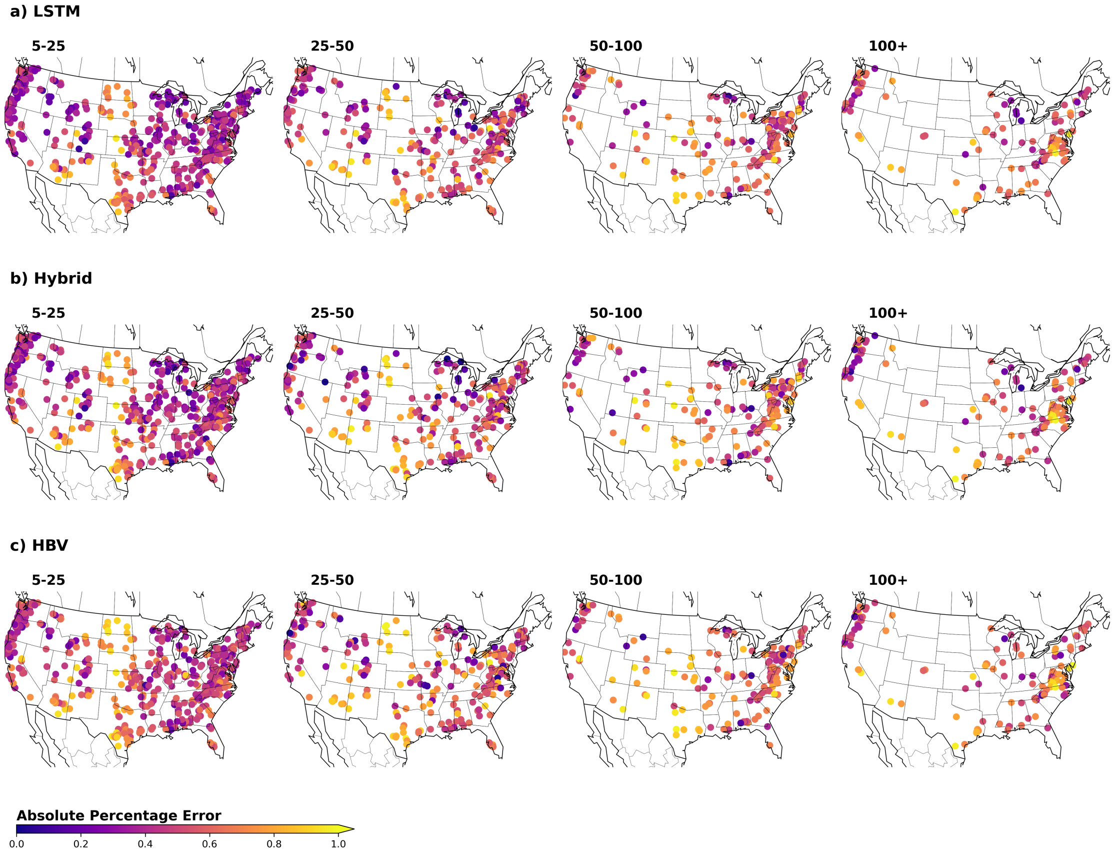

Figure B.4: Spatial visualization of absolute percentage error (APE) for the different models and the different return periods. Each point is associated with one basin. The scale indicates the median APE between observed and simulated values for all the events associated with the respective basin and return period.

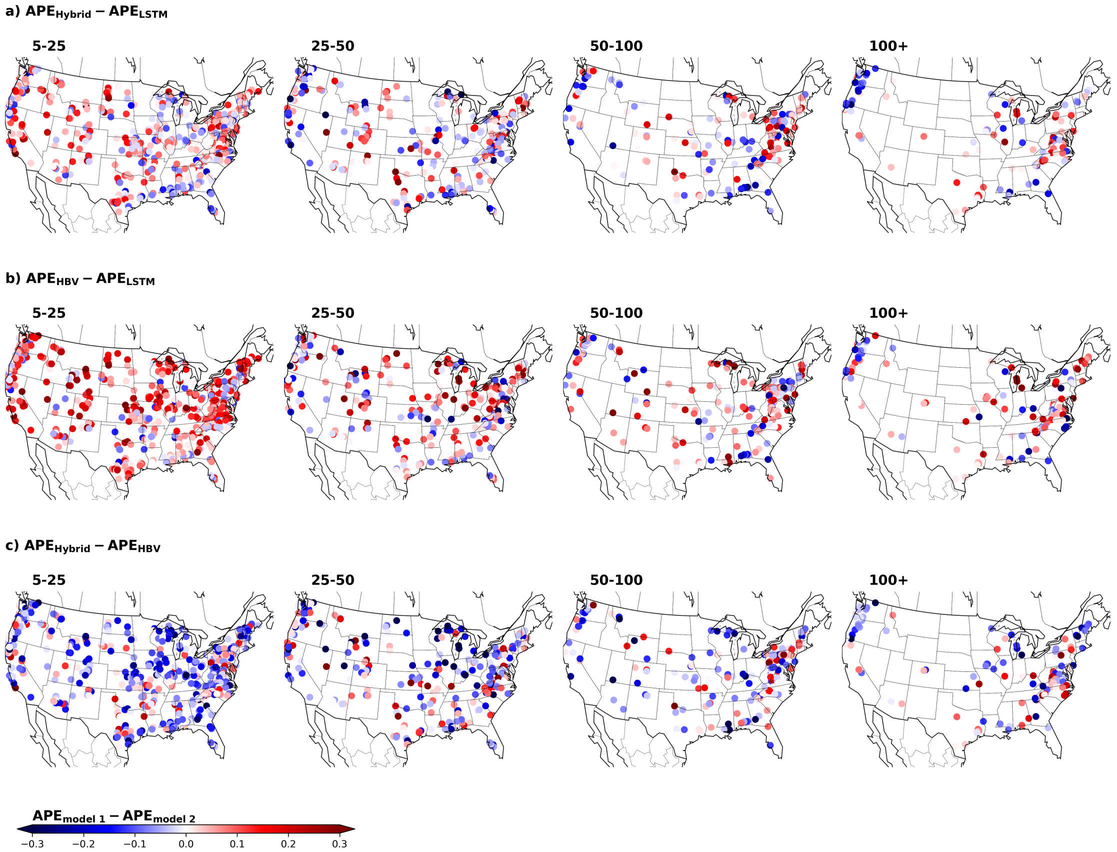
a) $\mathrm{APE_{Hybrid} - APE_{LSTM}}$
5-25
25-50
50-100
100+
b) $\mathrm{APE_{HBV} - APE_{LSTM}}$
5-25
25-50
50-100
100+
c) $\mathrm{APE_{Hybrid} - APE_{HBV}}$
5-25
25-50
50-100
100+
$\mathrm{APE_{model\ 1} - APE_{model\ 2}}$
−0.3
−0.2
−0.1
0.0
0.1
0.2
0.3

Figure B.5: Spatial visualization of the difference between the absolute percentage error (APE) for the different models. Each point is associated with one basin. The scale indicates the difference between the median APE of the different models. The difference is calculated based on the order in which the models are named: $APE_{\text{model 1}} - APE_{\text{model 2}}$; therefore, negative (blue) values indicate that model 1 performed better while positive (red) values indicate that model 2 performed better.

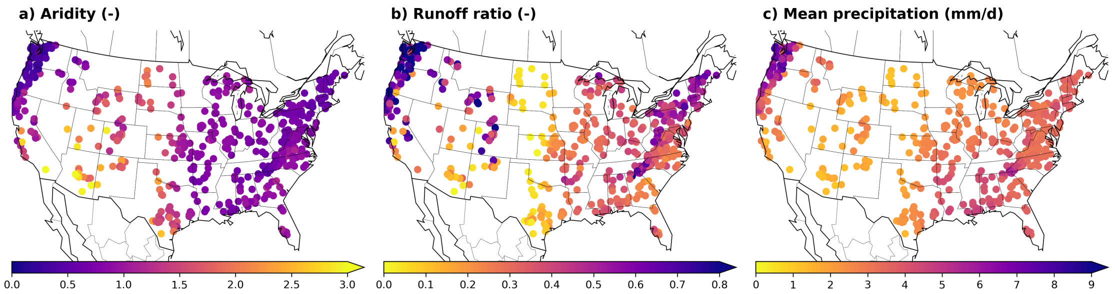

Figure B.6: Spatial visualization of basin-averaged static attributes. Each point is associated with one basin. a) Spatial variability of aridity b) Spatial variability of runoff ratio c) Spatial variability of mean daily precipitation.

B.4 TEMPORAL VARIATION OF DYNAMIC PARAMETERS

Figure B.7 shows the temporal variation in the dynamic parameters of the hybrid model for three basins. For the first two basins, we can see clear cyclic patterns, in which the parameters are adjusted in dry or wet seasons to produce less or more water. These patterns were discussed in Acuña Espinoza et al. (2024b) for experiments conducted in CAMELS-GB and suggest the possibility that the LSTM controls the HBV models in a consistent way. However, further investigation is needed to understand the LSTM–HBV interaction. In the third basin, the hydrograph does not present such a clear distinction between dry and wet seasons, and we can observe more variation in the parameters.

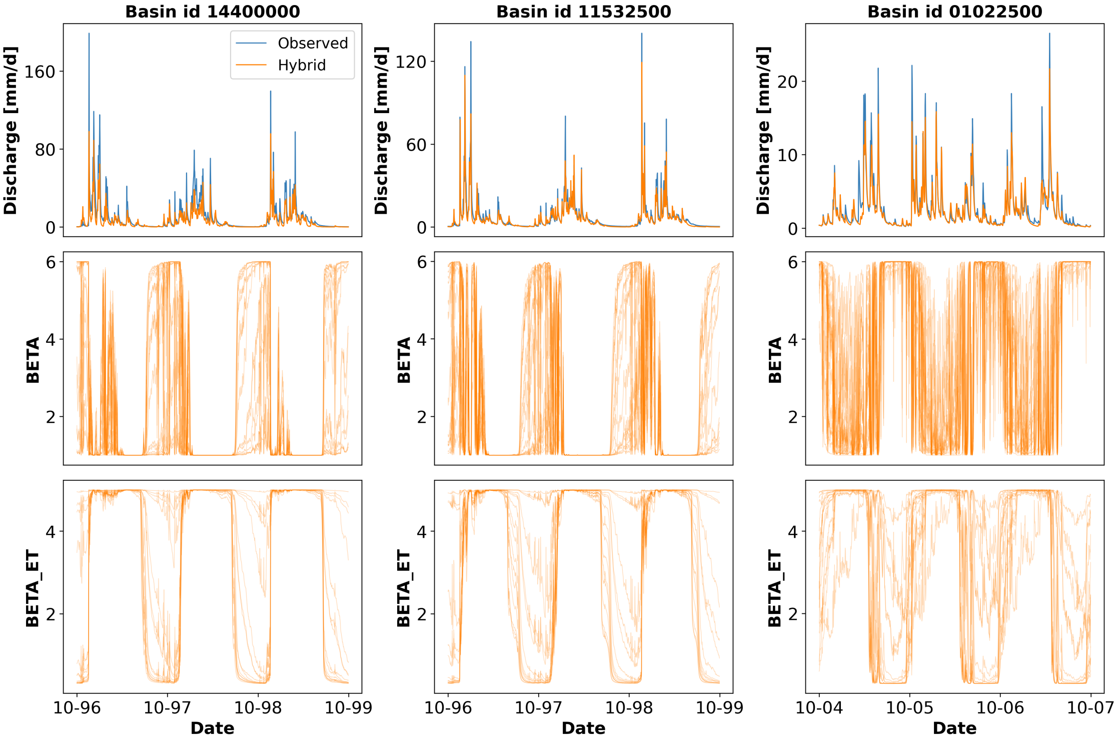

Figure B.7: Time series indicating the variation in parameters for three different basins and their association with the simulated hydrographs.

C APPENDIX TO CHAPTER IV

C.1 ADDITIONAL INFORMATION OF EXPERIMENTAL DESIGN

The following tables present the variables used in the experiments associated with this study. Tables C.1 and C.2 present the variables used in the first and second experiments. The third and fourth columns of each table indicate whether the variable was used at daily frequency, hourly frequency, or both. Table C.3 shows the 26 static attributes used as additional inputs in the models.

Table C.1: Dynamic input variables used in the first experiment, where the same number of variables is used for the daily and hourly frequencies.

Variable name	**Forcing**	**Daily frequency**	**Hourly frequency**
convective_fraction	NLDAS hourly	✓	✓
longwave_radiation	NLDAS hourly	✓	✓
potential_energy	NLDAS hourly	✓	✓
potential_evaporation	NLDAS hourly	✓	✓
pressure	NLDAS hourly	✓	✓
shortwave_radiation	NLDAS hourly	✓	✓
specific_humidity	NLDAS hourly	✓	✓
temperature	NLDAS hourly	✓	✓
total_precipitation	NLDAS hourly	✓	✓
wind_u	NLDAS hourly	✓	✓
wind_v	NLDAS hourly	✓	✓

Table C.2: Dynamic input variables used in the second experiment, where different numbers of variables are used for the daily and hourly frequencies.

Variable name	Forcing	Daily frequency	Hourly frequency	Note
prcp(mm/day)	Daymet daily	✓	✓	*LRR
srad(W/m2)	Daymet daily	✓	✓	*LRR
tmax(C)	Daymet daily	✓	✓	*LRR
tmin(C)	Daymet daily	✓	✓	*LRR
vp(Pa)	Daymet daily	✓	✓	*LRR
prcp(mm/day	Maurer daily	✓	✓	*LRR
srad(W/m2)	Maurer daily	✓	✓	*LRR
tmax(C	Maurer daily	✓	✓	*LRR
tmin(C)	Maurer daily	✓	✓	*LRR
vp(Pa)	Maurer daily	✓	✓	*LRR
convective_fraction	NLDAS hourly	-	✓	
longwave_radiation	NLDAS hourly	-	✓	
potential_energy	NLDAS hourly	-	✓	
potential_evaporation	NLDAS hourly	-	✓	
pressure	NLDAS hourly	-	✓	
shortwave_radiation	NLDAS hourly	-	✓	
specific_humidity	NLDAS hourly	-	✓	
temperature	NLDAS hourly	-	✓	
total_precipitation	NLDAS hourly	-	✓	
wind_u	NLDAS hourly	-	✓	
wind_v	NLDAS hourly	-	✓	

**LRR: low-resolution re-discretization is done when the original daily value is used at hourly frequency. Therefore, the original daily value is repeated 24 times.*

Table C.3: Names of the 26 static attributes used in the experiments

elev_mean	slope_mean	area_gages2	frac_forest
lai_max	lai_diff	gvf_max	gvf_diff
soil_depth_pelletier	soil_depth_statsgo	soil_porosity	soil_conductivity
max_water_content	sand_frac	silt_frac	clay_frac
carbonate_rocks_frac	geol_permeability	p_mean	pet_mean
aridity	frac_snow	high_prec_freq	high_prec_dur
low_prec_freq	low_prec_dur		

C.2 STRUCTURE OF THE MTS-LSTM MODEL ARCHITECTURE

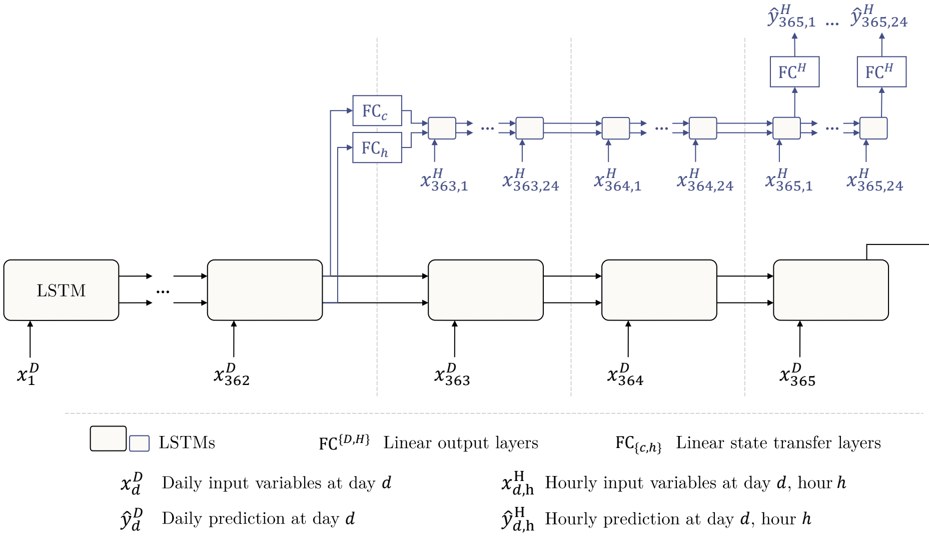
$\hat{y}^H_{365,1}$ … $\hat{y}^H_{365,24}$
$\hat{y}^D_{365}$
FC^H
FC^H
FC^D
FC_c
FC_h
$x^H_{363,1}$
$x^H_{363,24}$
$x^H_{364,1}$
$x^H_{364,24}$
$x^H_{365,1}$
$x^H_{365,24}$
LSTM
x^D_1
x^D_{362}
x^D_{363}
x^D_{364}
x^D_{365}
LSTMs
$FC^{\{D,H\}}$ Linear output layers
$FC_{\{c,h\}}$ Linear state transfer layers
x^D_d Daily input variables at day d
$x^H_{d,h}$ Hourly input variables at day d, hour h
$\hat{y}^D_d$ Daily prediction at day d
$\hat{y}^H_{d,h}$ Hourly prediction at day d, hour h

Figure C.1: Illustration of the MTS-LSTM architecture that uses one distinct LSTM model per timescale. In the depicted example, the daily and hourly input sequence lengths are $T^D = 365$ and $T^H = 72$ (we chose this value for the sake of a tidy illustration; the benchmarked model uses $T^H = 336$). In the sMTS-LSTM model (i.e. without distinct LSTM branches), FC_C and FC_h are identity functions, and the two branches (including the fully connected output layers FC^H and FC^D) share their model weights.
Source: this figure and its description were taken from Gauch et al. (2021).

C.3 ADDITIONAL RESULTS

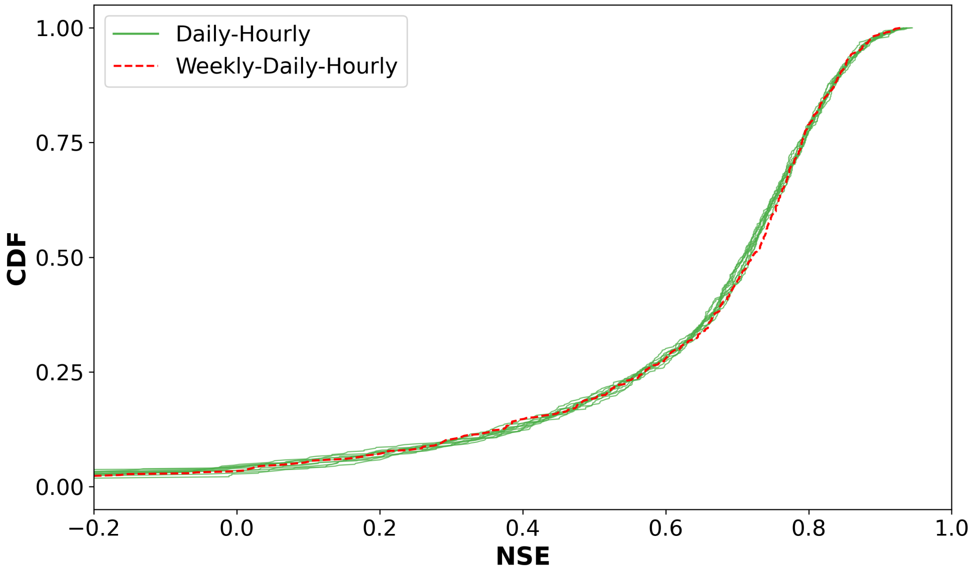

Figure C.2: Comparison of cumulative NSE distributions for different frequency sequences. The daily-hourly experiment includes 10 distributions, each corresponding to an ensemble member generated through different random initializations. The average of the 10 median NSE values is 0.71. In contrast, the weekly-daily-hourly experiment consists of a single simulation, yielding a median NSE of 0.72.

AUTHOR CONTRIBUTIONS AND CODE AVAILABILITY

Chapter 2: Acuña Espinoza, E., Loritz, R., Álvarez Chaves, M., Bäuerle, N., and Ehret, U.: To bucket or not to bucket? Analyzing the performance and interpretability of hybrid hydrological models with dynamic parameterization, Hydrol. Earth Syst. Sci., 28, 2705–2719, https://doi.org/10.5194/hess-28-2705-2024, 2024.

The original idea of the paper was developed by all authors. The codes were written by EAE with support from RL and MAC. The simulations were conducted by EAE. Results were further discussed by all authors. The draft was prepared by EAE. Reviewing and editing was provided by all authors. Funding was acquired by UE and NB. All authors have read and agreed to the current version of the paper.

The codes used to conduct all the analyses in this paper are publicly available at https://github.com/KIT-HYD/Hy2DL/tree/v1.0 (last access: 25 June 2024) (https://doi.org/10.5281/zenodo.11103634) (Acuña Espinoza et al., 2024a).

The CAMELS-US dataset is freely available at https://doi.org/10.5065/D6MW2F4D (Newman et al., 2022). The CAMELS-GB dataset is freely available at https://doi.org/10.5285/8344e4f3-d2ea-44f5-8afa-86d2987543a9 (Coxon et al., 2020b). All the data generated for this publication can be found at https://doi.org/10.5281/zenodo.11103634 (Acuña Espinoza et al., 2024a).

Chapter 3: Acuña Espinoza, E., Loritz, R., Kratzert, F., Klotz, D., Gauch, M., Álvarez Chaves, M., and Ehret, U.: Analyzing the generalization capabilities of a hybrid hydrological model for extrapolation to extreme events. Hydrology and Earth System Sciences, 29(5), 1277–1294. https://doi.org/10.5194/hess-29-1277-2025, 2025

The original idea of the paper was developed by all the authors. The codes were written by EAE, with support from FK, MG, and DK. The simulations were conducted by EAE. The results were further discussed by all the authors. The draft of the paper was prepared by EAE. Reviewing and editing was provided by all the authors. Funding was acquired by UE. All the authors have read and agreed to the current version of the paper.

The code used to conduct all the analyses can be found at https://doi.org/10.5281/zenodo.14191623 (Acuña Espinoza, 2024) and as part of our GitHub repository https://github.com/eduardoAcunaEspinoza/hybrid_extrapolation/tree/v0.2

The CAMELS US dataset is freely available at https://doi.org/10.5065/D6MW2F4D (Newman et al., 2022). All the data generated for this publication can be found at https://doi.org/10.5281/zenodo.14191623 (Acuña Espinoza, 2024)

Chapter 4: Acuña Espinoza, E., Kratzert, F., Klotz, D., Gauch, M., Álvarez Chaves, M., Loritz, R., and Ehret, U. Technical note: An approach for handling multiple temporal frequencies with different input dimensions using a single LSTM cell. Hydrology and Earth System Sciences, 29(6), 1749–1758. https://doi.org/10.5194/hess-29-1749-2025, 2025

The original idea of the paper was developed by FK, MG, and DK. The codes were written by EAE. The simulations were conducted by EAE. The results were discussed further by all the authors. The draft of the manuscript was prepared by EAE. Reviewing and editing were provided by all the authors. Funding was acquired by UE. All the authors read and agreed to the current version of the paper.

The code used for all analyses in this paper is publicly available at https://doi.org/10.5281/zenodo.14780059 (Acuña Espinoza, 2025). It is part of the Hy2DL library, which can be accessed on GitHub: https://github.com/eduardoAcunaEspinoza/Hy2DL (last access: 4 February 2025).

All the data generated for this publication can be found at https://doi.org/10.5281/zenodo.14780059 (Acuña Espinoza, 2025). The benchmark models can be found at https://doi.org/10.5281/zenodo.4095485 (Gauch et al., 2020b). The hourly NLDAS forcing and the hourly streamflow can be found at https://doi.org/10.5281/zenodo.4072701 (Gauch et al., 2020a). The CAMELS-US dataset can be found at https://doi.org/10.5065/D6G73C3Q (Newman et al., 2022). However, one should replace the original Maurer forcings with the extended version presented in https://doi.org/10.4211/hs.17c896843cf940339c3c3496d0c1c077 (Kratzert, 2019)

Eidesstattliche Versicherung gemäß § 13 Absatz 2 Satz 1 Ziffer 4 der Promotionsordnung des Karlsruher Instituts für Technologie (KIT) für die KIT-Fakultät für Bauingenieur-, Geo- und Umweltwissenschaften:

1. Bei der eingereichten Dissertation zu dem Thema *Enhancing hydrological rainfall-runoff simulation using machine learning methods* handelt es sich um meine eigenständig erbrachte Leistung.

2. Ich habe nur die angegebenen Quellen und Hilfsmittel benutzt und mich keiner unzulässigen Hilfe Dritter bedient. Insbesondere habe ich wörtlich oder sinngemäß aus anderen Werken übernommene Inhalte als solche kenntlich gemacht.

3. Die Arbeit oder Teile davon habe ich bislang nicht an einer Hochschule des In- oder Auslands als Bestandteil einer Prüfungs- oder Qualifikationsleistung vorgelegt.

4. Die Richtigkeit der vorstehenden Erklärungen bestätige ich.

5. Die Bedeutung der eidesstattlichen Versicherung und die straf-rechtlichen Folgen einer unrichtigen oder unvollständigen eidesstattlichen Versicherung sind mir bekannt.

Ich versichere an Eides statt, dass ich nach bestem Wissen die reine Wahrheit erklärt und nichts verschwiegen habe.

Karlsruhe, 2025

Eduardo José Acuña
Espinoza

OWN PUBLICATIONS

FIRST AUTHOR; PEER-REVIEWED INTERNATIONAL PUBLICATIONS

Acuña Espinoza, E., Loritz, R., Álvarez Chaves, M., Bäuerle, N., and Ehret, U.: To bucket or not to bucket? Analyzing the performance and interpretability of hybrid hydrological models with dynamic parameterization, Hydrol. Earth Syst. Sci., 28, 2705–2719, https://doi.org/10.5194/hess-28-2705-2024, 2024.

Acuña Espinoza, E., Loritz, R., Kratzert, F., Klotz, D., Gauch, M., Álvarez Chaves, M., and Ehret, U.: Analyzing the generalization capabilities of a hybrid hydrological model for extrapolation to extreme events. Hydrology and Earth System Sciences, 29(5), 1277–1294. https://doi.org/10.5194/hess-29-1277-2025, 2025

Acuña Espinoza, E., Kratzert, F., Klotz, D., Gauch, M., Álvarez Chaves, M., Loritz, R., and Ehret, U. Technical note: An approach for handling multiple temporal frequencies with different input dimensions using a single LSTM cell. Hydrology and Earth System Sciences, 29(6), 1749–1758. https://doi.org/10.5194/hess-29-1749-2025, 2025

COLLABORATIONS

Loritz, R., Dolich, A., **Acuña Espinoza, E.**, Ebeling, P., Guse, B., Götte, J., Hassler, S. K., Hauffe, C., Heidbüchel, I., Kiesel, J., Mälicke, M., Müller-Thomy, H., Stölzle, M., and Tarasova, L. (2024). CAMELS-DE: Hydro-meteorological time series and attributes for 1582 catchments in Germany. Earth System Science Data, 16(12), 5625–5642. https://doi.org/10.5194/essd-16-5625-2024

Mouris, K., **Acuña Espinoza, E.**, Schwindt, S. et al. Stability criteria for Bayesian calibration of reservoir sedimentation models. Model. Earth Syst. Environ. 9, 3643–3661 (2023). https://doi.org/10.1007/s40808-023-01712-7

OPEN SOURCE CODES

Hybrid Hydrological Modeling using Deep Learning methods (Hy^{2}DL): Python library to create hydrological models for rainfall-runoff prediction using deep learning methods,(https://github.com/eduardoAcunaEspinoza/Hy2DL).

Hydrological Model Calibration (HyMC): Python library designed to facilitate the calibration of process-based hydrological models for rainfall-runoff predictions,(https://github.com/eduardoAcunaEspinoza/HyMC).

BIBLIOGRAPHY

Acuña Espinoza, Eduardo (Nov. 2024). *Analyzing the generalization capabilities of hybrid hydrological models for extrapolation to extreme events*. DOI: 10.5281/zenodo.14191623.

Acuña Espinoza, Eduardo (Jan. 2025). *An approach for handling multiple temporal frequencies with different input dimensions using a single LSTM cell*. DOI: 10.5281/zenodo.14780059.

Acuña Espinoza, Eduardo, Ralf Loritz, and Manuel Álvarez Chaves (2024a). *KIT-HYD/Hy2DL: Preview release for submission (1.0)*. DOI: 10.5281/zenodo.11103634.

Acuña Espinoza, Eduardo, Ralf Loritz, Manuel Álvarez Chaves, Nicole Bäuerle, and Uwe Ehret (2024b). "To Bucket or not to Bucket? Analyzing the performance and interpretability of hybrid hydrological models with dynamic parameterization." In: *Hydrology and Earth System Sciences* 28.12, 2705–2719.

Addor, N., A. J. Newman, N. Mizukami, and M. P. Clark (2017). "The CAMELS data set: catchment attributes and meteorology for large-sample studies." In: *Hydrology and Earth System Sciences* 21.10, pp. 5293–5313. DOI: 10.5194/hess-21-5293-2017.

Bárdossy, A. and F. Anwar (2023). "Why do our rainfall–runoff models keep underestimating the peak flows?" In: *Hydrology and Earth System Sciences* 27.10, pp. 1987–2000. DOI: 10.5194/hess-27-1987-2023.

Beck, Hylke E. et al. (2020). "Global Fully Distributed Parameter Regionalization Based on Observed Streamflow From 4,229 Headwater Catchments." In: *Journal of Geophysical Research: Atmospheres* 125.17. e2019JD031485 10.1029/2019JD031485, e2019JD031485. DOI: 10.1029/2019JD031485.

Beck, Hylke et al. (Apr. 2016). "Global-scale regionalization of hydrologic model parameters." In: *Water Resources Research* 52, pp. 3599–3622. DOI: 10.1002/2015WR018247.

Bengio, Y., P. Simard, and P. Frasconi (1994). "Learning long-term dependencies with gradient descent is difficult." In: *IEEE Transactions on Neural Networks* 5.2, pp. 157–166. DOI: 10.1109/72.279181.

Bergström, S. (1992). *The HBV model – Its structure and applications (RH No. 4; SMHI Reports)*. Tech. rep. Last access: 23 June 2024. Swedish Meteorological and Hydrological Institute (SMHI).

Beven, K. J. and M. J. Kirkby (1979). "A physically based, variable contributing area model of basin hydrology / Un modèle à base physique de zone d'appel variable de l'hydrologie du bassin versant." In: *Hydrological Sciences Bulletin* 24.1, pp. 43–69. DOI: 10.1080/02626667909491834.

Beven, Keith (2012). *Rainfall-Runoff Modelling: The Primer*. 2nd. Wiley-Blackwell. DOI: 10.1002/9781119951001.

Bladé, E. et al. (2014). "Iber: herramienta de simulación numérica del flujo en ríos." In: *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería* 30.1, pp. 1–10. DOI: 10.1016/j.rimni.2012.07.004.

Boughton, W and Owen Droop (2003). "Continuous simulation for design flood estimation—a review." In: *Environmental Modelling & Software* 18.4, pp. 309–318.

Burnash, R. J. C., R. L. Ferral, and R. A. McGuire (1973). *A generalized streamflow simulation system: Conceptual modeling for digital computers*. Tech. rep. Last access: 23 June 2024. US Department of Commerce, National Weather Service.

Caviedes-Voullième, D., M. Morales-Hernández, M. R. Norman, and I. Özgen-Xian (2023). "SERGHEI (SERGHEI-SWE) v1.0: a performance-portable high-performance parallel-computing shallow-water solver for hydrology and environmental hydraulics." In: *Geoscientific Model Development* 16.3, pp. 977–1008. DOI: 10.5194/gmd-16-977-2023.

Chien, Hsiang-Yu S et al. (2021). "Slower is better: revisiting the forgetting mechanism in LSTM for slower information decay." In: *arXiv preprint arXiv:2105.05944*. DOI: 10.48550/arXiv.2105.05944.

Clark, Martyn P, Dmitri Kavetski, and Fabrizio Fenicia (2011). "Pursuing the method of multiple working hypotheses for hydrological modeling." In: *Water Resources Research* 47.9. DOI: 10.1029/2010WR009827.

Coxon, G. et al. (2020a). "CAMELS-GB: hydrometeorological time series and landscape attributes for 671 catchments in Great Britain." In: *Earth System Science Data* 12.4, pp. 2459–2483. DOI: 10.5194/essd-12-2459-2020.

Coxon, G. et al. (2020b). *Catchment attributes and hydro-meteorological timeseries for 671 catchments across Great Britain (CAMELS-GB)*. DOI: 10.5285/8344e4f3-d2ea-44f5-8afa-86d2987543a9.

Craig, James R. et al. (2020). "Flexible watershed simulation with the Raven hydrological modelling framework." In: *Environmental Modelling and Software* 129, p. 104728. DOI: 10.1016/j.envsoft.2020.104728.

Dal Molin, M., D. Kavetski, and F. Fenicia (2021). "SuperflexPy 1.3.0: an open-source Python framework for building, testing, and improving conceptual hydrological models." In: *Geoscientific Model Development* 14.11, pp. 7047–7072. DOI: 10.5194/gmd-14-7047-2021.

Di Baldassarre, G. and A. Montanari (2009). "Uncertainty in river discharge observations: a quantitative analysis." In: *Hydrology and Earth System Sciences* 13.6, pp. 913–921. DOI: 10.5194/hess-13-913-2009.

Donoho, David (2017). "50 years of data science." In: *Journal of Computational and Graphical Statistics* 26.4, pp. 745–766.

Duan, Qingyun, Soroosh Sorooshian, and Vijai K. Gupta (1994). "Optimal use of the SCE-UA global optimization method for calibrating watershed models." In: *Journal of Hydrology* 158.3, pp. 265–284. DOI: https://doi.org/10.1016/0022-1694(94)90057-4.

Ehret, U. et al. (2020). "Adaptive clustering: reducing the computational costs of distributed (hydrological) modelling by exploiting time-variable similarity among model elements." In: *Hydrology and Earth System Sciences* 24.9, pp. 4389–4411. DOI: 10.5194/hess-24-4389-2020.

England Jr, John F et al. (2019). *Guidelines for determining flood flow frequency—Bulletin 17C*. Tech. rep. US Geological Survey.

Feng, Dapeng, Kuai Fang, and Chaopeng Shen (2020). "Enhancing Streamflow Forecast and Extracting Insights Using Long-Short Term Memory Networks With Data Integration at Continental Scales." In: *Water Resources Research* 56.9. e2019WR026793 2019WR026793, e2019WR026793. DOI: 10.1029/2019WR026793.

Feng, Dapeng, Jiangtao Liu, Kathryn Lawson, and Chaopeng Shen (2022). "Differentiable, Learnable, Regionalized Process-Based Models With Multiphysical Outputs can Approach State-Of-The-Art Hydrologic Prediction Accuracy." In: *Water Resources Research* 58.10. e2022WR032404 2022WR032404, e2022WR032404. DOI: 10.1029/2022WR032404.

Frame, J. M. et al. (2022). "Deep learning rainfall–runoff predictions of extreme events." In: *Hydrology and Earth System Sciences* 26.13, pp. 3377–3392. DOI: 10.5194/hess-26-3377-2022.

Frame, Jonathan M, Frederik Kratzert, Hoshin V Gupta, Paul Ullrich, and Grey S Nearing (2023). "On strictly enforced mass conservation constraints for modelling the Rainfall-Runoff process." In: *Hydrological Processes* 37.3, e14847. DOI: 10.1002/hyp.14847.

Frame, Jonathan M et al. (2021). "Post-processing the national water model with long short-term memory networks for streamflow predictions and model diagnostics." In: *JAWRA Journal of the American Water Resources Association* 57.6, pp. 885–905. DOI: 10.1111/1752-1688.12964.

Gauch, M. et al. (2021). "Rainfall–runoff prediction at multiple timescales with a single Long Short-Term Memory network." In: *Hydrology and Earth System Sciences* 25.4, pp. 2045–2062. DOI: 10.5194/hess-25-2045-2021.

Gauch, Martin et al. (2020a). *Data for "Rainfall-Runoff Prediction at Multiple Timescales with a Single Long Short-Term Memory Network"*. Accessed: 24 Oct 2024. DOI: 10.5281/zenodo.4072701.

Gauch, Martin et al. (2020b). *Models and Predictions for "Rainfall-Runoff Prediction at Multiple Timescales with a Single Long Short-Term Memory Network"*. Accessed: 24 Oct 2024. DOI: 10.5281/zenodo.4095485.

Herath, H. M. V. V., J. Chadalawada, and V. Babovic (2021). "Hydrologically informed machine learning for rainfall–runoff modelling: towards distributed modelling." In: *Hydrology and Earth System Sciences* 25.8, pp. 4373–4401. DOI: 10.5194/hess-25-4373-2021.

Hersbach, Hans et al. (2020). "The ERA5 global reanalysis." In: *Quarterly Journal of the Royal Meteorological Society* 146.730, pp. 1999–2049. DOI: 10.1002/qj.3803. eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3803.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). "Long Short-Term Memory." In: *Neural Computation* 9.8, pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

Hoge, M., A. Scheidegger, M. Baity-Jesi, C. Albert, and F. Fenicia (2022). "Improving hydrologic models for predictions and process understanding using neural ODEs." In: *Hydrology and Earth System Sciences* 26.19, pp. 5085–5102. DOI: 10.5194/hess-26-5085-2022.

Houska, Tobias, Philipp Kraft, Alejandro Chamorro-Chavez, and Lutz Breuer (2015). "SPOTting model parameters using a ready-made python package." In: *PloS one* 10.12, e0145180. DOI: 10.1371/journal.pone.0145180.

Jiang, Shijie, Yi Zheng, and Dimitri Solomatine (2020). "Improving AI system awareness of geoscience knowledge: Symbiotic integration of physical approaches and deep learning." In: *Geophysical Research Letters* 47.13, e2020GL088229.

Kingma, Diederik P and Jimmy Ba (2014). "Adam: A method for stochastic optimization." In: *arXiv preprint arXiv:1412.6980*. DOI: 10.48550/arXiv.1412.6980.

Kirchner, James W (2006). “Getting the right answers for the right reasons: Linking measurements, analyses, and models to advance the science of hydrology.” In: *Water resources research* 42.3. DOI: 10.1029/2005WR004362.

Klotz, D. et al. (2022). “Uncertainty estimation with deep learning for rainfall–runoff modeling.” In: *Hydrology and Earth System Sciences* 26.6, pp. 1673–1693. DOI: 10.5194/hess-26-1673-2022.

Kraft, B., M. Jung, M. Körner, S. Koirala, and M. Reichstein (2022). “Towards hybrid modeling of the global hydrological cycle.” In: *Hydrology and Earth System Sciences* 26.6, pp. 1579–1614. DOI: 10.5194/hess-26-1579-2022.

Kratzert, F., M. Gauch, D. Klotz, and G. Nearing (2024). “HESS Opinions: Never train a Long Short-Term Memory (LSTM) network on a single basin.” In: *Hydrology and Earth System Sciences* 28.17, pp. 4187–4201. DOI: 10.5194/hess-28-4187-2024.

Kratzert, F., D. Klotz, C. Brenner, K. Schulz, and M. Herrnegger (2018). “Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks.” In: *Hydrology and Earth System Sciences* 22.11, pp. 6005–6022. DOI: 10.5194/hess-22-6005-2018.

Kratzert, Frederik (2019). *CAMELS Extended Maurer Forcing Data*. Accessed: 24 Oct 2024. DOI: 10.4211/hs.17c896843cf940339c3c3496d0c1c077.

Kratzert, Frederik, Martin Gauch, Grey Nearing, and Daniel Klotz (2022). “NeuralHydrology — A Python library for Deep Learning research in hydrology.” In: *Journal of Open Source Software* 7.71, p. 4050. DOI: 10.21105/joss.04050.

Kratzert, Frederik, Mathew Herrnegger, Daniel Klotz, Sepp Hochreiter, and Günter Klambauer (2019a). “NeuralHydrology – Interpreting LSTMs in Hydrology.” In: *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Cham: Springer International Publishing, pp. 347–362. DOI: 10.1007/978-3-030-28954-6_19.

Kratzert, Frederik et al. (2019b). “Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets.” In: *Hydrology and Earth System Sciences* 23.12, pp. 5089–5110. DOI: 10.5194/hess-23-5089-2019.

LARSIM-Entwicklergemeinschaft (2022). *Das Wasserhaushaltsmodell LARSIM – Modellgrundlagen und Anwendungsbeispiele*.

Lan, T., K. Lin, C.-Y. Xu, X. Tan, and X. Chen (2020). “Dynamics of hydrological-model parameters: mechanisms, problems and solutions.” In: *Hydrology and Earth System Sciences* 24.3, pp. 1347–1366. DOI: 10.5194/hess-24-1347-2020.

Leavesley, G., R. Lichty, B. Troutman, and L Saindon (1983). *Precipitation-runoff modelling system: user's manual, Report 83–4238*. Tech. rep. Report 83–4238. last access: 23 June 2024. US Geological SurveyWater Resources Investigations.

Lechner, Mathias and Ramin Hasani (2020). “Learning Long-Term Dependencies in Irregularly-Sampled Time Series.” In: *arXiv preprint arXiv:2006.04418*. DOI: 10.48550/arXiv.2006.04418.

Lees, T. et al. (2021). “Benchmarking data-driven rainfall-runoff models in Great Britain: a comparison of long short-term memory (LSTM)-based models with four lumped conceptual models.” In: *Hydrology and Earth System Sciences* 25.10, pp. 5517–5534. DOI: 10.5194/hess-25-5517-2021.

Lees, T. et al. (2022). “Hydrological concept formation inside long short-term memory (LSTM) networks.” In: *Hydrology and Earth System Sciences* 26.12, pp. 3079–3101. DOI: 10.5194/hess-26-3079-2022.

Li, Bu, Ting Sun, Fuqiang Tian, and Guangheng Ni (2023). “Enhancing process-based hydrological models with embedded neural networks: A hybrid approach.” In: *Journal of Hydrology* 625, p. 130107. DOI: https://doi.org/10.1016/j.jhydrol.2023.130107.

Li, Z. et al. (2025). “SERGHEI v2.0: introducing a performance-portable, high-performance, three-dimensional variably saturated subsurface flow solver (SERGHEI-RE).” In: *Geoscientific Model Development* 18.2, pp. 547–562. DOI: 10.5194/gmd-18-547-2025.

Loritz, R., M. Hrachowitz, M. Neuper, and E. Zehe (2021). “The role and value of distributed precipitation data in hydrological models.” In: *Hydrology and Earth System Sciences* 25.1, pp. 147–167. DOI: 10.5194/hess-25-147-2021.

Loritz, R. et al. (2018). “On the dynamic nature of hydrological similarity.” In: *Hydrology and Earth System Sciences* 22.7, pp. 3663–3684. DOI: 10.5194/hess-22-3663-2018.

Loritz, R. et al. (2024). “CAMELS-DE: hydro-meteorological time series and attributes for 1582 catchments in Germany.” In: *Earth System Science Data* 16.12, pp. 5625–5642. DOI: 10.5194/essd-16-5625-2024.

Martinez, Guillermo F. and Hoshin V. Gupta (2010). “Toward improved identification of hydrological models: A diagnostic evaluation of the “abcd” monthly water balance model for the conterminous United States.” In: *Water Resources Research* 46.8. DOI: 10.1029/2009WR008294.

Mulvaney, T. J. (1850). “On the use of self-registering rain and flood gauges in making observations of the relations of rainfall and of flood discharges in a given catchment.” In: *Proceedings of the Institution of Civil Engineers, Dublin* 4, pp. 18–31.

Nash, J. E. and J. V. Sutcliffe (1970). “River flow forecasting through conceptual models part I — A discussion of principles.” In: *Journal of Hydrology* 10.3, pp. 282–290. DOI: 10.1016/0022-1694(70)90255-6.

Nearing, G. S. et al. (2022). “Technical note: Data assimilation and autoregression for using near-real-time streamflow observations in long short-term memory networks.” In: *Hydrology and Earth System Sciences* 26.21, pp. 5493–5513. DOI: 10.5194/hess-26-5493-2022.

Nearing, Grey S et al. (2021). “What role does hydrological science play in the age of machine learning?” In: *Water Resources Research* 57.3, e2020WR028091. DOI: 10.1029/2020WR028091.

Nearing, Grey et al. (2024). “Global prediction of extreme floods in ungauged watersheds.” In: *Nature* 627.8004, pp. 559–563. DOI: 10.1038/s41586-024-07145-1.

Nevo, S. et al. (2022). “Flood forecasting with machine learning models in an operational framework.” In: *Hydrology and Earth System Sciences* 26.15, pp. 4013–4032. DOI: 10.5194/hess-26-4013-2022.

Newman, Andrew J et al. (2015). “Development of a large-sample watershed-scale hydrometeorological data set for the contiguous USA: data set characteristics and assessment of regional variability in hydrologic model performance.” In: *Hydrology and Earth System Sciences* 19.1, pp. 209–223.

Newman, Andrew et al. (2022). *CAMELS: Catchment Attributes and MEteorology for Large-sample Studies. Version 1.2.*

Pascanu, Razvan, Tomas Mikolov, and Yoshua Bengio (2013). "On the difficulty of training Recurrent Neural Networks." In: DOI: 10.48550/arXiv.1211.5063. arXiv: 1211.5063 [cs.LG].

Paszke, Adam et al. (2019). "Pytorch: An imperative style, high-performance deep learning library." In: *Advances in neural information processing systems* 32. DOI: 10.48550/arXiv.1912.01703.

Perrin, Charles, Claude Michel, and Vazken Andréassian (2003). "Improvement of a parsimonious model for streamflow simulation." In: *Journal of Hydrology* 279.1, pp. 275–289. DOI: 10.1016/S0022-1694(03)00225-7.

Reichstein, Markus et al. (2019). "Deep learning and process understanding for data-driven Earth system science." In: *Nature* 566.7743, pp. 195–204. DOI: 10.1038/s41586-019-0912-1.

Sabater, J. Muñoz et al. (2021). "ERA5-Land: a state-of-the-art global reanalysis dataset for land applications." In: *Earth System Science Data* 13.9, pp. 4349–4383. DOI: 10.5194/essd-13-4349-2021.

Shen, C et al. (2018). "HESS Opinions: Incubating deep-learning-powered hydrologic science advances as a community." In: *Hydrology and Earth System Sciences* 22.11, pp. 5639–5656.

Shen, Chaopeng et al. (Aug. 2023). "Differentiable modelling to unify machine learning and physical models for geosciences." en. In: *Nature Reviews Earth & Environment* 4.8, pp. 552–567. DOI: 10.1038/s43017-023-00450-9.

Slater, L. J. et al. (2023). "Hybrid forecasting: blending climate predictions with AI models." In: *Hydrology and Earth System Sciences* 27.9, pp. 1865–1889. DOI: 10.5194/hess-27-1865-2023.

Spieler, D., J. Mai, J. R. Craig, B. A. Tolson, and N. Schütze (2020). "Automatic model structure identification for conceptual hydrologic models." In: *Water Resources Research* 56, e2019WR027009. DOI: 10.1029/2019WR027009.

Starmer, J. (Nov. 2022). *Long Short-Term Memory (LSTM), Clearly Explained*. https://www.youtube.com/watch?v=YCzL96nL7j0&t=884s. Accessed: 2025-02-20.

Tsai, Wen-Ping et al. (Oct. 2021). "From calibration to parameter learning: Harnessing the scaling effects of big data in geoscientific modeling." In: *Nature Communications* 12.1, p. 5988. DOI: 10.1038/s41467-021-26107-z.

US Geological Service (2016). *Water Data for the Nation*. Accessed: 19 Oct 2024. DOI: 10.5066/F7P55KJN.

Virtanen, Pauli et al. (Mar. 2020). "SciPy 1.0: fundamental algorithms for scientific computing in Python." en. In: *Nature Methods* 17.3, pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

Vrugt, Jasper A. (2016). "Markov chain Monte Carlo simulation using the DREAM software package: Theory, concepts, and MATLAB implementation." In: *Environmental Modelling and Software* 75, pp. 273–316. DOI: 10.1016/j.envsoft.2015.08.013.

Westerberg, I. K. and H. K. McMillan (2015). "Uncertainty in hydrological signatures." In: *Hydrology and Earth System Sciences* 19.9, pp. 3951–3968. DOI: 10.5194/hess-19-3951-2015.

Xia, Youlong et al. (2012). "Continental-scale water and energy flux analysis and validation for the North American Land Data Assimilation System project phase 2 (NLDAS-2): 1. Intercomparison and application of model products." In: *Jour-*

nal of Geophysical Research: Atmospheres 117.D3. DOI: https://doi.org/10.1029/2011JD016048.

Zhang, Xin and Jiali You (2020). “A gated dilated causal convolution based encoder-decoder for network traffic forecasting.” In: *IEEE Access* 8, pp. 6087–6097.

COLOPHON

This document was typeset using the typographical look-and-feel classicthesis developed by André Miede. The style was inspired by Robert Bringhurst's seminal book on typography "*The Elements of Typographic Style*". classicthesis is available for both LaTeX and LyX:

https://bitbucket.org/amiede/classicthesis/

Happy users of classicthesis usually send a real postcard to the author, a collection of postcards received so far is featured here:

http://postcards.miede.de/

Final Version as of July 3, 2025 (classicthesis v4.6).