Hybrid Physics-AI Approaches Integrating Neural Networks into Distributed Hydrological Models for Regional Flood Simulation

Ngo Nghi Truyen Huynh

▶ **To cite this version:**

Ngo Nghi Truyen Huynh. Hybrid Physics-AI Approaches Integrating Neural Networks into Distributed Hydrological Models for Regional Flood Simulation. Environmental Sciences. Aix-Marseile Université, 2025. English. ⟨NNT : 2025AIXM0174⟩. ⟨tel-05151541v1⟩

HAL Id: tel-05151541

https://hal.science/tel-05151541v1

Submitted on 8 Jul 2025 (v1), last revised 28 Jul 2025 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons CC BY-NC-ND 4.0 - Attribution - Non-commercial use - No Derivative Works - International License

amU Aix Marseille Université

NNT : 2025AIXM0174

THÈSE DE DOCTORAT

Soutenue à AMU — Aix-Marseille Université

le 25 juin 2025 par

Ngo Nghi Truyen HUYNH

Hybrid Physics-AI Approaches Integrating Neural Networks into Distributed Hydrological Models for Regional Flood Simulation

Approches hybrides physique-IA intégrant des réseaux neuronaux dans des modèles hydrologiques distribués pour la simulation des crues à l'échelle régionale

Discipline
Sciences de l'Environnement

Spécialité
Écologie

École doctorale
ED 251 Sciences de l'Environnement

Laboratoire/Partenaires de recherche
UMR RECOVER, INRAE Aix-en-Provence

Composition du jury

Eric GAUME *IPEF, Université Gustave Eiffel*	Rapporteur, Président du jury
Nicolas FLIPO *DR, Mines Paris – PSL*	Rapporteur
Flora BRANGER *IR, INRAE RiverLy*	Examinatrice
Nicolas LEMOINE *MCF, Institut Agro Rennes-Angers*	Examinateur
Pierre-André GARAMBOIS *CR, INRAE RECOVER*	Directeur de thèse
Hélène ROUX *PR, Toulouse INP/IMFT*	Co-encadrante de thèse
Claire LAUVERNET *CR, INRAE RiverLy*	Invitée

INRAE

Affidavit

I, the undersigned, Ngo Nghi Truyen Huynh, hereby declare that the work presented in this manuscript is my own, conducted under the scientific supervision of Pierre-André Garambois (supervisor), Benjamin Renard (co-supervisor), and Hélène Roux (co-supervisor), in accordance with the principles of honesty, integrity and responsibility inherent to the research mission. The research work and the writing of this manuscript have been carried out in compliance with both the French National Charter for Research Integrity and the AMU Charter on the Fight Against Plagiarism.

This work has not previously been submitted, in whole or in part, in this or any other country, for any academic examination.

Done in Aix-en-Provence, on April 15, 2025.

This work is licensed under Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International Public License

List of publications and conference participation

A List of publications and/or patents produced as part of the PhD project

A.1 Journal articles (rank A)

1. Huynh, N. N. T., Garambois, P.-A., Renard, B., Colleoni, F., *et al.* (2025). "A Distributed Hybrid Physics-AI Framework for Learning Corrections of Internal Hydrological Fluxes and Enhancing High-Resolution Regionalized Flood Modeling," *Hydrol. Earth Syst. Sci. (accepted)*. DOI : 10.5194/egusphere-2024-3665.
2. Huynh, N. N. T., Garambois, P.-A., Colleoni, F., Renard, B., Roux, H., *et al.* (2024). "Learning Regionalization Using Accurate Spatial Cost Gradients Within a Differentiable High-Resolution Hydrological Model : Application to the French Mediterranean Region," *Water Resour. Res.*, vol. 60, no. 11. DOI : 10.1029/2024WR037544.
3. Huynh, N. N. T., Garambois, P.-A., Colleoni, F., and Javelle, P. (2023). "Signatures-and-sensitivity-based multi-criteria variational calibration for distributed hydrological modeling applied to Mediterranean floods," *J. Hydrol.*, vol. 625, p. 129992. DOI : 10.1016/j.jhydrol.2023.129992.
4. Ettalbi, M., Garambois, P.-A., Huynh, N. N. T., *et al.* (2025). "Improving parameter regionalization learning for spatialized differentiable hydrological models by assimilation of satellite-based soil moisture data," *J. Hydrol.*, vol. 660, p. 133300. DOI : 10.1016/j.jhydrol.2025.133300
5. Garambois, P.-A., Colleoni, F., Huynh, N. N. T., *et al.* (2025). "Spatially Distributed Gradient-Based Calibration and Parametric Sensitivity of a Spatialized Hydrological Model Over 235 French Catchments," *J. Hydrol. : Reg. Stud. (accepted)*. URL : `https://hal.science/hal-04989183`.

A.2 Articles under review/Working papers

1. Huynh, N. N. T., Garambois, P.-A., Colleoni, F., and Monnier, J. (2025). "Hybrid Physics-AI and Neural ODE Approaches for Spatially Distributed Hydrological Modeling," *submitted*.
2. Huynh, N. N. T., Garambois, P.-A., Renard, B., Colleoni, F., *et al.* (2024). "Multiscale learnable physical modeling and data assimilation framework : application to high-resolution regionalized hydrological simulation of flash floods," *working paper*. DOI : 10.22541/au.170709054.44271526/v2
3. Colleoni, F., Huynh, N. N. T., Garambois, P.-A., *et al.* (2025). "SMASH v1.0 : A Differentiable and Regionalizable High-Resolution Hydrological Modeling and Data Assimilation Framework," *Geosci. Model Dev. (in discussion)*. DOI : 10.5194/egusphere-2025-690.

A.3 Conference proceedings

1. Huynh, N. N. T., Garambois, P.-A., Colleoni, F., Renard, B., and Roux, H. (2023). "Multi-gauge Hydrological Variational Data Assimilation : Regionalization Learning with Spatial Gradients using Multilayer Perceptron and Bayesian-Guided Multivariate Regression," in *Colloque SHF 2023—Prévision des crues et des inondations*, Toulouse. DOI : 10.48550/arXiv.2307.02497.
2. Huynh, N. N. T., Garambois, P.-A., Renard, B., and Monnier, J. (2025). "A General Framework for Integrating Neural Networks into Numerical Resolution Methods for Spatially Distributed Hydrological Models," in *EGU General Assembly 2025*, Vienna. DOI : 10.5194/egusphere-egu25-2740.

3. Huynh, N. N. T., Garambois, P.-A., Colleoni, F., Renard, B., and Roux, H. (2023). "Advanced Hybrid Data Assimilation for Parameter Regionalization within a Differentiable Spatially Distributed Hydrological Model and Uncertainty Correction with Bidirectional LSTM," in *AGU Fall Meeting 2023*, San Francisco. URL : https://hal.inrae.fr/hal-04389155.
4. Ettalbi, M., Garambois, P.-A., Huynh, N. N. T., Ferreira, E., and Baghdadi, N. (2025). "GPU-Enabled Cell-to-Cell Routing in a High-Resolution Hybrid Distributed Hydrological Model with Multi-Source Remote Sensing Data Assimilation : A Continental-Scale Computational Approach," in *EGU General Assembly 2025*, Vienna. DOI : 10.5194/egusphere-egu25-9176.
5. Ettalbi, M., Garambois, P.-A., Baghdadi, N., Ferreira, E., and Huynh, N. N. T. (2024). "Multi-source in situ and satellite variational data assimilation into a fully distributed hydrological model for floods and droughts modeling over poorly gauged and ungauged areas," in *EGU General Assembly 2024*, Vienna. DOI : 10.5194/egusphere-egu24-8214.
6. Garambois, P.-A., Larnier, K., Monnier, J., Huynh, N. N. T., Colleoni, F., Renard, B., *et al.* (2023). "Towards enhanced regionalization of hydrologic-hydraulic river network models with assimilation of multi-source data and SWOT hydraulic visibility," in *AGU Fall Meeting 2023*, San Francisco. URL : https://ui.adsabs.harvard.edu/abs/2023AGUFM.H43N2283G.

B Participation in conferences and seasonal schools during the PhD period

B.1 International conferences

1. *General Assembly 2025 of the European Geosciences Union (EGU)*, Vienna, Austria, 27 April–2 May 2025.
2. *American Geophysical Union (AGU) Fall Meeting 2023*, San Francisco, USA, 11–15 December 2023.

B.2 National symposia and congresses

1. *Colloque SHF 2023—Prévision des crues et des inondations—Avancées, valorisation et perspectives*, Toulouse, France, 28–30 November 2023.
2. *30 Years of Progress in Radar Altimetry Symposia*, Montpellier, France, 02–07 September 2024.
3. *32ème congrès de l'école doctorale Sciences de l'Environnement*, Marseille, France, 24–25 April 2025.

B.3 Seasonal schools

1. *Spring School on Data-driven Model Learning for Dynamic Systems*, Lyon, France, 07–11 April 2025.
2. *12th Advanced Training Course on Land Remote Sensing*, Wrocław, Poland, 26–29 September 2023.

Abstract

River discharge estimation and flood forecasting remain among the most compelling challenges in hydrological sciences, especially in ungauged basins and under climate change. Hydrological models are fundamental to flood warning systems, and enhancing their realism and accuracy is crucial. However, these models are inherently conceptual and subject to several limitations, including a lack of scale-relevant theory, intrinsic uncertainties in their structure and parameterization, and equifinality in parameter estimation, often due to sparse constraining data such as discharge measurements. In the age of artificial intelligence (AI) and big data, hydrological models can benefit from AI algorithms by exploiting massive amounts of information from heterogeneous datasets. This thesis introduces a set of innovative hybrid physics-AI approaches, seamlessly integrating process-based and AI-driven models, to improve hydrological modeling, with a particular focus on flash flood modeling in the French Mediterranean region. The suggested methods are implemented within `smash` (Spatially distributed Modeling and ASsimilation for Hydrology), a grid-based differentiable hydrological model developed for water research and operational applications. The key developments are:

- Enhancing model accuracy through a multi-criteria calibration approach using multi-scale signatures;
- Integrating neural networks into the forward model to estimate the transfer function between descriptors and parameters, enabling regionalization for ungauged sites;
- Seeking optimal solutions using a Bayesian-weighted estimation method in the context of equifinality;
- Reducing structural uncertainty by integrating neural networks to refine internal water fluxes within the model;
- Embedding neural ordinary differential equations (neural ODEs) solved with an implicit numerical scheme suited for hybrid approaches.

The results demonstrate that:

- Multi-criteria calibration with hydrological signatures improves flood simulation performance while maintaining acceptable global scores;
- Neural networks effectively learn complex non-linear functions between physical descriptors and conceptual parameters, exhibiting strong regionalization performance and better model controllability than linear regression in challenging calibration scenarios;
- Stability of the optimal solution is enhanced by a Bayesian-weighted estimate of background values;
- Neural networks, embedded for internal flux correction within a state-space system (from analytical or numerical ODE solutions), improve model accuracy and robustness. Detailed analysis reveals the nuanced behavior of hybrid models and their enhanced versatility, demonstrating the potential of hybrid physics-AI approaches to uncover laws and parameterizations in hydrology and beyond.

These advancements have improved spatially distributed hydrological modeling by integrating AI into physical models, which can be applied to operational flood forecasting and support the development of new hydraulic approaches for runoff and flood modeling.

Keywords: flood modeling, distributed hydrological models, artificial intelligence, data assimilation, differentiable optimization

Résumé

L'estimation du débit des rivières et la prévision des inondations restent parmi les défis les plus importants des sciences hydrologiques, en particulier pour les bassins versants non jaugés et sous l'effet du changement climatique. Les modèles hydrologiques sont fondamentaux pour les systèmes de prévision des crues, et améliorer leur réalisme et leur précision est crucial. Cependant, ces modèles sont intrinsèquement conceptuels et soumis à plusieurs limitations, notamment l'absence de théorie pertinente à l'échelle des modèles, les incertitudes inhérentes à leur structure et à leur paramétrage, ainsi que l'équifinalité dans l'estimation des paramètres, souvent due à des données d'observation éparses telles que les mesures de débit. A l'ère de l'intelligence artificielle (IA) et des données massives, les modèles hydrologiques peuvent tirer parti des algorithmes d'IA pour exploiter des quantités massives d'informations provenant de jeux de données hétérogènes. Cette thèse présente un ensemble d'approches hybrides innovantes physiques-IA, avec une intégration sans couture des processus physiques et des modèles pilotés par l'IA, afin d'améliorer la modélisation hydrologique, en se concentrant particulièrement sur la modélisation des crues rapides dans la région méditerranéenne française. Les méthodes proposées sont mises en œuvre au sein de `smash` (Spatially distributed Modeling and ASsimilation for Hydrology), un modèle hydrologique différentiable spatialisé, développé pour la recherche sur la ressource en eau et les applications opérationnelles. Les développements principaux sont :

- Amélioration de la précision du modèle par une approche de calibration multi-critères basée sur des signatures multi-échelles ;
- Intégration des réseaux de neurones dans le modèle direct pour estimer la fonction de transfert des descripteurs aux paramètres, permettant la régionalisation sur des sites non jaugés ;
- Recherche de solutions optimales en utilisant une méthode d'estimation pondérée bayésienne en contexte d'équifinalité ;
- Diminution de l'incertitude structurelle en intégrant des réseaux de neurones pour corriger les flux d'eau internes au modèle ;
- Intégration d'équations différentielles ordinaires neuronales (EDO neuronales) résolues avec un schéma numérique implicite adapté aux approches hybrides.

Les résultats montrent que :

- La calibration multi-critères avec des signatures hydrologiques améliore les performances de simulation des crues tout en maintenant des scores globaux acceptables ;
- Les réseaux de neurones apprennent efficacement des fonctions non-linéaires complexes entre descripteurs physiques et paramètres conceptuels, démontrant une forte capacité de régionalisation et une meilleure contrôlabilité du modèle par rapport à la régression linéaire dans des scénarios de calibration difficiles ;
- La stabilité de la solution optimale est améliorée par une estimation pondérée bayésienne des valeurs initiales ;
- Les réseaux de neurones, intégrés pour la correction de flux internes dans un système d'espace-état (à partir de solutions analytiques ou numériques d'EDO), améliorent la précision et la robustesse du modèle. Une analyse détaillée révèle le comportement nuancé des modèles hybrides et leur versatilité améliorée, démontrant le potentiel des approches hybrides physiques-IA pour découvrir des lois et paramétrisations en hydrologie et plus largement.

Ces avancées ont permis d'améliorer la modélisation hydrologique spatialisée en intégrant l'IA dans les modèles physiques qui pourront être appliqués pour la prévision opérationnelle des crues, sous-tendre le développement de nouvelles approches hydrauliques pour le ruissellement et les inondations.

Mots-clés : modélisation des crues, modèles hydrologiques distribuées, intelligence artificielle, assimilation de données, optimisation différentiable

Remerciements

Ce sont les beaux jours de fin juin, lorsque les lavandes fleurissent au Tholonet, le long de la route de Cézanne. Dans quelques semaines, les paysans récolteront ces trésors violets, symbole d'une saison réussie. J'ai eu, à cette occasion, la chance de récolter les fruits de mon engagement passionné, où le soleil de Provence et le chant des cigales ont accompagné l'aboutissement de ma thèse. Ce n'est pas un grand exploit de très longue durée, mais c'est suffisant pour ouvrir le premier chapitre de ma carrière.

Je tiens d'abord à remercier les financements qui ont rendu cette thèse possible : le Centre National de Prévision des Crues, Service Central Vigicrues, SCHAPI ; le projet ANR MUFFINS (MUltiscale Flood Forecasting with INnovating Solutions), ANR-21-CE04-0021-01 ; et le projet européen NEPTUNE (Nowcasting and impact-basEd PredicTions of inUndations in mediterraNean catchmEnts).

Je remercie chaleureusement les membres du jury : Eric Gaume, Nicolas Flipo, Flora Branger, Nicolas Lemoine, et Claire Lauvernet qui ont accepté d'évaluer ma thèse et qui ont enrichi ma soutenance par leurs questions pertinentes et leurs conseils avisés.

Je remercie également tous les membres de mon équipe RHAX, ceux qui restent et ceux qui sont partis, une belle équipe composée de belles personnes. Merci pour votre générosité à partager vos connaissances scientifiques ainsi que culturelles dont j'ai beaucoup appris. Merci François pour tout ce que j'ai appris à tes côtés, que ce soit sur le codage en tant que collègue ou lors des moments précieux que nous avons partagés en tant qu'ami.

J'adresse mes plus grandes gratitudes à mes encadrants-enseignants : Pierre-André Garambois, Benjamin Renard, et Hélène Roux. Sans vous, je n'aurais jamais connu un tel enrichissement qu'aujourd'hui. Je garde une immense gratitude envers mon directeur Pierre-André. Que de temps nous avons passé ensemble à partager et discuter des sciences, des mathématiques, du numérique et quel énorme apprentissage j'ai réalisé grâce à toi.

Je n'oublie pas de remercier ma famille : ma petite-sœur qui me supporte avec bienveillance et avec qui j'ai vécu une enfance inoubliable ; tout mon respect et ma gratitude vont à ma mère, qui m'a donné tout le nécessaire avec son amour immense pour devenir la personne que je suis aujourd'hui, et qui me guidera pour toujours.

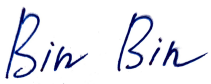
Enfin, je dédie ces lignes à ma femme Hà Trang. Tu es la première lumière dans mes joies, et la dernière présence dans mes chagrins. C'est pour toi que bat mon cœur, aujourd'hui, demain, et pour toujours.

Cette page est dédiée à mon père Huỳnh Viết Tiến ...

Gửi Ba—thần tượng duy nhất của đời con,

Nhớ hồi còn nhỏ, mỗi khi thi Toán được thành tích gì đó, con chỉ muốn lao thẳng về nhà thật nhanh để kể cho Ba. Đó là thành quả của những điều Ba dạy, của niềm đam mê bất tận với Toán học mà Ba khơi dậy trong con để có thể bước đi đầy tự tin trên con đường học tập của mình. Con vẫn đi theo con đường đó suốt bao năm qua, đầy kiên định và nhiệt huyết. Lần này con cũng muốn chạy về kể cho Ba những gì con vừa có được, nhưng biết tìm Ba ở đâu... Rồi con bước chậm lại, hít sâu hơn, cảm nhận tiếng thở của gió, con thấy được dáng đi của Ba trong bước chân của chính mình, con nghe được tiếng Ba trong giọng nói của mình, và con cảm được niềm vui của Ba trong trái tim mình. Con biết rằng Ba vẫn ở đây, trong từng hơi thở, từng nhịp đập trái tim, trong từng bước chân chậm rãi, vững chắc, và kiên định.

Con trai của Ba



Contents

1 From Traditional Flood Modeling to a Hybrid Physics-AI Approach **1**
- 1.1 Flood modeling and challenges . . . 2
- 1.2 Hydrology and modeling approaches . . . 3
  - 1.2.1 Hydrological processes at a glimpse . . . 3
  - 1.2.2 Hydrometric observations and databases . . . 5
  - 1.2.3 Evolution of different hydrological modeling approaches . . . 9
  - 1.2.4 Data assimilation and Bayesian inference . . . 14
- 1.3 Artificial intelligence at a glance . . . 17
  - 1.3.1 From mind to machine: A natural extension of human intelligence . . . 17
  - 1.3.2 From linear regression to deep learning models . . . 19
  - 1.3.3 Backpropagation and gradient-based optimization . . . 24
  - 1.3.4 Leveraging or over-relying on AI? Understanding its potential and limitations 26
- 1.4 Toward a hybrid physics-AI approach within a multi-source data assimilation framework 27
  - 1.4.1 Research context overview and thesis objectives . . . 27
  - 1.4.2 Multi-source data at high spatio-temporal resolution . . . 29
  - 1.4.3 Mathematical foundations of the proposed framework . . . 30
  - 1.4.4 `smash`: Spatially distributed Modeling and ASsimilation for Hydrology platform 33

2 Signatures and Sensitivity-Based Multi-Criteria Variational Calibration for Distributed Hydrological Modeling **36**
- 2.1 Introduction . . . 37
- 2.2 Method . . . 38
  - 2.2.1 Overview of the hydrological model studied and optimization algorithms . . . 38
  - 2.2.2 Hydrological signatures . . . 40
  - 2.2.3 Hydrograph segmentation . . . 40
  - 2.2.4 Sensitivity analysis . . . 42
  - 2.2.5 Multi-criteria calibration . . . 43
- 2.3 Catchment information . . . 44
- 2.4 Sensitivity analysis and selection of signatures for model calibration . . . 46
- 2.5 Performance comparison of classical and signature-based calibration metrics . . . 47
  - 2.5.1 Spatially uniform calibrations with NSGA . . . 48
  - 2.5.2 Spatially distributed calibrations with VDA algorithm . . . 51
- 2.6 Conclusion . . . 54

3 Seamless Regionalization Using Neural Networks within a Differentiable High-Resolution Hydrological Model **56**
- 3.1 Introduction . . . 57
- 3.2 Forward-inverse algorithms within a regionalization framework . . . 59

3.2.1 Forward model with regionalization . . . 60
3.2.2 Calibration cost function accounting for multiple observation sites . . . 61
3.2.3 The variationnal data assimilation algorithm . . . 62
3.3 Data and numerical experiment . . . 64
3.3.1 Study area and experimental design . . . 64
3.3.2 Regional learning performance . . . 66
3.3.3 Interpretation of the learning process and inferred parameters . . . 73
3.4 Discussions . . . 76
3.5 Conclusion . . . 77

4 A Bayesian-Weighted Estimation Approach for Spatially Distributed Models under Equifinality Issues 79
4.1 Introduction . . . 80
4.2 Method . . . 80
4.2.1 Revisiting regionalization and calibration framework . . . 81
4.2.2 From low-dimensional to high-dimensional Bayesian-weighted estimates . . . 82
4.3 Results . . . 83
4.3.1 Numerical experiment . . . 83
4.3.2 Model performance and analysis . . . 84
4.4 Discussion . . . 86
4.5 Conclusion . . . 87

5 Inferring a Process-Parameterization Neural Network for Refining Internal Fluxes within a Spatialized Rainfall-Runoff Model 88
5.1 Introduction . . . 89
5.2 Modeling approach . . . 90
5.2.1 Rainfall-runoff model with regionalizable parameters and process-parameterization neural network . . . 90
5.2.2 Pixel-to-pixel flow routing of runoff with a partial differential equation and regionalizable parameters . . . 93
5.2.3 Learnable mappings for spatialized GR-like model and kinematic wave routing 93
5.2.4 Inverse problem and analysis of the hybrid physics-AI framework . . . 94
5.3 Data and experimental design . . . 94
5.4 Results and discussion . . . 96
5.4.1 Model performance analysis . . . 96
5.4.2 Towards learning hydrological behaviors . . . 100
5.4.3 Research perspectives and further discussion . . . 106
5.5 Conclusion . . . 107

6 Integrating Neural Networks into an Ordinary Differential Equation System Solved Using an Implicit Numerical Scheme 108
6.1 Introduction . . . 109
6.2 Modeling approach . . . 109
6.2.1 Hybrid continuous state-space structure for a regionalizable and gridded hydrological model . . . 110
6.2.2 Resolution of neural ODEs within an implicit Euler scheme . . . 112
6.2.3 Model training . . . 113
6.3 Case study and results . . . 114
6.3.1 Study area and experimental design . . . 114

6.3.2 Validation of the ODE solver . . . 116
6.3.3 Model performance and interpretation . . . 117
6.4 Conclusion . . . 123

7 Synthesis and Future Pathways in Physics-AI Integration **124**

References **127**

Appendices **143**

A Hydrological Evaluation Metrics and Properties **i**
A.1 Nash-Sutcliffe Efficiency . . . i
A.2 Normalized Nash-Sutcliffe Efficiency . . . i
A.3 Kling-Gupta Efficiency . . . ii
A.4 Modified Kling-Gupta Efficiency . . . ii
A.5 Mean Square Error . . . ii
A.6 Root Mean Square Error . . . iii

B Multi-Objective Optimization with Spatially Uniform Model Parameters **iv**
B.1 Pareto front . . . iv
B.2 Overview of GA and NSGA . . . v
B.3 Selection of an optimal solution from Pareto front . . . v

C Incorporating Regionalization Neural Networks into the Differentiable Hydrological Model **vii**

List of Figures

1.1 Sediment discharge in the Mediterranean Sea caused by flooding from Storm Alex in France and Italy. *Source: European Union, Copernicus Sentinel-2 imagery (2020).* . 2

1.2 Topographic map of the Seine River basin in France. *Source: Topographic data from NASA SRTM (2012).* 4

1.3 Illustration of the hydrological cycle (*retrieved and adapted from Kuang et al. (2024)*). 5

1.4 A typical rain gauge that measures total accumulated precipitation over a period. *Image Credit: ©Kolling (2005).* 6

1.5 Illustration of radar scanning volume (*retrieved from Ochoa-Rodriguez et al. (2019)*), showing the geometric relationships between range (r), beamwidth (θ), and pulse length (H). The range resolution (Δr) is determined by the pulse duration (τ) and the speed of light (C), with $\Delta r = \frac{H}{2} = \frac{C \cdot \tau}{2}$. The horizontal (azimuthal, ℓ_{Az}) and vertical (elevation, ℓ_{El}) extents of the radar beam, at a distance r, are given by $\ell_{Az} = r \cdot \theta$ and $\ell_{El} = r \cdot \theta$. 6

1.6 Real-time global precipitation map from NASA's IMERG dataset, providing half-hourly updates by integrating data from the GPM satellite constellation. *Image Credit: ©NASA GPM IMERG (2025).* 7

1.7 A typical river gauging station setup at the Pude gauging station, located in Nanteuil-Auriac-de-Bourzac, France, illustrating the use of water level sensors to measure stage, which is then used to estimate streamflow. *Image Credit: ©Père Igor (2021).* 8

1.8 Illustration of the SWOT satellite mission, capable of measuring surface water extent and elevation across the Ohio River basin, USA. *Image Credit: ©NASA/JPL-Caltech (2024).* 9

1.9 Comparison of three fundamental hydrological modeling approaches: (a) Black-box models represent purely data-driven approaches that establish direct input-output relationships without explicit process representation, requiring extensive historical data for training; (b) White-box models incorporate physical principles and mathematical equations to simulate hydrological processes based on measurable physical parameters; (c) Gray-box models use simplified conceptual representations of watershed processes through interconnected storage elements (soil, groundwater, surface) governed by conceptual parameters that require calibration. 10

1.10 Illustration of a perceptron, where each node represents a neuron, and each line represents the connection between two neurons via the linear operators applied to the trainable parameters $\boldsymbol{W}$ and $\boldsymbol{b}$. 19

1.11 Illustration of a multilayer perceptron with a single hidden layer, where each node represents a neuron, and each line represents the connection between two neurons via the linear operators applied to the weight $\boldsymbol{W}_i$ and bias $\boldsymbol{b}_i$. 20

1.12 Comparison of a feedforward neural network (top) and an RNN (bottom), where the RNN introduces recurrent connections to capture temporal dependencies. The arrow with a square in the middle signifies a recursive relationship, indicating that the current hidden state is influenced by both the current input and the hidden state from the previous time step. . . . 22
1.13 Illustration of a convolution operation applied to an input matrix of 3×3, using a filter of 2×2, with valid padding and a stride of 1. . . . 23
1.14 Flow direction (left) and flow accumulation (right) maps, at a relatively high spatial resolution of 1 km^2, covering all of Metropolitan France. . . . 29
1.15 Schematic representation (based on `smash` documentation: Section Forward & Inverse Problems) of the operator composition within the forward model. . . . 31
1.16 Schematic representation (based on `smash` documentation: Section Forward & Inverse Problems) of the operator composition within the forward model, incorporating a learnable mapping via neural networks. . . . 32
1.17 The `smash` build system framework, presented in Colleoni, Huynh, et al. (2025), consists of: (0) Source files written in Fortran and Python; (1) Fortran sources are processed by Tapenade to generate the adjoint code, and wrapped using *f90wrap* to create Python interfaces (*f90wrap-itf*). The *F2PY* tool is then used to generate a Fortran/C/Python binding from the wrapped interfaces; (2) The original Fortran sources, adjoint code, and *f90wrap-itf* are compiled with appropriate compilers to produce a binary Python/C extension module. Finally, in the installation step; (3) The Python module is assembled, combining the original Python sources with the *f90wrap-itf* and the compiled binary Python/C extension module, making the high-performance Fortran code accessible from Python. . . . 34
1.18 `smash` documentation home page accessible at `https://smash.recover.inrae.fr`. . 35

2.1 Flowchart of the multi-criteria calibration process using hydrological signatures. The different cost functions are denoted by j_d, j_c and j_f, while the corresponding optimal weights are denoted by δ_d, δ_c and δ_f. The notations used in the cost function will be explained in 2.2.5. . . . 38
2.2 Spatially distributed hydrological model studied in the `smash` platform; (left) Model fields from top to bottom: meteorological inputs, conceptual parameters, internal and output flux maps; (right) Pixel scale and pixel-to-pixel flow operators of the model structure with 6 parameters, where $\boldsymbol{\theta}(x) \equiv (c_i(x), c_p(x), c_t(x), c_l(x), k_{exc}(x), l_{l_r}(x))^T$. 39
2.3 Example of a flood event observed at the outlet gauge of the Aude River in France. . 40
2.4 Spatial distribution of 141 catchments of the PICS dataset, consisting of 23 outlet gauges and 118 upstream gauges on the map of France with four regions denoted by different colors. . . . 45
2.5 Comparison, with spatially uniform parameters, of calibration and validation metrics (optimal fit for $cost = 0$) for three optimization approaches (CSOO, SSOO, SMOO) by constraining $1 - NSE$ in case of global algorithms (SBS or NSGA). From left to right: calibration (Cal), temporal validation (T_Val), spatial validation (S_Val) and spatio-temporal validation (S-T_Val). . . . 50
2.6 Comparison, with spatially uniform parameters, of calibration and validation metrics (optimal fit for $cost = 0$) for three optimization approaches (CSOO, SSOO, SMOO) by constraining $1 - KGE$ in case of global algorithms (SBS or NSGA). From left to right: calibration (Cal), temporal validation (T_Val), spatial validation (S_Val) and spatio-temporal validation (S-T_Val). . . . 50

2.7 Analysis of spatially uniform calibrated parameters over the whole catchment sample. In each scatterplot, the first column present the distribution of a parameter for 3 optimization strategies (CSOO, SSOO, SMOO) using j_d^{NSE}, whereas the strategies in the second column use j_d^{KGE} as the dominant (or constrained) objective function. 51

2.8 Comparison, with spatially distributed parameters, of calibration and validation metrics (optimal fit for $cost = 0$) for two optimization approaches (CSOO, SSOO) by constraining $1-NSE$ in case of distributed calibration. From left to right: calibration (Cal), temporal validation (T_Val), spatial validation (S_Val) and spatio-temporal validation (S-T_Val). 52

2.9 Comparison, with spatially distributed parameters, of calibration and validation metrics (optimal fit for $cost = 0$) for two optimization approaches (CSOO, SSOO) by constraining $1-KGE$ in case of distributed calibration. From left to right: calibration (Cal), temporal validation (T_Val), spatial validation (S_Val) and spatio-temporal validation (S-T_Val). 52

2.10 Comparison of scoring metrics computed on 111 events picked from 23 outlet gauges on the calibration period 2006-2013 for the five optimization strategies, by constraining $1 - NSE$ (left) and $1 - KGE$ (right), optimal fit for $score = 1$. 53

2.11 Spatially distributed optimal parameters for the Argens River basin, obtained by minimizing $j_d^{KGE}/2 + j_f^{Epf}/2$. 53

3.1 Flowchart of the forward-inverse algorithm used in HDA-PR. The forward hydrological model is a gridded model (spatio-temporal regular grid at 1 km^2 and 1 h) using GR operators (Perrin et al., 2003). 59

3.2 Map of France highlighting the ArcMed study area, covering 150,000 km^2 (100,000 km^2 excluding sea), comprising 126 catchments categorized as 38 catchments located upstream, 33 intermediate catchments, 24 catchments positioned downstream, and 31 independent catchments, representing a total drainage area of 26,000 km^2. 64

3.3 Observed and simulated discharges (in m^3/s) at several locations during the validation period P2, using three multi-gauge regional calibration methods (columns) within the first calibration setup, where upstream catchments are used for calibration on P1. The first row hence corresponds to temporal validation assessments, while the subsequent three rows correspond to spatio-temporal validation assessments. 66

3.4 Boxplots of NSE scores (optimal value = 1) across calibration and validation catchments for both calibration setups which are upstream (top) and downstream (bottom), compared to reference solutions obtained by local calibration methods (Uniform (loc) and Distributed (loc)). From left to right: results are displayed for calibration catchments on period P1 ("Cal"), validation catchments on P1 for spatial validation ("S_Val"), calibration catchments on P2 for temporal validation ("T_Val"), and validation catchments on P2 for spatio-temporal validation ("S-T_Val"). The numbers in parentheses indicate the count of catchments included in each boxplot. 67

3.5 Boxplots of KGE scores (optimal value = 1) across calibrated (with NSE cost function) and validation catchments for both calibration setups (upstream (top) and downstream (bottom)), compared to reference solutions obtained by local calibration methods (Uniform (loc) and Distributed (loc)). From left to right: results are displayed for calibration catchments on period P1 ("Cal"), validation catchments on P1 for spatial validation ("S_Val"), calibration catchments on P2 for temporal validation ("T_Val"), and validation catchments on P2 for spatio-temporal validation ("S-T_Val"). The numbers in parentheses indicate the count of catchments included in each boxplot. 68

3.6 Comparison of NSE scores (optimal value = 1) across validation catchments categorized by their nature (upstream, downstream, intermediate, or independent) for both calibration setups (upstream (top) and downstream (bottom)). From left to right: results are displayed for validation catchments on period P1 for spatial validation ("S_Val") and validation catchments on P2 for spatio-temporal validation ("S-T_Val"). The numbers in parentheses indicate the count of catchments included in each boxplot. 69

3.7 Comparison of KGE scores (optimal value = 1) across validation catchments categorized by their nature (upstream, downstream, intermediate, or independent) for both calibration setups (upstream (top) and downstream (bottom)). From left to right: results are displayed for validation catchments on period P1 for spatial validation ("S_Val") and validation catchments on P2 for spatio-temporal validation ("S-T_Val"). The numbers in parentheses indicate the count of catchments included in each boxplot. 70

3.8 Comparison of NSE improvement rates in spatial and spatio-temporal validation for HDA-PR with ANN mapping versus uniform (left) and multi-linear (right) mappings, utilizing models calibrated with upstream catchments. Results are shown across validation catchments positioned downstream, intermediate, and independent for spatial validation on period P1 (top) and spatio-temporal validation on period P2 (bottom). The improvement rate at a catchment i is calculated as $r^i = \frac{NSE^i_{\text{ANN}} - NSE^i_{\square}}{|NSE^i_{\square}|}$, where $\square$ represents either multi-linear or uniform mapping. 71

3.9 Relative error (optimal value = 0) of three flood event signatures—flood runoff coefficient (Erc), flood flow (Eff), and peak flow (Epf)—for both calibration setups: upstream (top) and downstream (bottom). The evaluation is based on a total of 1,737 flood events, with 1,124 selected during the calibration period P1 and 613 selected during the validation period P2, across 126 catchments. From left to right: results are displayed for calibration catchments on P1 ("Cal"), validation catchments on P1 for spatial validation ("S_Val"), calibration catchments on P2 for temporal validation ("T_Val"), and validation catchments on P2 for spatio-temporal validation ("S-T_Val"). The numbers in parentheses indicate the count of flood events included in each boxplot. 72

3.10 Spatially distributed cost gradients $\nabla_{\boldsymbol{\theta}} J\left(\boldsymbol{\theta}^*_{\square}\right)$ to conceptual hydrological parameters used in the optimization process (of $\boldsymbol{\rho}$ using $\nabla_{\boldsymbol{\rho}} J = \nabla_{\boldsymbol{\theta}} J . \nabla_{\boldsymbol{\rho}} \boldsymbol{\theta}$) showed for downstream calibration (for the sake of visibility) at initial points $\boldsymbol{\theta}^*_{\text{Multi-linear}} \coloneqq \mathcal{P}(., \boldsymbol{\rho}^*_{\text{Multi-linear}})$ and $\boldsymbol{\theta}^*_{\text{ANN}} \coloneqq \mathcal{N}n(., \boldsymbol{\rho}^*_{\text{ANN}})$, where $\boldsymbol{\rho}^*_{\text{Multi-linear}}$ (resp. $\boldsymbol{\rho}^*_{\text{ANN}}$) is the background value of the multi-linear regression (resp. ANN) mapping. Note that the cost gradient values are obtained for the active cells within the gauged partition, i.e., cells contributing to the simulated discharge at gauges (downstream here) where the observation cost function J is evaluated. 73

3.11 The descent of the cost function $J = 1 - NSE$ for both Multi-linear and ANN approaches across both calibration setups. 74
3.12 Analysis of input descriptors and output model parameters for two regionalization methods (Multi-linear and ANN) across both calibration setups (upstream and downstream): (a) Spatial distribution of physical descriptors (d_1-d_7), details provided in Table 1.1; (b) Spatial distribution of calibrated hydrological parameters (c_p, c_t, k_{exc}, l_{l_r}) with μ and σ indicating their spatial average and standard deviation over the active cells within the spatial domain Ω; (c) Linear correlation between descriptors and parameters over the active cells within the spatial domain Ω. 75

4.1 Flowchart of the regionalization framework using Bayesian-weighted estimate of background values. 81
4.2 Performance of all calibration and regionalization methods when calibrated on P1 (upper sub-figure) and on P2 (lower sub-figure). BM2R uses the first guess obtained by Low-Dimensional Bayesian estimation (LDB-FG), while the first guess of M2R is obtained by the SBS algorithm (SBS-FG), which also represents the solution of the regionalization method with lumped parameters (UR); n denotes the number of study catchments. 84
4.3 Normalized temporal stability over the periods P1 and P2 of the optimal distributed parameters $\frac{\hat{\theta}_k^{(P2)} - \hat{\theta}_k^{(P1)}}{u_k - l_k}, k \in [1..N_\theta]$, for the three regionalization approaches using physical descriptors. 86

5.1 Hybrid physics-AI framework, applied to the spatially distributed GR-like and kinematic wave model, involving a pair of process-parameterization and regionalization neural networks. The pair of neural networks is used to (i) correct internal fluxes (using neutralized atmospheric data) and (ii) estimate the model parameters (using physical descriptors), with their weights optimized through high-dimensional optimization algorithms using an adjoint model to obtain accurate gradients of the cost function. 91
5.2 Study areas used for evaluation. The first area consists of 235 non-nested catchments in France, while the second area includes 21 catchments in a multiple-catchment setup in the Eastern Mediterranean region, representing contrasting hydrological conditions. 94
5.3 Maps of seven physical descriptors in the MedEst area at a resolution of 0.01° in the WGS 84 projection, where μ and σ represent the spatial average and standard deviation for each descriptor. 95
5.4 Comparison of streamflow simulation across representative small, medium, and large catchments randomly selected among the 235 catchments in France during the validation period. 96
5.5 Model performance comparison of local calibration methods across 235 catchments in France. The two boxes in lighter colors represent the classical GR models (GR.U and GR.D), while the two boxes in darker colors represent the ϕ_1-hybrid models (GRNN.U and GRNN.D). The evaluation is based on simulated discharges over the calibration and validation period, using NSE, KGE, and RMSE metrics. 97
5.6 Comparison of model performance in simulating flood event signatures, presented as the cumulative distribution function (CDF) of the relative error (RE) between observed and simulated values for peak flow (Epf), runoff coefficient (Erc), flood flow (Eff), and baseflow (Ebf). The evaluation is based on 2,718 flood events across 235 catchments during the validation period (08/2013–07/2019). 98

5.7 Comparison of multi-catchment regionalization performance for different methods. The evaluation is based on NSE scores computed over three periods: the 3-year calibration period (excluding a 1-year warmup), the first 18 months, and the last 18 months of the 3-year validation period for 11 calibration catchments and 10 validation catchments. The numbers in parentheses on the boxplots indicate the total number of samples evaluated, with the validation period split into two 18-month periods (sample counts are doubled for all catchments). 99

5.8 (a) Comparison of NSE and RMSE metrics computed for 143 flood events detected in the validation period across 21 catchments in the MedEst region. (b) Comparison of streamflow simulations for representative small, medium, and large catchments during several flood events selected in the validation period. 100

5.9 Maps of spatio-temporal average flux corrections $\overline{\overline{g\left(\boldsymbol{f}_q\right)}^{x\in\Omega_j}}^{t}$, where $j = 1..N_g$, for the $N_g = 235$ catchments, obtained through local calibrations of spatially uniform parameters with the hybrid model structure (GRNN.U). The function $g(.)$ represents the transformation applied to the neural network output $\boldsymbol{f}_q$, which may differ depending on the specific flux being corrected. Red indicates corrections that tend to increase the current flux, while blue indicates corrections that reduce it, with white representing minimal or no effective correction. 101

5.10 (a) Spatially averaged flux correction time series $\overline{g\left(\boldsymbol{f}_q(t)\right)}^{x\in\Omega_j}$ for three randomly selected catchments from the set of 235. These are obtained through local calibrations of spatially uniform parameters with the hybrid model structure (GRNN.U). (b) Boxplot of spatio-temporal average flux corrections $\overline{\overline{g\left(\boldsymbol{f}_q\right)}^{x\in\Omega_j}}^{t}$, where each boxplot represents 235 catchment-specific spatio-temporal averages. 102

5.11 Visualization of flux corrections in the MedEst region obtained through regional calibration of spatially uniform parameters with the hybrid model (GRNN.U): (a) Spatial average of infiltrating flux correction $\overline{1 + f_{q,1}(t)}^{x}$, original and corrected infiltrating rainfall $\overline{P_s(t)}^{x}$, $\overline{\tilde{P}_s(t)}^{x}$; (b) Maps of time-averaged infiltrating flux correction $\overline{1 + f_{q,1}(x)}^{t}$, original and corrected infiltrating rainfall $\overline{P_s(x)}^{t}$, $\overline{\tilde{P}_s(x)}^{t}$, where μ and σ represent the spatial average and standard deviation. 103

5.12 Comparison of mass fluxes affecting water balance in interannual mean at basin scale over the 235 catchment set, for the classical GR model (GR.U) and the ϕ_1-hybrid model (GRNN.U) using uniform mapping "U" for conceptual parameters $\boldsymbol{\theta}$, in local calibration. The x-axis shows 235 catchments, sorted by their average precipitation. Grey dots represent the interannual averages of precipitation P, actual evapotranspiration E_s, exchange flux F, and lateral discharge Q_{lat} over the calibration period for the classical model. Red dots and lines represent increases according to the hybrid model, while blue ones indicate decreases. For cases where $F < 0$, red indicates a larger magnitude in F for the hybrid model (more negative), while blue indicates a lesser magnitude (closer to zero). 104

5.13 Hydrological interpretation of the non-linear response surface obtained using the learned flux correction neural network for infiltrating rainfall $\tilde{P}_s = (1 + f_{q,1})P_s$, plotted with a production capacity of $c_p = 15$ mm and neutralized evapotranspiration of $E_n = 0$ mm: (a) Original and corrected net rainfall P_r for different levels of the production state h_p; (b) Response surface of original and corrected net rainfall P_r as a function of both the production state h_p and neutralized rainfall P_n. 105

6.1 The Aude River basin, located in southern France, consists of 25 sub-catchments, including 12 upstream catchments (red) and 13 downstream catchments (gray). . . . 114
6.2 Maps of seven physical descriptors in the Aude River basin at a resolution of 0.01° in the WGS 84 projection, where μ and σ represent the spatial average and standard deviation for each descriptor. . . . 115
6.3 Comparison of normalized production state (top), normalized transfer state (middle), and simulated streamflow (bottom) at the most downstream gauge, obtained by algebraic resolution (GR.U) and numerical resolution (ODE.U) using the same lumped conceptual parameters and initial states. . . . 116
6.4 Maps of the normalized production state (h_p/c_p, top) and transfer state (h_t/c_t, bottom) at the final time step, as simulated by GR.U (left), ODE.U (middle), and the corresponding bias (right) computed as the difference between the ODE.U and GR.U states. For each map, μ and σ denote the spatial average and standard deviation. . . . 117
6.5 Comparison of model performance for different methods. The NSE scores are computed to evaluate: (i) calibration performance (scores computed over P1 for the 12 upstream gauges), (ii) spatial validation (scores computed over P1 for 13 downstream gauges), (iii) temporal validation (scores computed over P2 for 12 upstream gauges), and (iv) spatio-temporal validation (scores computed over P2 for downstream gauges). 118
6.6 Comparison of flood simulation performance across different methods for an upstream gauge (left) and a downstream gauge (right). . . . 119
6.7 Estimated conceptual parameters across different regionalization methods. The calibrated parameters are $\boldsymbol{\theta}(x) = (c_p(x), c_t(x), k_{exc}(x), a_{kw}(x), b_{kw}(x))^T$ with μ and σ denoting their spatial average and standard deviation. . . . 120
6.8 Comparison of the spatial average of the normalized production state $\overline{h_p(t)/c_p(t)}^x$ and transfer state $\overline{h_t(t)/c_t(t)}^x$ during the validation period P2, across the three model structures (GR, ODE, NODE), each employing an MLP for regionalization. . . . 121
6.9 Maps of the time-averaged normalized production state $\overline{h_p(x)/c_p(x)}^t$ and transfer state $\overline{h_t(x)/c_t(x)}^t$ over the validation period P2, for the three model structures (GR, ODE, NODE), each employing an MLP for regionalization. For each map, μ and σ denote the spatial average and standard deviation. . . . 122
6.10 High-pass filtered lateral discharge from the most downstream gauge, using a 3-day cutoff frequency, to emphasize flood events during the validation period P2. The results are shown for three model structures (GR, ODE, NODE), each employing an MLP for regionalization. . . . 122

B.1 Illustration of Pareto front in terms of objective space. . . . v

C.1 The architecture of the ANN consists of three hidden layers followed by the ReLU activation function and an output layer that uses the Sigmoid activation function in combination with a scaling function. In this particular case, we have $N_D = 7$, $N_L = 4$ and $N_\theta = 4$. The total number of trainable parameters is calculated in Table C.1. . . . vii

List of Tables

1.1 Multi-source data from national open-source databases over Metropolitan France, including atmospheric forcings $\boldsymbol{\mathcal{I}} = (\boldsymbol{P}, \boldsymbol{E})$, the flow direction map $\mathcal{D}_\Omega$, physical descriptors $\boldsymbol{D} = (d_1, ..., d_7)$ for regionalization, and discharge time series $\boldsymbol{Q^*}$. 30

2.1 List of all studied continuous signatures. 40

2.2 List of all studied flood event signatures. 41

2.3 Overview of 23 outlet gauges in the PICS dataset. The columns (left to right) represent the gauge code, river name, surface area, missing rate of rainfall (resp. discharge) at the outlet gauge (for the period 2006–2019), and the number of upstream gauges. 45

2.4 Relative error between simulated and observed signatures of the same model structure calibrated either with $1-NSE$ or $1-KGE$ by SBS algorithm for global optimization. The values (in the form of . [., .]) in each case represent respectively the median, mean and standard deviation of a signature over gauged catchments. 46

2.5 Median across gauged catchments of first- (respectively, total-) order variance-based sensitivity indices of the studied signatures to the model parameters. 47

2.6 Calibration, temporal, spatial, and spatio-temporal validation metrics for calibration methods with (a) spatially uniform parameters (CSOO, SSOO, SMOO), and (b) spatially distributed parameters, optimal fit for $cost = 0$. The mean of calibration and validation cost values for different objective functions are displayed for each calibration metric. 49

2.7 Uniform optimal parameters calibrated by SBS algorithm with 4 calibration metrics for each catchment on its outlet gauge. The values (in the form of . [., .]) in each case represent respectively the median, mean and standard deviation of the optimal parameter values over all catchments of the dataset. 53

2.8 Analysis of spatially distributed parameter sets of the models corresponding to 4 calibration metrics. First, spatial median ($\tilde{.}$), average ($\bar{.}$) and standard deviation ($\sigma_.$) for each parameter field are calculated for each catchment, then their median, mean and standard deviation over all catchments are represented in the form of . [., .]. . . 54

4.1 The optimal parameters obtained by different methods for each calibration period. Spatially distributed parameters are represented as the median (mean, std). The rows in italic and smaller text present the spatially uniform first guess obtained by the SBS algorithm (SBS-FG) used for M2R, and the Low-Dimensional Bayesian estimation (LDB-FG) used for BM2R. 85

5.1 Median (mean; standard deviation) of the calibrated hydrological parameters across the study area of 235 catchments. 102

6.1 Summary of evaluated models. The notation consists of two parts, separated by a dot. The first part describes the model structure: GR denotes the GR4 structure (Perrin et al., 2003), which uses an algebraic formulation for state updates; ODE denotes the continuous GR4-like state-space model (Santos et al., 2018); and NODE denotes the neural ODE structure that incorporates a neural network into the ODEs. The second part indicates the mapping used to constrain the conceptual parameters: U represents lumped parameters without constraints; while MLP or CNN indicate regionalization mappings based on MLP or CNN. All models include a kinematic wave routing over a flow direction grid. 115

C.1 Number of parameters of the ANN where $N_D = 7$, $N_L = 4$ and $N_\theta = 4$. viii

List of Abbreviations

API . . . . . Application Programming Interface
AI . . . . . Artificial Intelligence
ANN . . . . . Artificial Neural Network
AD . . . . . Automatic Differentiation
BFGS . . . . . Broyden–Fletcher–Goldfarb–Shanno algorithm
CNN . . . . . Convolutional Neural Network
DA . . . . . Data Assimilation
DL . . . . . Deep Learning
FCNN . . . . . Fully Connected Neural Network
GA . . . . . Genetic Algorithm
GPU . . . . . Graphics Processing Unit
HRU . . . . . Hydrological Response Unit
KGE . . . . . Kling-Gupta Efficiency
L-BFGS-B . . . . . Limited-memory BFGS algorithm with Bound constraints
LSTM . . . . . Long Short-Term Memory
MAE . . . . . Mean Absolute Error
ML . . . . . Machine Learning
MLP . . . . . Multilayer Perceptron
MSE . . . . . Mean Squared Error
NSE . . . . . Nash-Sutcliffe Efficiency
NSGA . . . . . Non-dominated Sorting Genetic Algorithm
ODE . . . . . Ordinary Differential Equation
PDE . . . . . Partial Differential Equation
PDF . . . . . Probability Density Function
PINN . . . . . Physics-Informed Neural Network
ReLU . . . . . Rectified Linear Unit
RMSE . . . . . Root Mean Squared Error
RNN . . . . . Recurrent Neural Network
SBS . . . . . Step-By-Step optimization
TanH . . . . . Hyperbolic Tangent
TPU . . . . . Tensor Processing Unit
VDA . . . . . Variational Data Assimilation

Chapter 1

From Traditional Flood Modeling to a Hybrid Physics-AI Approach

Contents

1.1 Flood modeling and challenges . . . 2
1.2 Hydrology and modeling approaches . . . 3
1.2.1 Hydrological processes at a glimpse . . . 3
1.2.2 Hydrometric observations and databases . . . 5
1.2.2.1 Precipitation monitoring . . . 5
1.2.2.2 River network measurements . . . 8
1.2.3 Evolution of different hydrological modeling approaches . . . 9
1.2.3.1 Empirical models: From rational method and unit hydrograph to statistical and data-driven approaches . . . 10
1.2.3.2 Physically-based models . . . 11
1.2.3.3 Conceptual hydrological representation: From lumped to fully distributed models . . . 13
1.2.4 Data assimilation and Bayesian inference . . . 14
1.3 Artificial intelligence at a glance . . . 17
1.3.1 From mind to machine: A natural extension of human intelligence . . . 17
1.3.2 From linear regression to deep learning models . . . 19
1.3.2.1 Linear regression . . . 19
1.3.2.2 Multilayer perceptron . . . 20
1.3.2.3 Recurrent neural networks . . . 21
1.3.2.4 Convolutional neural networks . . . 22
1.3.3 Backpropagation and gradient-based optimization . . . 24
1.3.4 Leveraging or over-relying on AI? Understanding its potential and limitations 26
1.4 Toward a hybrid physics-AI approach within a multi-source data assimilation framework . . . 27
1.4.1 Research context overview and thesis objectives . . . 27
1.4.2 Multi-source data at high spatio-temporal resolution . . . 29
1.4.3 Mathematical foundations of the proposed framework . . . 30
1.4.4 `smash`: Spatially distributed Modeling and ASsimilation for Hydrology platform . . . 33

This chapter aims to provide an overview of the context for this thesis. It starts by highlighting the modeling challenges, then revisits traditional hydrological approaches and presents artificial intelligence (AI) methods. Finally, it introduces the research context and direction towards hybrid AI approaches within the hydrological modeling process.

1.1 Flood modeling and challenges

Floods, particularly flash floods triggered by heavy and localized rainfall, cause a rapid rise in river discharge within minutes to several hours, ranking among the most destructive natural hazards and inflicting immense economic, social, and environmental damage worldwide (e.g., Vinet (2008), B. He et al. (2018), and Alipour et al. (2020)). With their sudden onset and high magnitude, such events typically provide minimal opportunity for warning and mitigation (Špitalar et al., 2014). The prediction of flash floods is an essential challenge, especially in the context of climate change, urbanization, and land-use changes (Winsemius et al., 2016). Throughout history, flash floods have caused substantial destruction to communities, infrastructure, and ecosystems globally. In European Mediterranean regions, extreme flash floods pose a serious risk to life and often affect ungauged watersheds that lack measurement instruments, making them a poorly documented phenomenon (Gaume et al., 2009). For example, the Alex storm of 2020 severely impacted southeastern France and north-western Italy, leading to catastrophic flooding that destroyed homes, roads, bridges, and resulted in significant casualties (Figure 1.1). Such extreme weather events highlight the urgency of

Figure 1.1 – Sediment discharge in the Mediterranean Sea caused by flooding from Storm Alex in France and Italy. *Source: European Union, Copernicus Sentinel-2 imagery (2020).*

improving flood prediction and mitigation measures since economic damages from flash floods are enormous, affecting agriculture, transportation networks, and water supply systems. In the context of climate change, spatio-temporal patterns are altered and become increasingly difficult to predict (Hettiarachchi et al., 2018; Kuang et al., 2024). Given these severe consequences, efficient flash

flood modeling and water management strategies are crucial.

The fundamentals of flood modeling involve predicting river discharge and water levels to provide early warnings, enabling authorities and communities to take preventive actions. This can be achieved through hydrological modeling approaches that utilize meteorological data and real-time observations for discharge prediction. Accurate flood modeling is generally influenced by several key factors:

- **Hydrological models:** These models are typically rainfall-runoff models that simulate watershed hydrology to estimate runoff and river discharge;
- **Meteorological data:** Spatially distributed rainfall, temperature and snow data obtained from radar, satellite observations, etc., are used as input for hydrological models;
- **Real-time observations:** Ground-based measurements of river levels, soil moisture, etc., are integrated through data assimilation techniques to improve model realism and accuracy.

Despite significant research and developments in these topics, flood modeling persistently faces major challenges, including:

- **Sparse data:** The lack of observational data, particularly the absence of discharge measurements for many ungauged rivers, makes it challenging to perform the (regional) calibration of hydrological models and to achieve accurate spatio-temporal predictions.
- **Model complexity:** This should be tailored to the physical complexity of the system and the availability of data. Describing numerous physical processes using a physics-based approach typically requires extensive data, but this does not always translate to improved predictive performance. Conceptual hydrological models, on the other hand, offer a simplified representation of key physical mechanisms. Although conceptual models may face challenges related to realism and parameterization, they can still achieve high performance after calibration. However, a challenge with conceptual models is incorporating physical descriptors for spatialization or regionalization constraints.
- **Data uncertainties:** Input data and observations are prone to measurement errors and may suffer from insufficient sampling and resolution, particularly for phenomena with high spatio-temporal variability, such as heavy rainfall events that trigger floods.

The following section provides basic concepts of hydrological processes, observational data sources, and different modeling approaches for flood simulation, as well as the challenges associated with them.

1.2 Hydrology and modeling approaches

1.2.1 Hydrological processes at a glimpse

Water has been present on Earth for billions of years, continuously circulating through the environment across different epochs and environmental changes (Wilde et al., 2001). Water moves through the atmosphere, biosphere and lithosphere in three phases: liquid (rain, rivers, lakes, oceans), gas (water vapor, clouds) and solid (snow, ice, glaciers, permafrost). The total amount of water on the planet remains constant, forming a closed system water cycle. Two main forces control water dynamics: solar energy, which powers evaporation and atmospheric water transport, and Earth's gravity, which retains water within the atmosphere and drives precipitation, runoff, and groundwater flow. While many of these principles have been observed and documented by humans over centuries, fully understanding and predicting the behavior of the water cycle remains a challenge due to its complexity and variability across scales (Sivapalan, 2003; Oki et al., 2006).

Hydrology is the discipline that studies the water cycle, which is composed of complex interconnected physical processes, aiming to quantify water storage, fluxes, and quality within the critical zone of the Earth system (top of the canopy to groundwater bottom).

The concept of a catchment, also known as watershed or drainage basin, is central to hydrological science. A topographic catchment is a geographical area defined by a watershed divide, which collects and drains surface water to a specific outlet, such as a river or lake (Figure 1.2). It can

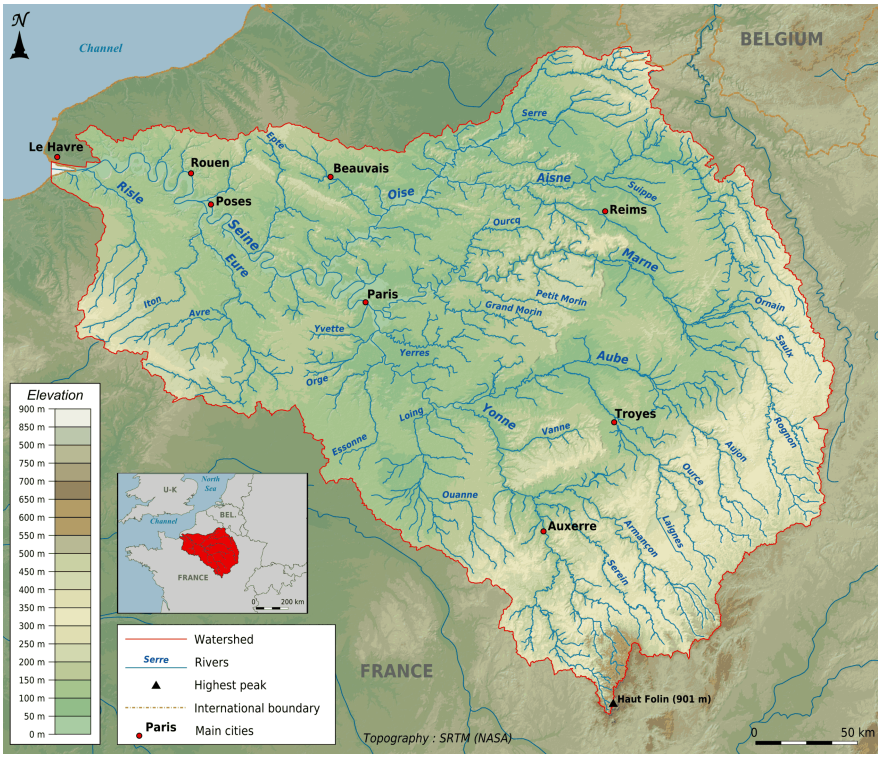


Figure 1.2 – Topographic map of the Seine River basin in France. *Source: Topographic data from NASA SRTM (2012).*

be characterized by various factors including its size, slope, shape, soil type, and land use. These factors influence the catchment's hydrological response to climatic events such as rainfall, drought, and snowmelt, and are crucial for catchment classification based on hydrological similarity (Wagener et al., 2007). Understanding these characteristics helps in assessing how different catchments behave under similar climatic conditions, which is essential for effective water management and flood prediction (Burn, 1997).

Water travels through a succession of recurring states that constitute the hydrological cycle. Surface hydrology specifically studies the water cycle on continental surfaces, focusing on exchanges between soil, biosphere, and atmosphere, as well as the lateral redistribution of water among different components of the system, which includes critical phenomena such as runoff, river flows, and flooding. The hydrological cycle consists of a variety of interconnected processes, including evaporation from water bodies and bare soil, evapotranspiration (evaporation and transpiration) from vegetated areas, the transport of water vapor through the atmosphere, the formation and dynamics of cloud droplets, the mechanisms responsible for precipitation (rain and snow), as well as the movement and alteration of soil moisture in the unsaturated zone, including root activity, surface runoff, river flow, and groundwater movement (Oki et al., 2006), as illustrated in Figure 1.3. **Precipitation**, in the form of rain or snow, initiates the cycle by delivering water to the Earth's surface. This water can follow various pathways:

— **Surface Runoff**: Water flows over the land as surface runoff, moving towards rivers, lakes, and ultimately the ocean;

— **Infiltration and percolation**: A portion of the precipitation infiltrates the soil, replenish-

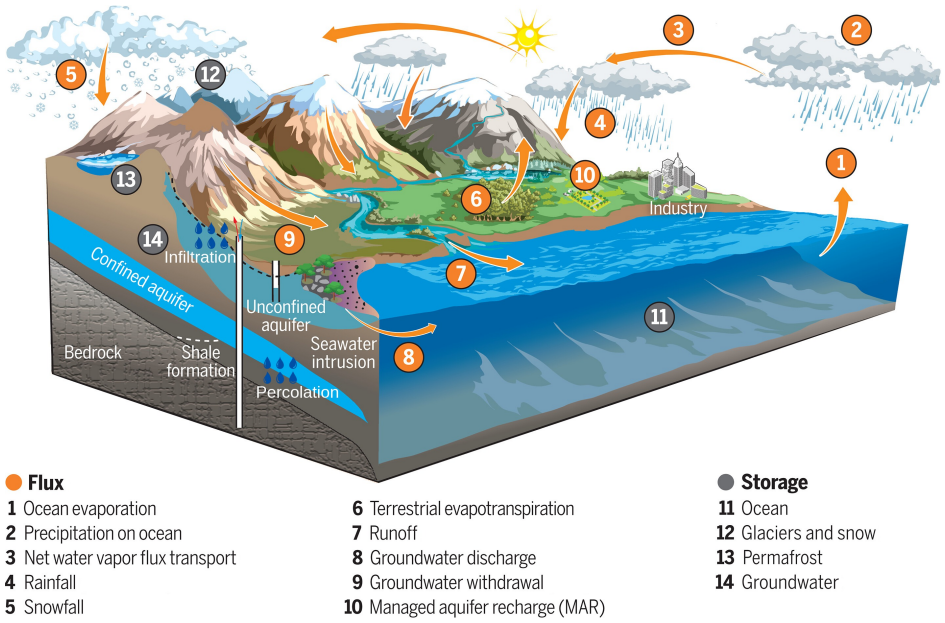


Figure 1.3 – Illustration of the hydrological cycle (*retrieved and adapted from Kuang et al. (2024)*).

ing soil moisture and percolating deeper to recharge groundwater reserves;

— **Evapotranspiration**: Water from soil and surface water bodies evaporates into the atmosphere, while plants contribute to this process through transpiration. Moreover, the vast majority of evaporation occurs from the oceans, driven by solar energy. These processes collectively transform liquid water into vapor, returning it to the atmosphere and sustaining the atmospheric moisture necessary for precipitation;

— **Groundwater runoff**: Groundwater moves laterally through aquifers, eventually discharging into streams, rivers, and the ocean, contributing to baseflow in rivers during dry periods;

— **Permafrost and ice**: In colder regions, water exists as ice and snow, which can melt and contribute to runoff, or remain frozen as permafrost, influencing soil moisture dynamics.

These interconnected processes highlight the dynamic nature of the hydrological cycle, emphasizing the importance of understanding each component to effectively model and manage water resources.

1.2.2 Hydrometric observations and databases

Accurate hydrological modeling relies on comprehensive observational data networks that monitor the key components of the water cycle. These networks provide essential measurements for understanding hydrological processes, as well as model calibration and validation. Various types of observations, including temperature, humidity, and soil moisture, are crucial for hydrological studies. However, this section focuses specifically on precipitation and streamflow measurements.

1.2.2.1 Precipitation monitoring

The traditional approach for direct precipitation measurement is to use **rain gauge** networks, which measure precipitation at point locations (Figure 1.4). These instruments typically consist of containers that collect rainwater, with the collected volume used to calculate the amount of precipitation over a given time period. However, rain gauges can produce significant measurement uncertainty under heavy wind or rainfall and vary greatly in density across regions, with developed areas typically having more extensive coverage than remote or developing regions (Kidd et al., 2017).

Figure 1.4 – A typical rain gauge that measures total accumulated precipitation over a period. *Image Credit: ©Kolling (2005).*

Weather radar systems complement gauge networks by providing spatially continuous precipitation estimates over large areas at finer resolutions, typically 1 km spatially and hourly to sub-hourly temporally. These systems estimate rainfall by emitting electromagnetic pulses and detecting backscattered signals from precipitation particles, with reflectivity depending on particle size, type, and distribution. This reflectivity is used to calculate rainfall rates at a distance. Unlike rain gauges, radars provide instantaneous volumetric measurements, which is crucial for real-time rainfall monitoring and forecasting. However, radar measurements require significant adjustment (e.g., gauge-based calibration) to account for their complex error structure, including errors due to attenuation, vertical profile of reflectivity, beam blockage, and the conversion of reflectivity to rainfall rate, which can lead to underestimation or misplacement of precipitation, especially in heavy rain or areas with challenging terrain (Berne et al., 2013). The radar-rain gauge merging measurement methods effectively enhance precipitation estimation accuracy by combining the high temporal resolution of radar with the ground truth reliability of rain gauges, improving their potential for various applications such as urban hydrology and flood prediction (Ochoa-Rodriguez et al., 2019).

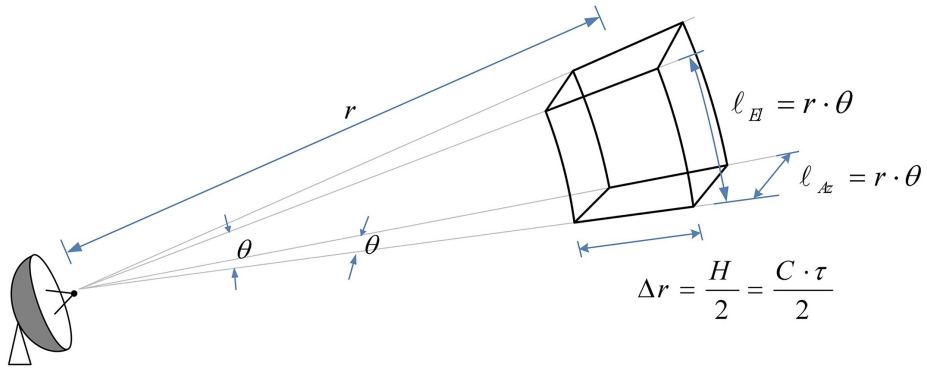


Figure 1.5 – Illustration of radar scanning volume (*retrieved from Ochoa-Rodriguez et al. (2019)*), showing the geometric relationships between range (r), beamwidth (θ), and pulse length (H). The range resolution (Δr) is determined by the pulse duration (τ) and the speed of light (C), with $\Delta r = \frac{H}{2} = \frac{C \cdot \tau}{2}$. The horizontal (azimuthal, ℓ_{Az}) and vertical (elevation, ℓ_{El}) extents of the radar beam, at a distance r, are given by $\ell_{Az} = r \cdot \theta$ and $\ell_{El} = r \cdot \theta$.

For sparsely gauged regions, global precipitation products like the Global Precipitation Measurement (GPM) mission integrate **satellite observations** with ground measurements. These products

provide consistent precipitation estimates with near-global coverage (Figure 1.6), particularly valuable for regions with sparse monitoring networks. While offering unprecedented spatial coverage, GPM products typically operate at coarser resolutions (10-25 km spatially and 3-hourly to daily temporally) than ground-based systems, presenting important trade-offs between extensive coverage and finer spatio-temporal detail for hydrological applications (Huffman et al., 2020).

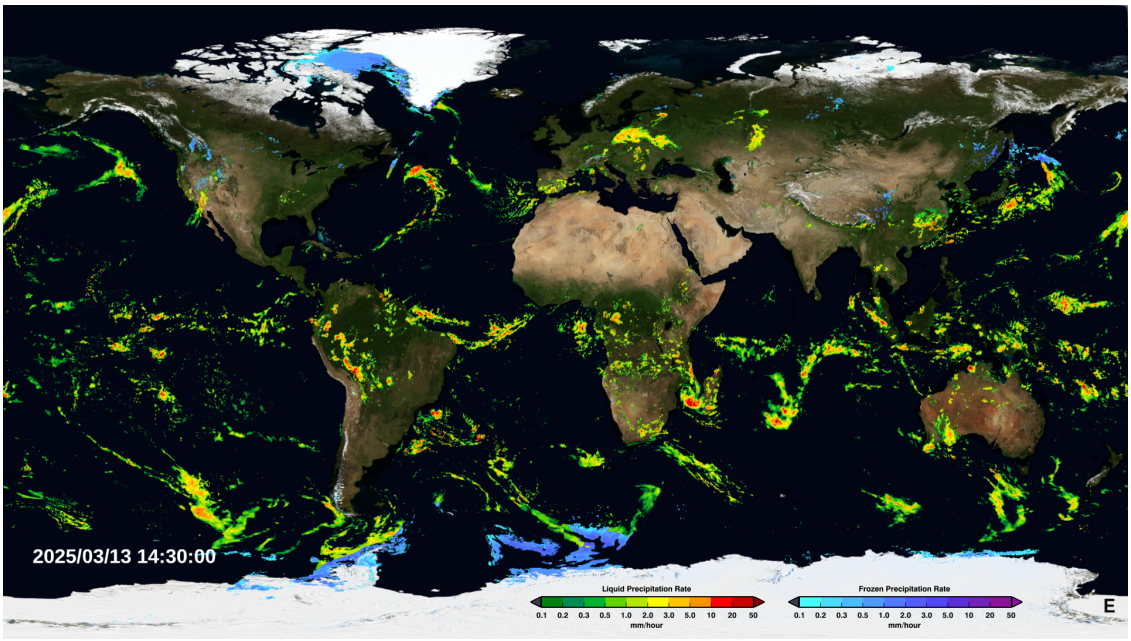


Figure 1.6 – Real-time global precipitation map from NASA's IMERG dataset, providing half-hourly updates by integrating data from the GPM satellite constellation. *Image Credit: ©NASA GPM IMERG (2025).*

Beyond direct measurement techniques, reanalysis systems integrate meteorological observations with numerical weather prediction models to produce consistent precipitation datasets. The ERA5 global reanalysis from the European Centre for Medium-Range Weather Forecasts (ECMWF), which replaced the previous ERA-Interim reanalysis, represents a significant advancement in this domain, incorporating a decade of improvements in model physics, core dynamics, and data assimilation techniques (Hersbach et al., 2020). ERA5 provides precipitation estimates at hourly intervals with approximately 31 km (9 km for ERA5-Land) spatial resolution globally, compared to 80 km for ERA-Interim. These reanalysis products are particularly valuable for hydrological modeling in regions with sparse rain gauge networks, offering spatially and temporally complete precipitation inputs for model forcing. In addition to global reanalysis products, regional systems such as SAFRAN ("Système d'Analyse Fournissant des Renseignements Atmosphériques à la Neige" in French) provide high-resolution meteorological analyses tailored to specific geographic areas. SAFRAN, developed by Météo-France [1], offers detailed precipitation data at finer spatial scales of 8 km (Vidal et al., 2010), which can be further downscaled to 1 km by combining observations from various sources with optimal and variational interpolation techniques (Durand et al., 2009). This high-resolution data is crucial for hydrological modeling in complex terrains, where local variations in precipitation can significantly impact runoff and streamflow dynamics.

1. https://meteofrance.com

1.2.2.2 River network measurements

Traditional river network measurements primarily rely on **gauging stations** strategically positioned along rivers to continuously monitor water levels (Figure 1.7). These stations employ various technologies, including water level sensors and direct velocity measurements, to estimate streamflow. The key hydrological variable of interest is discharge (or streamflow, typically expressed in

Figure 1.7 – A typical river gauging station setup at the Pude gauging station, located in Nanteuil-Auriac-de-Bourzac, France, illustrating the use of water level sensors to measure stage, which is then used to estimate streamflow. *Image Credit: ©Père Igor (2021).*

m^3/s or ft^3/s), as it represents the cumulative effect of upstream hydrological processes. The relationship between water level (stage) and discharge is generally non-linear and established using a stage-discharge rating curve (Kennedy, 1984), which converts continuous water level measurements into discharge estimates, enabling hydrologists to monitor streamflow over time (Turnipseed et al., 2010). Rating curves are calibrated using episodic stage-discharge measurements called gaugings. For instance, acoustic Doppler Current Profilers (ADCPs) are frequently used in large rivers to measure velocity profiles accurately (Gordon, 1989). For smaller streams, typical gauging methods include current-meters or dilution (Le Coz et al., 2014). The temporal resolution of discharge series ranges from sub-hourly to daily observations, with most operational networks providing at least daily mean discharge values. Global discharge databases, such as the Global Runoff Data Centre [2] (GRDC), along with national databases like the USGS Water Data [3] in the USA and HydroPortail [4] in France, compile these measurements, making them accessible for research and operational applications. However, traditional river gauging can be subject to uncertainties in rating curves (Horner et al., 2018), often due to lack of discharge measurements for various flow conditions, changes in channel geometry, sedimentation, or variations in flow conditions caused by flood events or human activities such as dredging or construction (Rantz, 1982). These changes can alter the relationship between stage and discharge, necessitating periodic recalibration of rating curves to maintain discharge estimation accuracy. The uncertainty of the rating curve can be quantified using Bayesian approaches (Le Coz et al., 2014), among many others (Kiang et al., 2018).

Complementing traditional gauging methods, Earth **observation satellites** increasingly contribute to hydrological monitoring by providing spatially distributed measurements of water heights

2. https://grdc.bafg.de
3. https://waterdata.usgs.gov/nwis
4. https://www.hydro.eaufrance.fr

and extents over river networks. For instance, the Surface Water and Ocean Topography (SWOT) mission, launched in 2022, uses wide-swath radar interferometry to measure surface water elevation, slope and width, offering unprecedented insights into (large) river dynamics, particularly in remote or poorly gauged basins (Biancamaria et al., 2016) (Figure 1.8). These observations complement

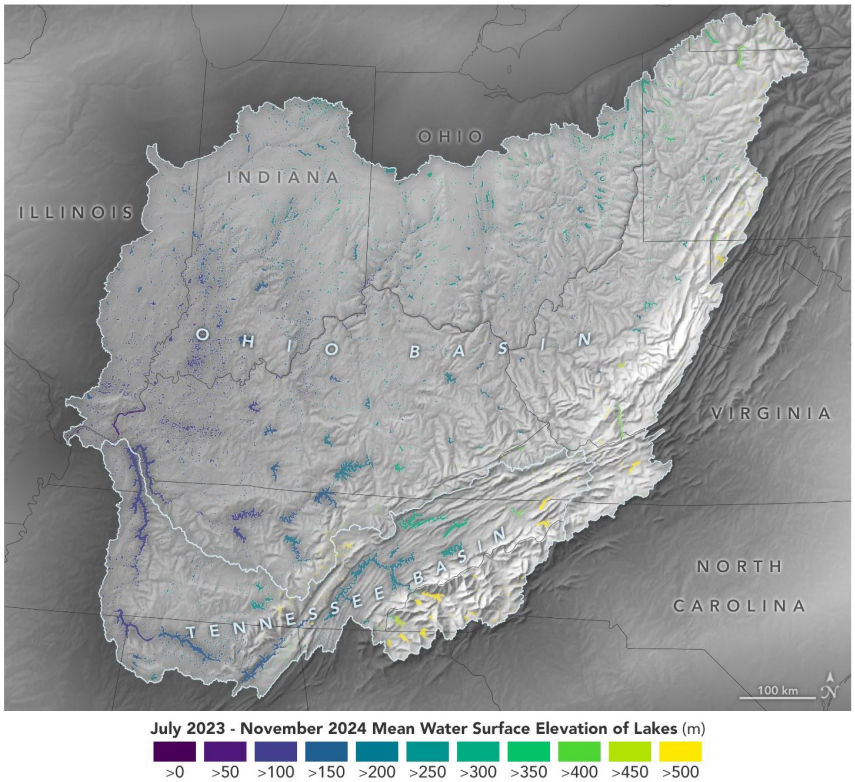


Figure 1.8 – Illustration of the SWOT satellite mission, capable of measuring surface water extent and elevation across the Ohio River basin, USA. *Image Credit: ©NASA/JPL-Caltech (2024).*

in-situ networks and are particularly valuable for ungauged or poorly gauged basins, though they typically come with coarser temporal resolution and various sources of uncertainty that must be accounted for in modeling applications (McCabe et al., 2017). The integration of these diverse data sources presents challenges related to scale discrepancies, measurement uncertainties, and data gaps, but also offers opportunities for developing more robust hydrological models. Nevertheless, estimating discharge from water height, width, and slope alone remains a challenging ill-posed inverse problem, especially when bathymetry and friction are uncertain or unknown and cannot be observed from space (cf. Garambois and Monnier (2015), Larnier, Monnier, et al. (2021) in reach scale hydraulic models). This issue is currently being investigated using hydrological closures via a basin-scale hydrological model chained with a network-scale hydraulic model to integrate altimetry observations consistently (Pujol, Garambois, and Monnier, 2022; Larnier, Garambois, et al., 2025).

1.2.3 Evolution of different hydrological modeling approaches

Despite significant advancements in hydrometric observation technologies offering extensive coverage and real-time data collection, fundamental limitations persist in hydrological monitoring. The impossibility of comprehensive data collection at all locations and time scales (Beven, 2012), along with the critical need for predictive capabilities to mitigate extreme events, necessitates hydrological modeling approaches, which serve as mathematical representations that transform meteorological

inputs (primarily precipitation) into hydrological responses (typically streamflow), commonly referred to as rainfall-runoff models. These models have evolved from simple empirical approaches established in the early 20th century to complex physics-based or conceptual distributed models and, more recently, to emerging hybrid approaches incorporating artificial intelligence (AI). They play a crucial role in water resources management and environmental science by extending limited observational records both spatially and temporally, and providing hydrological predictions under various scenarios.

This section reviews the evolution of three fundamental hydrological modeling approaches: data-driven models (traditionally based on empirical models and referred to as black-box models), physically-based modeling approaches based on physical principles and governing equations such as conservation of mass and momentum (referred to as white-box models), and conceptual models that provide a simplified representation of hydrological processes (gray-box models), often relying on conceptual storages or reservoirs to mimic the behavior of the watershed (Figure 1.9).

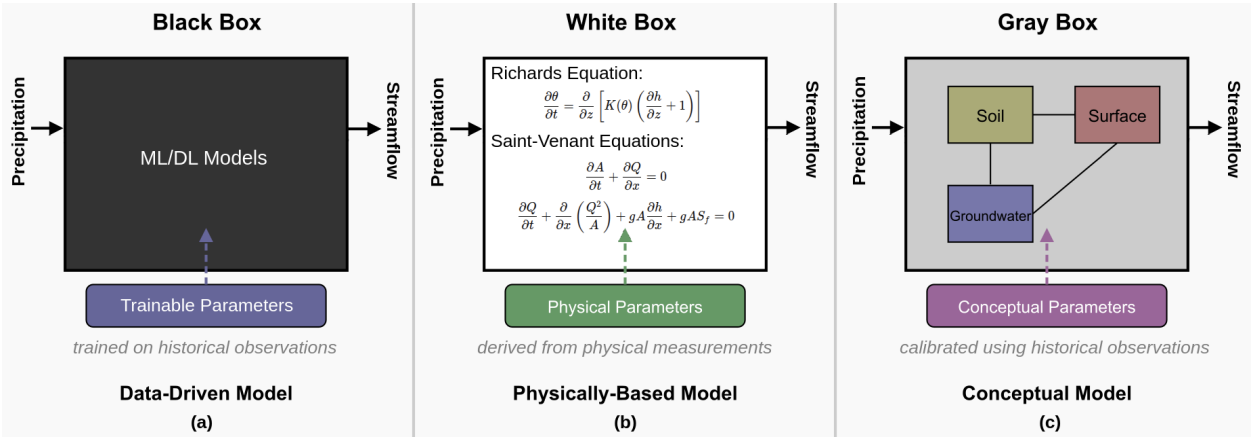


Figure 1.9 – Comparison of three fundamental hydrological modeling approaches: (a) Black-box models represent purely data-driven approaches that establish direct input-output relationships without explicit process representation, requiring extensive historical data for training; (b) White-box models incorporate physical principles and mathematical equations to simulate hydrological processes based on measurable physical parameters; (c) Gray-box models use simplified conceptual representations of watershed processes through interconnected storage elements (soil, groundwater, surface) governed by conceptual parameters that require calibration.

1.2.3.1 Empirical models: From rational method and unit hydrograph to statistical and data-driven approaches

While hydrology has a long history dating back to several millennia (Kirkby et al., 1971), the birth of hydrological modeling can be traced to the 1850s with the rational method introduced by Mulvaney (1850), according to Singh (2018). This method represents one of the earliest attempts at quantifying the rainfall-runoff relationship through a simple linear equation relating peak discharge to rainfall intensity, drainage area, and runoff coefficient. Despite its simplicity, this method remains in use today for small catchments and urban drainage design (see reviews in Singh (2018)).

A significant advancement came with the unit hydrograph theory introduced by Sherman (1932) in the 1930s, representing a quantitative method for modeling the rainfall-runoff relationship. This approach assumes a linear and time-invariant system, where the hydrograph resulting from a unit of effective rainfall is used to predict streamflow for other rainfall events. The unit hydrograph concept postulates that the response of a watershed to a unit depth of effective rainfall distributed uniformly

over the watershed for a specified duration is constant in time, which remains a fundamental concept in hydrology and formed the basis for early hydrological models, relying on empirical relationships derived from observed data (Nash, 1957; Dooge, 1973).

As computational capabilities improved, statistical methods gained prominence in hydrological modeling. Multiple regression techniques enabled the development of relationships between streamflow and various hydrometeorological variables (Beard, 1962). These methods facilitated the incorporation of multiple predictors and allowed for more nuanced representations of hydrological processes. Frequency analysis has emerged as a crucial tool for estimating the probability of extreme events, with methods like the log-Pearson type III or generalized extreme value (GEV) distributions becoming standard practice in flood frequency analysis (Te Chow et al., 1988). These approaches provided a statistical framework for estimating return periods of flood events, critical for infrastructure design and risk assessment.

Recent growth in computational power and data availability has catalyzed the development of machine learning (ML) approaches in hydrology. Artificial neural networks (ANNs) gained popularity in the 1990s for their ability to model complex, non-linear relationships without explicit physical representations (Hsu et al., 1995). These models demonstrated improved performance in rainfall-runoff modeling, particularly in catchments with limited physical data. The emergence of deep learning (DL) techniques has further expanded the capabilities of data-driven models. Long short-term memory (LSTM) networks, a type of recurrent neural network (RNN), have shown promising performance in capturing long-term dependencies in time series data, addressing a limitation of traditional ANNs (Kratzert, Klotz, Brenner, et al., 2018). Recently, increasingly sophisticated data-driven approaches, exemplified by random forests and gradient boosting machines, have proven effective for parameter regionalization due to their ability to capture non-linear relationships while maintaining robustness against overfitting (Saadi et al., 2019; Z. Song et al., 2022; Ding et al., 2024).

Empirical models show impressive improvements through their evolution from simple rational methods to complex data-driven approaches. These approaches require minimal physical assumptions while being able to capture complex relationships without explicit process representation (Solomatine et al., 2008). However, data-driven models often exhibit strong dependence on the quality and quantity of training data, potentially limiting their reliability in data-scarce regions or under changing climatic conditions (Reichstein et al., 2019; Beven, 2020; Sit et al., 2020). The "black-box" nature of many ML/DL models (Figure 1.9a) presents challenges for physical interpretation and extrapolation beyond the range of training data. As Reichstein et al. (2019) note, these models often struggle to respect fundamental physical constraints, potentially leading to physically implausible predictions. The lack of physical interpretability remains a central challenge. While recent efforts have focused on developing interpretable ML techniques (Molnar et al., 2020), extracting meaningful physical insights from complex models remains difficult. This limitation is particularly problematic for understanding causal relationships and system behavior under changing conditions, such as climate change or land use transformations.

1.2.3.2 Physically-based models

In the mid-20th century, alongside the evolution of empirical models, a new form of hydrological modeling emerged to address the limitations of existing empirical approaches, which was driven by advancements in computational capabilities and a deeper understanding of hydrological processes. The blueprint model introduced by Freeze et al. (1969) marked a watershed moment in physically-based, distributed modeling. This model is based on coupled subsystems, which include subsurface flow (groundwater and unsaturated zone), surface flow (overland and channel flow), and atmosphere-soil interface (precipitation, evaporation, and infiltration). These subsystems are linked

by partial differential equations (PDEs) that describe the physical laws of mass, energy, and momentum conservations with boundary conditions across different processes: (i) Richards' equation, which combines Darcy's law and mass conservation, describes unsaturated flow; (ii) 3D groundwater flow equation, derived from Darcy's law, governs groundwater movement; and (iii) Saint-Venant equations describe surface flow, including runoff generation, streamflow routing, and river dynamics.

As pointed out by Beven (2002), while the blueprint model provides a structure that was generally applicable, the descriptive equations used for each process require certain simplifying assumptions in real-word applications. Despite its theoretical complexity, this foundational work has led to the development of modern physically-based models in the subsequent decades. For example, TOPMODEL (Beven and Kirkby, 1979) introduced a significant departure from the full blueprint implementation by leveraging topographic similarity to reduce computational complexity. Rather than solving Richards' equation throughout the soil profile, TOPMODEL uses a topographic index (combining slope and contributing area) to identify hydrologically similar areas, effectively reducing the dimensionality of the modeling problem.

The SHE (Système Hydrologique Européen) model, developed by Abbott et al. (1986), represented one of the first comprehensive attempts to implement the blueprint concept in a fully distributed framework. SHE discretizes the watershed into a grid network and solves the governing equations for each grid cell, maintaining spatial representation of hydrological processes.

The Variable Infiltration Capacity (VIC) model (Liang et al., 1994) adopts a statistical approach to represent subgrid variability in soil moisture and infiltration processes. VIC replaces the complex PDEs of the blueprint with a parameterized representation of moisture storage and flux between soil layers. The model's distinctive feature is its use of a variable infiltration curve to represent the heterogeneity of infiltration capacity across a grid cell without explicitly resolving fine-scale processes.

Parflow (Kollet et al., 2006) represents a more computationally intensive approach that preserves much of the blueprint model's complexity. The model solves the 3D Richards' equation for subsurface flow coupled with the kinematic wave approximation of the Saint-Venant equations for surface flow, addressing computational challenges through efficient numerical methods, parallel computing architectures, and advanced solver techniques.

In the context of flash flood simulation, the spatially distributed MARINE model (Roux et al., 2011) comprises an event-based hydrological watershed module that describes infiltration, surface and subsurface runoff as well as river flows using a 1D kinematic wave model. The model structure is built on the understanding of the modeled flash flood-prone catchments, addressing effective physically-based flood modeling that accounts for spatially distributed information. The authors demonstrated that subsurface water transfer plays a crucial role in hydrograph response during extreme events, particularly in the recession period, marking an advance in spatially distributed flood modeling and extreme event analysis. This model has since been widely used in various contexts. For example, Douinot et al. (2018) developed a multi-hypothesis framework to further improve the understanding of flow dynamics during flash floods.

Despite the variety of physically-based modeling approaches, each of these models represents a unique approach to balancing the trade-off between theoretical complexity and practical implementation, demonstrating the evolution of physically-based modeling toward real-word applications while maintaining fundamental physical laws.

However, physically-based models, often referred to as "white-box" models (Figure 1.9b), continue to face challenges in parameter estimation, scale inconsistencies, and the representation of heterogeneity (Beven, 2006). Unlike empirical or conceptual models, the physical parameters in white-box models should, at least in principle, be obtained from physical measurements, rather than through a calibration or training process. However, these parameters often cannot be measured directly for several reasons. For instance, scale discrepancies arise, where parameters mea-

sured at a point scale frequently fail to represent the effective parameters needed at the model grid scale (Beven, 1989). Additionally, some parameters, such as those describing deeper soil layers or groundwater systems, are inaccessible for direct measurement across the spatial extents required for watershed modeling.

1.2.3.3 Conceptual hydrological representation: From lumped to fully distributed models

Conceptual hydrological models, often termed "gray-box" models, occupy the middle ground between empirical and physically-based approaches. These models represent watershed processes through simplified conceptualizations, typically using interconnected storage elements or reservoirs to mimic watershed behavior (Figure 1.9c). While conceptual models incorporate physical principles, they do not explicitly solve the governing equations of hydrodynamics. Instead, their conceptual parameters, though physically interpretable yet not directly measurable, must be estimated through a calibration process using observed data.

Unlike physically-based models, which focus on the direct numerical resolution of conservation laws, conceptual models aim to capture hydrological responses through simplified process representations that balance model complexity and computational efficiency. This makes them particularly suitable for operational hydrology and real-time forecasting, as they require fewer input data and have lower computational demands than fully distributed physics-based models. Similar to empirical models, conceptual models still rely on substantial observational data for calibration. Note that physically-based models and conceptual models are not entirely distinct, as both aim to balance the theoretical rigor of the original blueprint with practical computational constraints. However, physically-based models generally prioritize the preservation of physical principles and the resolution of governing equations rather than focusing on model calibration.

The evolution of conceptual hydrological models can be characterized along two main axes: (i) the pursuit of simplified yet efficient representations of physical processes (e.g., effective conceptual parameters and improved spatial representation) and (ii) advancements in data assimilation techniques to enhance model calibration.

One of the earliest conceptual models is the Stanford watershed model (SWM) developed by Crawford et al. (1966). This lumped conceptual model introduced a reservoir-based representation for different hydrological components using parameterized storage functions to route water through a catchment. However, lumped models like SWM assume spatial homogeneity, meaning that hydrological responses are averaged over an entire catchment, making them less suited for capturing spatial variability in hydrological processes. Recognizing the need to account for spatial heterogeneity, particularly in humid regions where variable source area dynamics play a crucial role in runoff generation, the Xinanjiang model (Zhao et al., 1980) introduced a variable infiltration curve, allowing runoff generation to vary within a watershed based on soil moisture conditions.

The concept of hydrological response units (HRUs) in the 1970s led to the development of semi-distributed "bucket-type" models, typically the HBV model developed by Bergström (1976). This model divides the watershed into HRUs based on topography, land use, and soil characteristics, thereby improving spatial representation while maintaining computational efficiency. Both lumped and semi-distributed models frequently employ heuristic-search optimization algorithms for parameter calibration, as the relatively small number of parameters allows for efficient optimization. Over time, optimization techniques applied to these models have evolved, incorporating more advanced methods. Notable examples include the shuffled complex evolution (SCE) algorithm (Duan et al., 1992), the distributed evolutionary algorithms (DEAP) (Fortin et al., 2012), and meta-heuristic search methods such as genetic algorithms (e.g., Deb et al. (2002)) and AMALGAM algorithm (Vrugt and Robinson, 2007). However, these algorithms remain non-gradient-based optimization

approaches, as they do not rely on derivative information but instead explore the solution space of limited dimension. This is problematic before gradient-based algorithms constitute a necessary ingredient in high-dimensional estimation problems, especially when neural networks are involved (LeCun, Bottou, et al., 1998).

The recognition of spatial heterogeneity in watershed processes has driven the evolution from lumped or semi-distributed approaches to fully distributed models. For example, the GR4J model, introduced by Perrin et al. (2003), belongs to the family of soil moisture accounting models, initially developed as a daily lumped rainfall-runoff model with only four effective conceptual parameters. GR4J has since been adapted for various contexts, exemplified by GRD (e.g., Jay-Allemand, Javelle, et al. (2020)) which extends GR4J to account for spatially distributed parameters, or GR5H (e.g., Astagneau, Bourgin, Andréassian, and and (2024)) which introduces an additional model parameter and is adapted for hourly time steps (collectively referred to as GR or GR-like models). Recently, GR-like structures have been implemented in fine time step, fully distributed models within a variational data assimilation (VDA) method addressing the challenge of high-dimensional optimization using automatic differentiation and gradient-based optimization methods (e.g., Castaings et al. (2009), Lee et al. (2012), and Jay-Allemand, Javelle, et al. (2020)).

In the realm of fully distributed, coupled surface–subsurface hydrological modeling, recent advancements have been made to enhance the representation of hydrological processes. For instance, Flipo et al. (2023) introduced a model that integrates surface and subsurface hydrological processes to improve the simulation of water flow dynamics. This coupled approach allows for a more accurate representation of interactions between surface water and groundwater, which is crucial for understanding hydrological responses in various environmental settings.

Alongside conceptual representation enhancements, flexible modeling frameworks addressing model structure errors have been developed, exemplified by the FUSE (framework for understanding structural errors) model developed by Clark et al. (2008), which enables systematic exploration of model structure uncertainty by combining elements from multiple conceptual models.

Despite their advantages, conceptual models face several challenges, particularly in sparse data environments, where limited observations constrain parameter calibration. They often require extensive calibration efforts and are prone to equifinality (Beven, 2006), where multiple parameter sets yield similar model outputs (i.e., non-unique solutions), especially in the context of high-dimensional optimization of spatially distributed parameters. Furthermore, conceptual models, being simplifications of physical processes, are inherently sensitive to observational data for calibration and are affected by multiple sources of uncertainty, including structural errors and measurement errors, which can be systematically analyzed and reduced by decomposing predictive uncertainty into its contributing sources using Bayesian inference (Renard, Kavetski, Leblois, et al., 2011). Addressing these issues also necessitates the development of advanced data assimilation approaches, which will be discussed in the subsequent section. Additionally, a promising avenue is the integration of conceptual models with neural networks in hybrid modeling approaches. This can help mitigate parameterization challenges and improve predictive performance, as will be discussed in Section 1.4.1.

1.2.4 Data assimilation and Bayesian inference

Irrespective of their type and complexity, hydrological models are more or less empirical and uncertain representations of multiscale coupled hydrological processes whose observability is limited. Conceptual hydrological model parameters are generally effective quantities that cannot be directly measured. Instead, they are typically inferred through a calibration procedure aimed primarily at obtaining satisfactory streamflow simulations (e.g., Beven (2001), Kirchner (2006), Gupta, Beven, et al. (2006), and Vrugt, Ter Braak, et al. (2008)). This calibration can be framed as an inverse problem, where the goal is to determine unknown (sought) parameters from limited and noisy

observations.

Data assimilation (DA) provides a systematic approach to integrating observational data with model simulations, aiming to fit model response to the available data (cf. Bouttier et al. (2002) and Monnier (2021)). DA methods have evolved around two families: the variational approach, rooted in optimal control theory (Lions, 1971), and sequential methods based on statistical estimation. Sequential methods, such as Kalman filters, recursively correct model states at observation times using only the previous state and observations, while variational methods consider all observations simultaneously to guide the model trajectory by adjusting its parameters. Both approaches can be unified within a Bayesian framework under specific assumptions.

In the context of Bayesian inference, the DA process can be expressed as computing the posterior distribution of the vector of unknown parameters $\boldsymbol{\rho}$ given observed data Y (obtained by measurements here):

$$P(\boldsymbol{\rho}|Y) = \frac{P(Y|\boldsymbol{\rho})P(\boldsymbol{\rho})}{P(Y)} \tag{1.1}$$

where $P(\boldsymbol{\rho}|Y)$ is the posterior probability density function (PDF) of the unknown parameters $\boldsymbol{\rho}$, $P(Y|\boldsymbol{\rho})$ is the likelihood function (how well the model explains the observations), $P(\boldsymbol{\rho})$ is the prior PDF of the unknown parameters, and $P(Y)$ is the marginal likelihood (or evidence). Since the evidence $P(Y)$ is a constant relative to $\boldsymbol{\rho}$, the posterior PDF is proportional to the product of the likelihood and the prior PDF:

$$P(\boldsymbol{\rho}|Y) \propto P(Y|\boldsymbol{\rho})P(\boldsymbol{\rho}) \tag{1.2}$$

Sequential DA methods update model states incrementally as new observations become available. A classical example is the Kalman filter (KF), introduced by Kalman (1960), which updates the model state estimate recursively using a "predict-update" cycle. This forms the foundation of a so-called "recursive Bayesian estimation" (or Bayesian filtering) framework in subsequent studies, where the posterior distribution of the model state is updated sequentially as new data are observed. Let $\boldsymbol{\rho}_t$ denote the model state and Y_t the observation at a time step t. The sequence of observations up to this time step is $Y_{1:t}$. The recursive Bayesian framework is based on the following equation:

$$P(\boldsymbol{\rho}_t|Y_{1:t}) \propto P(Y_t|\boldsymbol{\rho}_t)P(\boldsymbol{\rho}_t|Y_{1:t-1}) \tag{1.3}$$

where the prior $P(\boldsymbol{\rho}_t|Y_{1:t-1})$ represents the probability distribution of the state $\boldsymbol{\rho}_t$ given all observations up to the previous time step $t-1$. Then, the predict-update cycle involves:

— Prediction step: The prior distribution of the state is predicted based on the previous state estimate and the system dynamics:

$$P(\boldsymbol{\rho}_t|Y_{1:t-1}) = \int P(\boldsymbol{\rho}_t|\boldsymbol{\rho}_{t-1})P(\boldsymbol{\rho}_{t-1}|Y_{1:t-1})\, d\boldsymbol{\rho}_{t-1} \tag{1.4}$$

where $P(\boldsymbol{\rho}_t|\boldsymbol{\rho}_{t-1})$ represents the transition model, describing the evolution of the state from time $t-1$ to t. This calculation relies on the Markov assumption, stating that the future state depends only on the current state and not on any earlier states: $P(\boldsymbol{\rho}_t|\boldsymbol{\rho}_{1:t-1}) = P(\boldsymbol{\rho}_t|\boldsymbol{\rho}_{t-1})$.

— Update step: The posterior distribution is updated as follows:

$$P(\boldsymbol{\rho}_t|Y_{1:t}) \propto P(Y_t|\boldsymbol{\rho}_t)P(\boldsymbol{\rho}_t|Y_{1:t-1}) \tag{1.5}$$

where $P(Y_t|\boldsymbol{\rho}_t)$ is the likelihood of the observation given the state.

Sequential DA methods are particularly effective for real-time applications where immediate corrections are necessary. In this work, we focus on the variational approach, which is well-suited to non-linear dynamics, high-dimensional parameter vectors, and heterogeneous observations.

Variational DA (VDA) methods aim to reduce the discrepancies (in the sense of a cost function to be defined) between observed data and model output by optimizing the model parameters, commonly referred to as the "control vector", in an iterative process. This calibration process is analogous to the training process in machine learning (ML), where trainable parameters (weights and biases of a neural network) form the control vector in DA. The so-called 3D-Var and 4D-Var methods are commonly used variational approaches in meteorology, where 3D-Var considers observations at a single time point (with static model state), and 4D-Var considers observations over a time window (with dynamic model state). The inverse problem involves finding an optimal control vector $\hat{\boldsymbol{\rho}}$ that minimizes the difference between simulation and observation computed based on a cost function J:

$$\hat{\boldsymbol{\rho}} = \arg\min_{\boldsymbol{\rho}} J(\boldsymbol{\rho}) \tag{1.6}$$

This cost function typically includes two main components:

— the observation term, which measures the misfit between the model simulations and the observations over the assimilation window;

— the background term, typically a Tikhonov regularization (Tikhonov et al., 1943), which penalizes deviations of the model parameter from a prior estimate (first guess or background parameter).

Mathematically, the cost function $J(\boldsymbol{\rho})$ can be expressed as:

$$J(\boldsymbol{\rho}) = J_{obs}(\boldsymbol{\rho}) + \gamma J_{bkg}(\boldsymbol{\rho}) \tag{1.7}$$

where $J_{obs}(\boldsymbol{\rho})$ is the observation term, $J_{bkg}(\boldsymbol{\rho})$ is the background term, and γ is the regularization parameter that controls the strength of the penalty applied to deviations from the background parameter. The observation term $J_{obs}(\boldsymbol{\rho})$ is given by:

$$J_{obs}(\boldsymbol{\rho}) = ||\boldsymbol{O}^{-1}\left(M(\boldsymbol{\rho}) - Y\right)||^2 \tag{1.8}$$

where M is the forward model that simulates values in the observation space and $\boldsymbol{O}$ is the observation error covariance matrix. The background term $J_{bkg}(\boldsymbol{\rho})$ is given by:

$$J_{bkg}(\boldsymbol{\rho}) = ||\boldsymbol{B}^{-1}\left(\boldsymbol{\rho} - \boldsymbol{\rho}_0\right)||^2 \tag{1.9}$$

where $\boldsymbol{\rho}_0$ is the background parameter and $\boldsymbol{B}$ is the background error covariance matrix.

The minimization of the cost function $J(\boldsymbol{\rho})$ can be interpreted within the Bayesian framework as maximizing the posterior probability $P(\boldsymbol{\rho}|Y)$, which is equivalent to minimizing the negative logarithm of the posterior. Using Bayes' theorem (Equation 1.2), the corresponding cost function can be defined as:

$$J(\boldsymbol{\rho}) = -\log\left(P(Y|\boldsymbol{\rho})\right) - \log\left(P(\boldsymbol{\rho})\right) \tag{1.10}$$

Under the assumption of a Gaussian distribution for the prior $P(\boldsymbol{\rho})$ and a Gaussian likelihood, minimizing the cost function in Equation 1.10 is equivalent to minimizing the one in Equation 1.7, where the likelihood $P(Y|\boldsymbol{\rho})$ corresponds to the observation term $J_{obs}(\boldsymbol{\rho})$ and the prior probability $P(\boldsymbol{\rho})$ corresponds to the background (Tikhonov regularization) term $J_{bkg}(\boldsymbol{\rho})$ (see demonstration in Monnier (2021)). This regularization is analogous to the weight decay technique in ML or ridge regression in statistics, penalizing large deviations from the background parameter by imposing a penalty proportional to the squared magnitude of these deviations. These stabilization techniques address ill-posed problems by preventing over-reliance on observational data and by providing constrain in the VDA process, thereby enhancing the robustness of the solution.

An advantage of the Bayesian framework is its ability to generalize the problem and provide a clearer statistical interpretation, which is particularly useful when considering stochastic factors

and uncertainties. For instance, when considering observations that consist of both measured values and uncertainty terms, the observations can be expressed as:

$$Y = M(\boldsymbol{\rho}) + \delta + \epsilon \tag{1.11}$$

where δ represents the structural error of the model and ϵ represents the observations errors, both of which can be modeled probabilistically. This formulation allows the inverse problem of maximizing the posterior $P(\boldsymbol{\rho}|Y)$ to account for different sources of uncertainty. In this context, the equivalence between the inverse problems defined by Equation 1.7 and Equation 1.10 remains valid under the assumption that $\delta + \epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{O})$.

The optimality condition for the optimization problem, $\nabla_{\boldsymbol{\rho}} J = 0$, is typically solved using gradient-based algorithms, especially in high-dimensional control spaces (e.g., L-BFGS-B (Zhu et al., 1997) or ADAM (Kingma et al., 2014)). The adjoint state method is commonly employed as it enables accurate computation of the gradient of the cost function with respect to the control vector, regardless of the control vector's size, even for complex dynamic models. Additionally, this method can provide gradients of model outputs to input parameters, which represents local sensitivity. These local sensitivities can then be utilized to compute global derivative-based sensitivity measures (e.g., Sobol et al. (2010), Lamboni et al. (2013), Gupta and S. Razavi (2018), and S. Razavi et al. (2019)).

Automatic differentiation (AD) tools are commonly used in such adjoint state methods. AD has evolved significantly, with both static and dynamic computation graphs offering unique advantages. Static computation graphs, as employed in tools like Tapenade (Hascoet et al., 2013), are constructed before the program's execution. This approach allows for optimizations such as common sub-expression elimination, where redundant calculations are identified and removed to enhance computational efficiency. In contrast, dynamic computation graphs, utilized in frameworks like PyTorch (Paszke et al., 2019), are built on-the-fly during execution. This method offers flexibility for models with adaptive computations, such as those involving loops or conditional statements, where the computation flow can vary based on input data.

1.3 Artificial intelligence at a glance

1.3.1 From mind to machine: A natural extension of human intelligence

By 2025, artificial intelligence (AI) has become ubiquitous, often perceived as a "magical" tool capable of addressing numerous challenges. While the outcomes of AI applications can indeed seem remarkable, the evolution of AI is deeply rooted in the advancement of computational power and the application of well-established mathematical principles. The foundational concepts of modern AI are grounded in linear algebra, probability, statistics (e.g., Bayesian inference), and numerical optimization (see Goodfellow, Bengio, et al. (2016) for fundamental concepts of modern AI).

Every biological organism on Earth possesses intelligence, which has evolved over time, according to Darwin's theory of evolution. The aspiration to imbue non-biological entities, such as machines, with intelligence has existed throughout human history. This ambition can be traced back to myths and legends of automatons, such as Talos in Greek mythology, and early computational theorists and science fiction writers, such as Ada Lovelace's speculative work on computational machines in the 19th century. However, the term "artificial intelligence" was first formally recognized when Turing (1950) presented his seminal paper exploring the question, "Can machines think?" This marked the beginning of a new era in computational science.

The concept of machine learning (ML) gained widespread recognition, marking the birth of the ML domain. Since then, the field has evolved alongside advancements in related mathematical domains, such as statistics, data assimilation, and optimization. Early ML models include the

perceptron introduced by Rosenblatt (1958), which first defined "neural networks," although the underlying idea had been recognized by McCulloch et al. (1943) without the use of this terminology.

The subsequent decades witnessed the development of more complex ML models, characterized by increased depth and a larger number of trainable parameters. The multi-layer perceptron, introduced by Rumelhart et al. (1986), provided a clearer framework for training neural networks using backpropagation. Gradient-based optimization algorithms became a crucial approach for addressing the problem of dimensionality when training deep neural networks (LeCun, Bottou, et al., 1998), a challenge that had already been acknowledged in geophysical sciences through earlier developments in data assimilation techniques and adjoint state methods based on control theory (Lions, 1971). The term "deep learning" (DL) emerged with the rise of ML models, typically referring to deep neural networks such as recurrent neural networks (RNNs) (Elman, 1990) and long short-term memory networks (LSTMs) (Hochreiter and Schmidhuber, 1997). However, the field experienced periods of stagnation, known as "AI winters" (Toosi et al., 2021), wherein technological advancement was constrained by critical limitations in computational infrastructure, data scarcity, and consequent funding restrictions that impeded research progression.

The resurgence of AI began in the late 2000s, driven by advancements in computational power, large-scale data availability, and efficient training techniques. The efficiency of AI models became apparent when trained on large datasets, enabling the extraction of multi-level features (LeCun, Bengio, et al., 2015). Increased computational power allowed for the training of complex models at a lower cost, leading to the application of AI in diverse domains. This necessitated the development of specialized DL models, such as convolutional neural networks (CNNs), originally introduced by LeCun, Boser, et al. (1989) for handwritten digit recognition, and later popularized by Krizhevsky et al. (2012) for image processing. Other notable architectures include U-Net (Ronneberger et al., 2015) for biomedical image segmentation, and Transformers (Vaswani et al., 2023) for text and speech processing. More recently, generative AI has surged, with models such as generative adversarial networks (GANs) (Goodfellow, Pouget-Abadie, et al., 2020) and large-scale transformer-based architectures like GPT (Brown et al., 2020) and DALL-E (Ramesh et al., 2021) demonstrating unprecedented language understanding and generative potential.

Despite the diversity of AI developments, current AI research focuses on four main axes:

— **Data:** This encompasses the development of data collection methods, pre-processing techniques, and feature engineering to provide meaningful and high-quality input data for training models;
— **Model:** The development of model architectures is currently the primary focus of the field. This involves designing neural networks and other ML models that can effectively capture the underlying patterns in the data tailored to specific needs;
— **Optimization:** Optimization algorithms and various training techniques address the challenges of training DL models efficiently. This includes gradient-based methods, regularization techniques, and hyperparameter tuning to enhance model performance and convergence.
— **Computational efficiency:** The use of specialized hardware such as GPUs (graphics processing units) and TPUs (tensor processing units) has significantly accelerated the training and deployment of AI models. This axis focuses on leveraging computational resources to handle large-scale data and complex model architectures.

Each of these axes is closely related to any modeling approach, especially in hydrology, where the complexity of natural systems requires sophisticated data handling, model design, optimization strategies, and computational power. This focus aligns with the main goal of this thesis, which seeks to integrate AI techniques not as an isolated tool, but as part of an advanced modeling approach built on the foundational principles and advancements in the field of hydrology over the past century.

1.3.2 From linear regression to deep learning models

While AI architectures are numerous and varied, this section aims to explain the fundamental ideas behind state-of-the-art AI models used in hydrological modeling.

1.3.2.1 Linear regression

First, a natural way to understand a phenomenon is to explain its output (response or target) in relation to its input (data or features). A naive approach to modeling this relationship is to assume that the output depends in an affine way on the input. In the simplest case, where both the input and output are scalars, this can be expressed as:

$$y = \alpha x + \beta \tag{1.12}$$

This simple model is called linear regression, but note that linearity is between the output y and the model parameters (i.e., α, β), and not between y and x (the latter relation being affine and not linear). A relation of the form $y = \alpha x^2 + \beta x + \gamma$, for instance, is still a linear regression (y is a linear function of (α, β, γ)).

For a more general case, where we aim to explain multiple outputs $\boldsymbol{y} = (y_1, ..., y_n)^T$ using a multivariate input $\boldsymbol{x} = (x_1, ..., x_m)^T$, multiple linear regression assumes that the relationship between each dependent variable y_j and the model parameters is linear, $\forall j = 1..n$:

$$y_j = \left(\sum_{i=1}^{m} \alpha_{i,j} x_i \right) + \beta_j \tag{1.13}$$

Now if we pose a matrix $\boldsymbol{W} = (\alpha_{i,j})^T_{1 \leq i \leq m,\, 1 \leq j \leq n} \in \mathbb{R}^{n \times m}$, and a vector $\boldsymbol{b} = (\beta_j)^T_{1 \leq j \leq n} \in \mathbb{R}^n$, Equation 1.13 is equivalent to:

$$\boldsymbol{y} = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b} \tag{1.14}$$

This representation of the linear regression model is similar to Equation 1.12 but in higher data and response spaces.

While the linear model may seem like a very basic approach, it forms the foundation of the very first ML model called the "perceptron" (McCulloch et al., 1943; Rosenblatt, 1958). The mathematics of the linear model and the perceptron are essentially the same; however, the perceptron vividly illustrates the concept of interconnected neurons (Figure 1.10). This is one of the earliest representations that opened the idea that machines can learn (Turing, 1950).

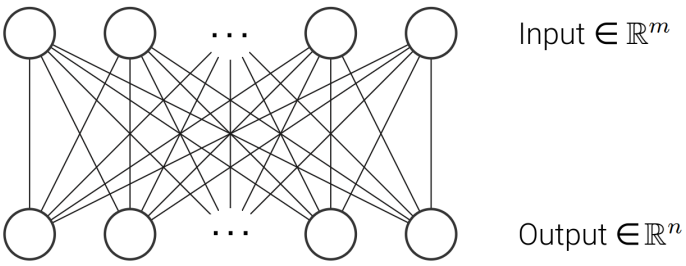


Figure 1.10 – Illustration of a perceptron, where each node represents a neuron, and each line represents the connection between two neurons via the linear operators applied to the trainable parameters $\boldsymbol{W}$ and $\boldsymbol{b}$.

1.3.2.2 Multilayer perceptron

However, many real-world phenomena exhibit complex dependencies that cannot be fully captured by a linear model, necessitating the use of more flexible approaches. A natural approach is based on the idea that, we can introduce an intermediate variable $\boldsymbol{z} = (z_1, ..., z_p)^T \in \mathbb{R}^p$ that can be explained by $\boldsymbol{x}$ using a linear model, and then can be used to explain $\boldsymbol{y}$ using a second linear model:

$$\begin{cases} \boldsymbol{z} &= \boldsymbol{W}_1 \boldsymbol{x} + \boldsymbol{b}_1 \\ \boldsymbol{y} &= \boldsymbol{W}_2 \boldsymbol{z} + \boldsymbol{b}_2 \end{cases} \tag{1.15}$$

where $\boldsymbol{W}_1 \in \mathbb{R}^{p \times m}, \boldsymbol{b}_1 \in \mathbb{R}^p$ and $\boldsymbol{W}_2 \in \mathbb{R}^{n \times p}, \boldsymbol{b}_2 \in \mathbb{R}^n$ are the parameters of the two linear models.

Note that developing Equation 1.15 leads to $\boldsymbol{y} = \boldsymbol{W}_2 \boldsymbol{W}_1 \boldsymbol{x} + (\boldsymbol{W}_2 \boldsymbol{b}_1 + \boldsymbol{b}_2)$, which still remains a linear model. The flexibility of such an approach only emerges when non-linear activation functions—such as ReLU (rectified linear unit) or TanH (hyperbolic tangent)—are introduced between the linear transformations:

$$\begin{cases} \boldsymbol{z} &= \psi\left(\boldsymbol{W}_1 \boldsymbol{x} + \boldsymbol{b}_1\right) \\ \boldsymbol{y} &= \boldsymbol{W}_2 \boldsymbol{z} + \boldsymbol{b}_2 \end{cases} \tag{1.16}$$

where ψ denotes an activation function, for instance, ReLU, defined as $\psi(x) = \max(0, x), \forall x \in \mathbb{R}$.

This is the principle behind the most basic neural network and AI model called a multilayer perceptron (MLP) (Rumelhart et al., 1986), where each linear model (followed by an activation function) represents a layer, and the parameters $\boldsymbol{W}_{i=1..2}$ are the weights, while $\boldsymbol{b}_{i=1..2}$ are the biases. Such an intermediate variable $\boldsymbol{z}$ corresponds to the output of the hidden layer and contains multiple neurons (p neurons in this case), where each neuron is fully connected to neurons in both the previous and subsequent layers (Figure 1.11). These layers are often called dense layers, and an MLP is a type of fully connected neural network (FCNN).

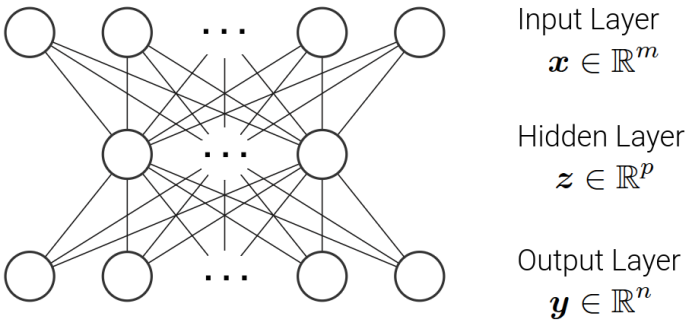


Figure 1.11 – Illustration of a multilayer perceptron with a single hidden layer, where each node represents a neuron, and each line represents the connection between two neurons via the linear operators applied to the weight $\boldsymbol{W}_i$ and bias $\boldsymbol{b}_i$.

In addition to introducing non-linear transformations in the hidden layers, activation functions are also used to meet specific needs for the output layers, depending on the task at hand (e.g., classification or regression). Furthermore, adding more hidden layers creates more non-linear features, which can better approximate the desired output response:

$$\begin{cases} \boldsymbol{z}_1 &= \psi_1\left(\boldsymbol{W}_1 \boldsymbol{x} + \boldsymbol{b}_1\right) \\ \boldsymbol{z}_2 &= \psi_2\left(\boldsymbol{W}_2 \boldsymbol{z}_1 + \boldsymbol{b}_2\right) \\ \vdots & \\ \boldsymbol{y} &= \psi_{k+1}\left(\boldsymbol{W}_{k+1} \boldsymbol{z}_k + \boldsymbol{b}_{k+1}\right) \end{cases} \tag{1.17}$$

where $\psi_{i=1..k+1}$ represent the activation functions (which can be the same or different) used in the k hidden layers and the output layer.

The idea of adding more hidden layers, which forms a deep neural network, is similar to the concept of approximating a curve by discretizing it into many linear segments. This approach is analogous to techniques used in calculus, such as differentiation and integration, where complex functions are approximated by simpler ones such as piecewise linear functions (approximation of function in a given space with functions from a dense subset).

Despite its simple mathematical form, the MLP remains a foundational building block of modern neural networks. This introduced the first view of a machine "brain" via the perceptrons, fully connected in a manner that mimics a biological brain. The MLP remains widely used due to its flexibility in adapting to most modeling problems and its simple implementation.

1.3.2.3 Recurrent neural networks

The need for recurrent neural networks (RNNs) arises when handling sequential data (e.g., time series). Feedforward neural networks, such as MLPs, which propagate information in one direction, are less efficient in this context. The principle behind any RNN is to create recurrent links, providing networks with a dynamic memory through feedback from hidden states (Rumelhart et al., 1986).

To understand this principle, we first consider what an MLP does when handling sequential data. Let $\left(\boldsymbol{x}^{(1)}, ..., \boldsymbol{x}^{(\mathcal{T})}\right) \in \mathbb{R}^{m \times \mathcal{T}}$ be input sequential data of length $\mathcal{T}$, with m features; that is, at each time step $t \in [1, \mathcal{T}]$, $\boldsymbol{x}^{(t)} = \left(x_1^{(t)}, ..., x_m^{(t)}\right)^T \in \mathbb{R}^m$. To simplify the mathematical representation, we consider neural networks without hidden layers (the principle applies in the same manner for networks with hidden layers). The outputs $\left(\boldsymbol{y}^{(1)}, ..., \boldsymbol{y}^{(\mathcal{T})}\right) \in \mathbb{R}^{n \times \mathcal{T}}$ are simply obtained by applying the following transformation for each time step, $\forall t \in [1, \mathcal{T}]$:

$$\boldsymbol{y}^{(t)} = \psi\left(\boldsymbol{W}\boldsymbol{x}^{(t)} + \boldsymbol{b}\right) \tag{1.18}$$

where ψ represents the non-linear activation function (e.g., ReLU, Sigmoid, TanH, etc.). This approach computes each time step explicitly without accounting for the temporal dependencies that are often crucial in sequential data.

In general, any way to introduce a recursive formulation into a feedforward neural network forms an RNN. For example, a classical way to form an RNN is to introduce hidden states that are computed recursively over time as follows (see Figure 1.12 for illustration):

$$\begin{cases} \boldsymbol{s}^{(0)} \in \mathbb{R}^n, \\ \boldsymbol{s}^{(t)} = \omega\left(\boldsymbol{U}\boldsymbol{x}^{(t)} + \boldsymbol{V}\boldsymbol{s}^{(t-1)} + \boldsymbol{h}\right), \forall t \in [1, \mathcal{T}] \\ \boldsymbol{y}^{(t)} = \psi\left(\boldsymbol{W}\boldsymbol{s}^{(t)} + \boldsymbol{b}\right), \forall t \in [1, \mathcal{T}] \end{cases} \tag{1.19}$$

where $\boldsymbol{U} \in \mathbb{R}^{n \times m}$, $\boldsymbol{V} \in \mathbb{R}^{n \times n}$ are the weights, $\boldsymbol{h} \in \mathbb{R}^n$ is the bias, and ω is the associated activation function in the hidden-state layer. Here, the trainable parameters are $(\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{W}, \boldsymbol{h}, \boldsymbol{b})$.

This design allows the network to learn and maintain a memory of past inputs through feedback mechanisms, capturing the temporal dynamics inherent in sequential data. This property makes RNNs well-suited for time series prediction and sequence modeling such as language understanding and speech recognition.

Now, let us consider an approach to constructing an RNN inspired by residual neural networks (ResNet) (K. He et al., 2016). In their paper, the authors highlighted the difficulty of training deep neural networks (e.g., a deep RNN in Equation 1.19) due to vanishing gradients. To address this, they explicitly reformulated the layers as learning residual functions with reference to the layer

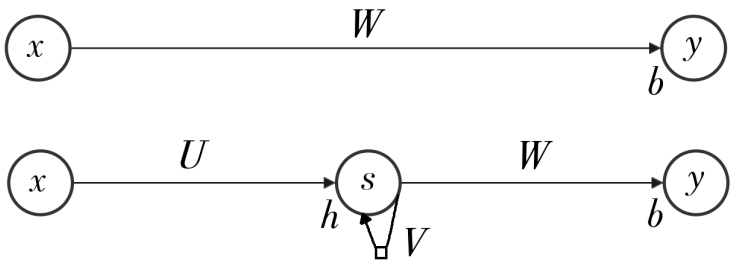


Figure 1.12 – Comparison of a feedforward neural network (top) and an RNN (bottom), where the RNN introduces recurrent connections to capture temporal dependencies. The arrow with a square in the middle signifies a recursive relationship, indicating that the current hidden state is influenced by both the current input and the hidden state from the previous time step.

inputs. Applying this principle to the state update equation in Equation 1.19, we obtain a residual RNN, where the state update is expressed as:

$$\boldsymbol{s}^{(t)} = \boldsymbol{s}^{(t-1)} + \omega\left(\boldsymbol{U}\boldsymbol{x}^{(t)} + \boldsymbol{h}\right) \tag{1.20}$$

This approach allows the RNN to learn only the residual functions with respect to the layer inputs (via trainable parameters $\boldsymbol{U}$ and $\boldsymbol{h}$), rather than the "unreferenced" functions like the state update in Equation 1.19, as termed by K. He et al. (2016).

The concept of residual networks later inspired an interesting idea proposed by R. T. Q. Chen et al. (2019), suggesting that the residual function can be reformulated as a continuous state dynamic represented by an ordinary differential equation (ODE):

$$\frac{d\boldsymbol{s}^{(t)}}{dt} = \omega\left(\boldsymbol{U}\boldsymbol{x}^{(t)} + \boldsymbol{h}\right) \tag{1.21}$$

The states are computed by solving the ODEs using numerical schemes. This approach is closely related to the real-word modeling of dynamic systems. More generally, the function ω can be any differentiable function that describes the state dynamics. An adjoint model is crucial for backpropagation since the neural network incorporates a numerical scheme to solve the ODEs. This example shows a mathematical representation similar to real-world modeling problems, making the integration of neural networks into an ODE solver—forming a neural ODE—very promising, which is also a key research objective in this thesis.

Another type of RNN is the long short-term memory network (LSTM) (Hochreiter and Schmidhuber, 1997), which proves more efficient than traditional RNNs for handling sequential data with long-term dependencies (e.g., for weather forecasting). In essence, the key idea behind LSTMs is the introduction of cell states alongside hidden states, enabling the network to capture and retain information across extended sequences more effectively. These cell states function as a memory unit, preserving information over long durations, and are regulated by three types of gates: the input gate, the forget gate, and the output gate. These gates govern the flow of information into and out of the cell state, allowing the network to selectively remember or forget information.

1.3.2.4 Convolutional neural networks

Convolutional neural networks (CNNs) represent a specialized class of neural networks designed to efficiently process grid-like data, such as images or spatio-temporal maps. CNNs were first introduced by LeCun, Boser, et al. (1989), initially designed for document recognition, and later

improved by Krizhevsky et al. (2012) for large-scale image classification. Regardless of the diverse techniques employed in these models, the core functionality of CNNs centers around the convolution operation.

The concept of convolution dates back to the early works of Fourier in signal processing in the 19th century. Let us consider a signal $x(t)$ that produces a real value at each time t. Suppose this signal is very noisy and cannot be directly interpreted. Now, imagine a filter $f(u)$ acting like a lens that we slide over the noisy signal, focusing more on recent parts and reducing the influence of distant parts. In other words, this filter emphasizes the values of the signal near the current time t, while giving less importance to the values further away in time. Applying such operation at every moment results in a process that can be mathematically represented by the convolution operation:

$$(x * f)(t) = \int_{-\infty}^{+\infty} x(u) \cdot f(t-u) du \tag{1.22}$$

where "$*$" denotes the convolution operation of x and f.

Now, consider an infinite 2D image $\boldsymbol{X}$ and an infinite 2D filter $\boldsymbol{F}$. The matrix convolution of $\boldsymbol{X}$ and $\boldsymbol{F}$ can be defined as:

$$(\boldsymbol{X} * \boldsymbol{F})(i,j) = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} \boldsymbol{X}(k,l) \boldsymbol{F}(i-k, j-l) \tag{1.23}$$

where $\boldsymbol{X} * \boldsymbol{F}$ is a new 2D image from $\boldsymbol{X}$ filtered by $\boldsymbol{F}$. Such a filter is referred to as a kernel in a CNN, which contains trainable parameters.

In real-world applications, where inputs and filters are of finite dimensions, the formula commonly used is based on a related function called cross-correlation (similar to convolution but without flipping the kernel), which is more straightforward to implement. To illustrate, consider a filter starting at the top-left corner of an image. An operation, typically a sum of products, is performed at the current position. The filter then slides from left to right and top to bottom, applying the same operation at each position. Each operation produces a value for the new filtered image (Figure 1.13).

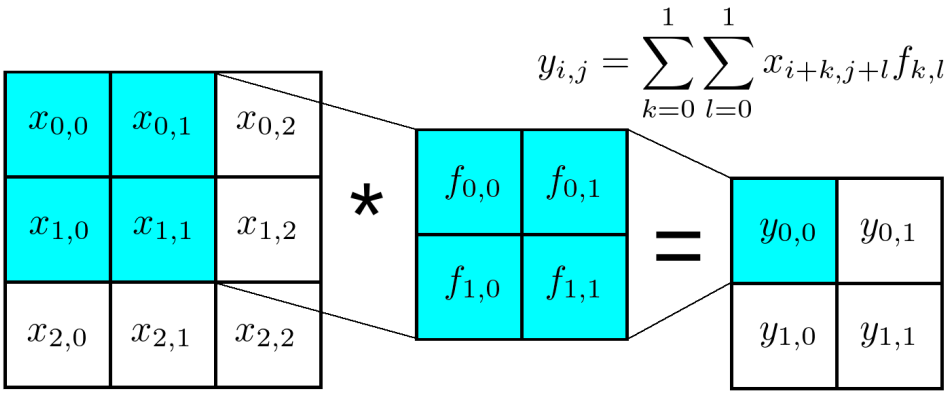


Figure 1.13 – Illustration of a convolution operation applied to an input matrix of 3×3, using a filter of 2×2, with valid padding and a stride of 1.

Other techniques are also employed in the practical implementation of convolution operations. For example, padding involves adding values (typically zeros) around the borders of the input image to control the output dimensions after convolution. Without padding, each convolution operation reduces the spatial dimensions of the feature maps. "Same" padding ensures that the output has the

same spatial dimensions as the input, while "valid" padding uses no padding, resulting in smaller outputs. Stride refers to the step size of the filter as it slides across the input. For instance, a stride of 1 means the filter moves one pixel at a time, while a larger stride (e.g., 2 or 3) reduces the spatial dimensions of the output more rapidly.

In addition to the convolutional operator, a typical CNN also includes:

— Pooling: A downsampling operation that reduces the spatial dimensions of the feature maps, typically by taking the maximum (max pooling) or average (average pooling) value within a local window. Note that stride can be used as an alternative to pooling for downsampling;

— Activation functions: Non-linear functions applied after convolution, such as ReLU, introduce non-linearity into the network;

— Fully connected layers: Often placed at the end of the CNN architecture, these layers, which are typically FCNNs like MLP, flatten the feature maps and connect every neuron to every neuron in the subsequent layer, facilitating the final classification or regression task.

The original CNN introduced by LeCun, Boser, et al. (1989) was initially designed to handle tasks such as 2D image recognition. In his work for digit recognition, images were considered as 2D matrices, where the value at each pixel represented the luminosity or grayscale intensity. This architecture is referred to as a 2D-CNN. The 3D-CNN developed by Krizhevsky et al. (2012) allows handling 3D input data like color images, where the input is a three-dimensional matrix with the third dimension representing the RGB color channels. In 3D-CNN, the key improvement is that the third dimension of the image is captured by multiple filters operating across all input channels. The trainable kernels in a 3D-CNN have depth channels that match the number of input channels, allowing the network to learn patterns that span across different color channels. Each filter produces a single feature map, but by using multiple filters, the network can extract various features from the input.

Throughout the history of DL, the role of CNNs in the resurgence of AI cannot be overstated. As highlighted in the book by Goodfellow, Bengio, et al. (2016), CNNs exemplify the successful application of brain-inspired insights to ML applications. They were among the first deep models to demonstrate strong performance, long before other deep architectures were considered viable. CNNs were also among the first deep networks effectively trained using backpropagation, at a time when many other architectures were considered to have failed (see LeCun, Kavukcuoglu, et al. (2010) and Goodfellow, Bengio, et al. (2016) for a more in-depth history of CNNs and DL). The success of CNNs has paved the way for the development of deep neural networks, where the principles of backpropagation and gradient-based optimization algorithms are crucial.

In the next section, we will explore how backpropagation works to train neural networks using gradient-based optimization algorithms.

1.3.3 Backpropagation and gradient-based optimization

Similar to the conceptual parameters in hydrological models, the neural network parameters (i.e., weights and biases) need to be "trained" to produce outputs that satisfy several criteria, typically defined by a loss function measuring the error between the model outputs and the true values (usually called labels). In ML, a loss function often refers to the model error at a single data point instead of the averaged error over the entire dataset, as a cost function does. This enables training for mini-batches, where the dataset is split into many batches, and for each training epoch, the weights and biases are iteratively updated across the mini-batches. However, in this work, we integrate neural networks into a broader modeling process, where the cost function is computed based on the response of the dynamic system rather than directly on the network outputs and labels. Here, we consider batch training, where the loss computation and weight update are performed over the entire dataset, thus one iteration per epoch.

Let us denote the input points $\boldsymbol{x}_1, ..., \boldsymbol{x}_N$ as N training samples with the corresponding labels $\boldsymbol{y}_1^*, ..., \boldsymbol{y}_N^*$. A neural network Ψ contains $M > 1$ layers that map an input $\boldsymbol{x}$ in the training set to an output $\boldsymbol{y}$ as follows:

$$\boldsymbol{y} = \Psi\left(\boldsymbol{W}; \boldsymbol{x}\right) = \left(\psi_M(\boldsymbol{W}_M) \circ \cdots \circ \psi_1(\boldsymbol{W}_1)\right)(\boldsymbol{x}) \tag{1.24}$$

where $(\psi_i)_{i=1..M}$ are the operations performed at each layer with the associated weight $\boldsymbol{W}_i$ (and also bias, for simplicity, we will only call weight), and $\boldsymbol{W}$ gathers trainable weights from all layers of the neural network. Now, denote $\ell(\boldsymbol{y}_i, \boldsymbol{y}_i^*)$ as a loss function that computes the model error at each training point. The averaged loss function computed over the entire training set is defined as:

$$L(\boldsymbol{W}; \boldsymbol{y}^*) = \frac{1}{N} \sum_{i=1}^{N} \ell\left(\Psi\left(\boldsymbol{W}; \boldsymbol{x}_i\right), \boldsymbol{y}_i^*\right) \tag{1.25}$$

The corresponding optimization problem that finds the trained weights $\hat{\boldsymbol{W}}$ is written as:

$$\hat{\boldsymbol{W}} = \arg\min_{\boldsymbol{W}} L(\boldsymbol{W}; \boldsymbol{y}^*) \tag{1.26}$$

Gradient-based optimization algorithms use the Jacobian matrix (first-order derivative) of the neural network to update the weights (some algorithms also use the Hessian matrix, which is the second-order derivative). These updates are guided by the optimality condition $\nabla_{\boldsymbol{W}} L = 0$, requiring the gradient of the loss function with respect to the weight at each layer. This gradient is computed using backpropagation, which applies the chain rule as follows:

$$\frac{\partial L}{\partial \boldsymbol{W}_i} = \frac{\partial L}{\partial \psi_M} \frac{\partial \psi_M}{\partial \psi_{M-1}} \cdots \frac{\partial \psi_i}{\partial \boldsymbol{W}_i} \tag{1.27}$$

This calculation is feasible due to the differentiability (or at least numerical differentiability) of both the neural network Ψ and the loss function L.

Consider a simple gradient descent algorithm, the weight update is generally written as:

$$\boldsymbol{W}_i^{(j+1)} = \boldsymbol{W}_i^{(j)} - \eta \frac{\partial L}{\partial \boldsymbol{W}_i^{(j)}} \tag{1.28}$$

where $\boldsymbol{W}_i^{(j)}$ and $\boldsymbol{W}_i^{(j+1)}$ respectively represent the current and updated weights, and η is the learning rate that controls the step size of the update. A high value of the learning rate can lead to large updates to the weights, potentially causing the model to converge quickly but risking overshooting the optimal solution and leading to oscillations or divergence. Conversely, a lower learning rate results in smaller updates, which can provide more stable and precise convergence but may require more iterations to reach the optimal solution, thus increasing training time.

Such an optimization process in Equation 1.26 is known as supervised learning, where labeled data is required to compute the loss function. However, several ML methods do not necessarily rely on labels; for instance, K-means clustering (MacQueen, 1967) defines a cluster assignment based on distance metrics and is categorized as unsupervised learning.

More generally, a neural network or ML model can be trained using knowledge of its outputs. This knowledge can be encoded by any differentiable function that satisfies specific criteria (which can be based on external data or observations). The broader optimization problem, which does not necessarily require labeled data, is written as:

$$\hat{\boldsymbol{W}} = \arg\min_{\boldsymbol{W}} (\delta \circ \Psi)\left(\boldsymbol{W}; \boldsymbol{x}\right) \tag{1.29}$$

where δ represents a differentiable function that defines the minimization criteria based on the neural network outputs.

In this sense, unsupervised learning is somewhat closer to the thinking process of a biological brain, as the brain naturally learns by experiencing the world around it without explicit guidance. This means that while guidance or indications exist within the environment in various forms, they may not be immediately recognized. More intelligent brains have a greater capacity to learn from increasingly complex environments where meaningful patterns and relationships are difficult to derive from obvious observations. This interpretation enables the use of neural networks in various contexts and is particularly valuable for geophysical applications, where neural networks can be used as learnable functions and trained on physical knowledge, which represents the key idea behind the hybrid physics-AI model proposed in this work.

One of the challenges in training (deep) neural networks is the "vanishing gradient" problem, where gradients become exceedingly small as they propagate backward through the layers, causing numerical error and ineffective weight updates. Solutions to this problem include the use of alternative activation functions, such as ReLU, and architectural modifications, such as LSTM networks, which are designed to maintain gradients over long sequences (Hochreiter, 1998). Addressing the vanishing gradient problem is not straightforward, as it presents different challenges for each architecture, and remains an active topic in the research community for effectively training deep networks.

1.3.4 Leveraging or over-relying on AI? Understanding its potential and limitations

It has been nearly 80 years since Alan Turing first introduced the concept of a Turing machine, paving the way for the realization of thinking machines (Turing, 1950). Since then, scientists have made impressive efforts to simulate biological brain functions based on mathematical principles and the understanding of natural learning processes. AI models, with their generalization ability to learn multi-level abstractions from large datasets through backpropagation algorithms, have dramatically helped automate various tasks in scientific and engineering applications (LeCun, Bengio, et al., 2015).

Recent research aims to make machine intelligence more realistic by delving into the learning processes of humans and animals, such as reinforcement learning (e.g., Mnih et al. (2015)), which bridges unsupervised and supervised learning by using a reward system to guide decision-making. Additionally, self-supervised learning allows models to learn from unlabeled data by generating their own supervisory signals (e.g., Assran et al. (2023)). Despite these advancements, and as pointed out by LeCun (2022), current AI architectures do not represent (or do so only in a very primitive way) the essential characteristics of intelligent entities. For example, they lack the ability to understand the physical world and do not possess persistent memory. More importantly, they struggle to act correctly in situations they have never encountered. Thus, even state-of-the-art AI models are still far from achieving human-like intelligence (LeCun, 2018).

The question then arises: how can we effectively employ AI in modeling of physical systems given its current capabilities? The answer is to leverage AI's strengths while acknowledging its limitations, depending on the situation and application in which it is used. For example, while AI excels in pattern recognition, data processing, and extracting meaningful information from large datasets, we must avoid the trap of over-reliance on data-driven approaches that may yield impressive scores on benchmark datasets but fail to generalize to the physical complexities of real-world systems under unseen conditions. In geophysical modeling, this trap can be avoided by not solely relying on data-driven models but also incorporating models driven by physical laws and explainable learning processes (Reichstein et al., 2019; Molnar et al., 2020).

Given current advancements in both hydrological modeling and AI, we believe that a comprehensive understanding of complex hydrological processes through a stand-alone AI model is not yet feasible. Even as datasets become more extensive, they remain insufficient to fully capture all relevant physical processes due to observational limitations, biases, and the inherent complexity of hydrological systems (Beven, 2019). Researchers can easily fall into the trap of optimizing for impressive performance metrics on local datasets while failing to develop models that accurately represent underlying physical mechanisms. In this study, we seek to develop more robust, interpretable, and generalizable models for hydrological processes by leveraging AI capabilities within physically-based frameworks.

1.4 Toward a hybrid physics-AI approach within a multi-source data assimilation framework

1.4.1 Research context overview and thesis objectives

The escalating impact of historical and recent flash floods has intensified the need for accurate predictions of these extreme events. Current flood modeling systems necessitate modeling approaches that maximize the use of all available data sources to produce accurate hydrological predictions with high spatio-temporal resolution, achievable through spatially distributed hydrological modeling approaches. Nevertheless, given the complexity and non-linearity of the coupled surface and subsurface physical processes involved, along with their limited observability with respect to the number of parameters to estimate ("curse of dimensionality"), hydrological modeling, particularly spatially distributed modeling, remains a challenging task fraught with uncertainties (e.g., Y. Liu et al. (2007)) and equifinality (Beven, 2006). Moreover, in the absence of directly exploitable first principles in hydrology (e.g., Dooge (1986)), as opposed to flow mechanistic equations in continuous media such as river hydraulics, meteorology or oceanography, and given the high heterogeneities of continental hydrosystems compartments, process-based hydrological models generally include a certain amount of empiricism. This creates a natural opportunity for the fusion of data assimilation and uncertainty quantification with machine learning (ML) and deep learning (DL) techniques to better exploit the informative richness of multi-source data.

A recent explosion of artificial intelligence (AI) applications, stemming from the rise of big data, computational power, and their capabilities to extract multi-level information from large datasets (LeCun, Bengio, et al., 2015), has led to a bloom of studies, in particular in hydrology (e.g., reviews by Nearing et al. (2021) and Shen and Lawson (2021)) and water related disciplines (e.g., Tripathy et al. (2024)). Research on innovative approaches that leverage the power of AI for flood forecasting and hydrological modeling has demonstrated significant efficiency and advantages in terms of accuracy and computational cost compared to traditional rainfall-runoff models (Sit et al., 2020). The use of AI in hydrology is motivated by the recognition that conceptual hydrological models often lack "scale-relevant theories" (Beven, 1987), which can potentially be addressed through data-driven modeling approaches (Reichstein et al., 2019).

In general, there are two main approaches for integration of ML and DL techniques in hydrological modeling. The first approach involves using a full black-box model that replaces traditional rainfall-runoff models. These purely data-driven models have been applied successfully to hydrological prediction, achieving state-of-the-art performance in various applications using long short-term memory (LSTM) network (Kratzert, Klotz, Brenner, et al., 2018; Kratzert, Klotz, Herrnegger, et al., 2019; Feng, Fang, et al., 2020; Cho et al., 2022) and their variants like LSTM-based Seq2Seq model (Xiang et al., 2020). Despite their success, pure ML/DL models, functioning essentially as interpolators with performance and reliability not guaranteed outside their training data range, while also

lacking physical interpretability (Reichstein et al., 2019; Molnar et al., 2020), are generally highly sensitive to large and high-quality training datasets, making them less reliable than process-based models in data-scarce regions (Shen, 2018; Reichstein et al., 2019; Beven, 2020; Sit et al., 2020). Unlike physics-based models, which rely on physical conservation equations of mass, momentum, energy and empirical closures, ML models do not inherently impose physical constraints, resulting in reduced generalizability under extreme or unseen hydrological conditions (Beven, 2020).

A second and more promising approach is the hybrid method, which seamlessly integrates neural networks into process-based numerical models, adhering to the principle of "learning under physical constraints." This strategy aims to harness the predictive power of AI and its multi-scale information extraction capabilities from large databases (LeCun, Bengio, et al., 2015), while maintaining the physical consistency and interpretability of hydrological processes within the context of differentiable modeling (Shen, Appling, et al., 2023). Furthermore, merging process-based differential equations with ML can be highly advantageous, as demonstrated by physics-informed neural networks (PINNs) in Raissi et al. (2019), and by universal differential equations that embed a universal approximator (R. T. Q. Chen et al., 2019; Rackauckas et al., 2021; Yin et al., 2021). In hydrology, the integration of ML into process-based models shows promise, as demonstrated in recent studies on daily lumped models (Kumanlioglu et al., 2019; Jiang et al., 2020; Höge et al., 2022; Feng, J. Liu, et al., 2022). For instance, Kumanlioglu et al. (2019) enhanced the GR4J model by integrating an ANN to dynamically adjust one conceptual parameter of the production reservoir, demonstrating improved prediction performance of the hybrid model compared to the original GR4J as well as the stand-alone ANN (purely data-driven model). Jiang et al. (2020) proposed a symbiotic integration approach that combines DL with geoscience knowledge via a recurrent neural network (RNN), emphasizing physical consistency. Similarly, Höge et al. (2022) introduced neural ODEs, which embed neural networks within differential equation solvers to enhance process representation and adaptability, while Feng, J. Liu, et al. (2022) developed a differentiable process-based model with regionalized parameterization via a LSTM network. However, these approaches do not natively account for spatially distributed information (e.g., detailed meteorological forcings and descriptors of basin physical properties) or fine temporal scales (e.g., hourly or sub-hourly data), both of which are crucial for predicting extreme events characterized by strong variability and short duration. Therefore, hybrid hydrological modeling at high spatio-temporal resolution has received little attention, likely due to model complexity (with coupled processes) and the challenges of optimizing high-dimensional spaces (with gridded conceptual parameters).

The objective of this thesis is to harness AI's predictive capabilities using multi-source data, with AI integration informed by process-based numerical models to address limitations in distributed hydrological modeling at high spatio-temporal resolution. This research bridges two domains: (i) traditional hydrological modeling with physically-based governing equations, numerical solvers, and data assimilation methods developed over decades; and (ii) relatively recent AI techniques offering flexible approximation, along with efficient optimization algorithms, collectively capable of capturing information from multi-source data. Additionally, this thesis addresses other key challenges in calibrating spatially distributed hydrological models from sparse data (regionalization for prediction in ungauged location), especially in the context of high-dimensional optimization problems with weakly constrained cost functions and solution sensitivity to background values. This thesis aims to answer several research questions that progressively build an understanding of the challenges in the modeling process, as well as the rationale behind the application of AI approaches and their efficiency. The research questions are:

— Can research addressing the inverse problem of calibrating a spatially distributed hydrological model, which is inherently an ill-posed problem in the absence of parameter constraints, reduce equifinality and improve hydrological response prediction? Specifically, can adding constraints on flood signatures simulation to the cost function (Chapter 2) or reducing sensi-

tivity to initial optimization conditions (Chapter 4) lead to more robust model calibration?
— Can improvements to the forward model, such as employing physiographic data to constrain conceptual model parameters via a learnable mapping (Chapter 3) or refining internal water fluxes, via another learnable function, to reduce model structure errors (Chapter 5 and Chapter 6), enhance model realism and performance? Furthermore, in such scenarios, why and how can AI be effectively used, and does it preserve the physical interpretability of the process-based model?
— Finally, should efforts focus on developing more complex AI models to partially or fully replace process-based models, or should the priority be on understanding the physical interpretability of hybrid approaches, starting from the simplest AI architectures? (Chapter 7)

Before delving into the details of answering the questions above, the remaining sections of this chapter will present the materials and fundamentals that support the proposed methods in the subsequent chapters. Section 1.4.2 provides a general description of the data that will be used in all studies. Section 1.4.3 outlines the foundational mathematical notation and concepts applied to the methods. Finally, Section 1.4.4 offers an overview of the `smash` platform into which the proposed methods are implemented and built.

1.4.2 Multi-source data at high spatio-temporal resolution

All case studies in this thesis are evaluated using a national database covering Metropolitan France (Figure 1.14) with multi-source data at a relatively high spatio-temporal resolution of $dx = 1$ km and $dt = 1$ h (Table 1.1).

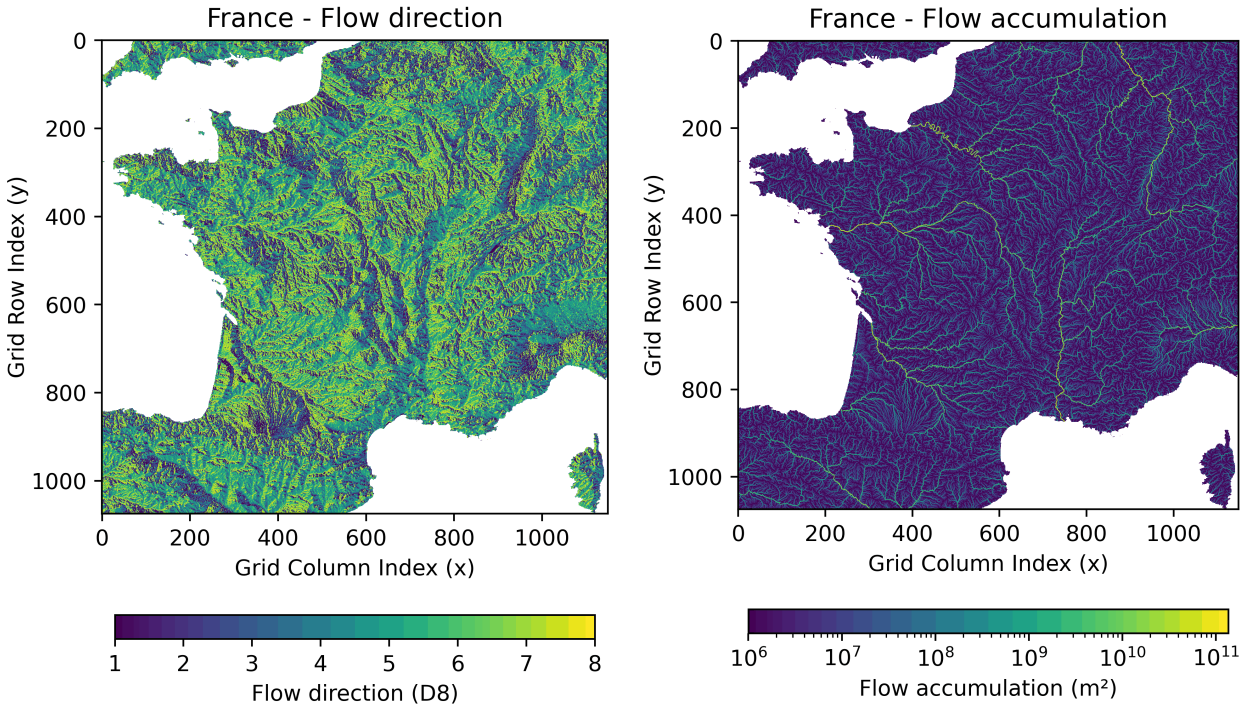


Figure 1.14 – Flow direction (left) and flow accumulation (right) maps, at a relatively high spatial resolution of 1 km^2, covering all of Metropolitan France.

The hydrological models are forced by the following data:
— Observed rainfall grids ($\boldsymbol{P}$): Based on hourly ANTILOPE J+1 radar-gauge observation reanalysis, which merges radar data with in-situ gauge observations (Champeaux et al.,

Table 1.1 – Multi-source data from national open-source databases over Metropolitan France, including atmospheric forcings $\boldsymbol{\mathcal{I}} = (\boldsymbol{P}, \boldsymbol{E})$, the flow direction map $\mathcal{D}_\Omega$, physical descriptors $\boldsymbol{D} = (d_1, ..., d_7)$ for regionalization, and discharge time series $\boldsymbol{Q}^*$.

Notation	Type	Description	Unit	Source
$\boldsymbol{P}$	Atmospheric forcing	Liquid precipitation	mm/h	Champeaux et al. (2009)
$\boldsymbol{E}$	Atmospheric forcing	PET using the Oudin formula (Oudin, Hervieu, et al., 2005)	mm/h	Quintana-Seguí et al. (2008) and Vidal et al. (2010)
$\mathcal{D}_\Omega$	Topography	Flow direction	-	Leblois et al. (1999)
d_1	Topography	Slope preprocessed by Odry (2017)	°	Agency (2016)
d_2	Morphology	Drainage density	-	Organde et al. (2013)
d_3	Influence	Percentage of basin area in karst zone	%	Caruso et al. (2013)
d_4	Land use	Forest cover rate	%	Agency (2019)
d_5	Land use	Urban cover rate (including artificial and non-vegetated areas)	%	Agency (2019)
d_6	Hydrogeology	Potential available water reserve	mm	Poncelet (2016)
d_7	Hydrogeology	High storage capacity basin rate	%	Finke et al. (1998)
$\boldsymbol{Q}^*$	Hydrometric	Discharge time series	m^3/s	HydroPortail

2009). This data is provided by Météo-France at a grid resolution of 1 km^2, matching the resolution of the model grid rasters;

— Potential evapotranspiration (PET, denoted $\boldsymbol{E}$): Computed based on temperature data from SAFRAN reanalysis, provided by Météo-France at a resolution of 8×8 km^2 (Quintana-Seguí et al., 2008; Vidal et al., 2010) and downscaled to a 1×1 km^2 spatial grid using the Oudin formula (Oudin, Hervieu, et al., 2005);

— 2D flow directions map $\mathcal{D}_\Omega$: Obtained from terrain elevation processing and provided by HydroDem (Leblois et al., 1999);

— Physical descriptor maps ($\boldsymbol{D}$): A set of seven descriptors, each at a resolution of 0.01° in the WGS 84 projection, is used as input for the regionalization methods detailed in Chapter 3. These descriptors cover various characteristics, including topography, morphology, land use, and hydrogeology. The set consists of: d_1 the local slope, derived from the Copernicus database (Agency, 2016) and preprocessed by Odry (2017); d_2 the drainage density from Organde et al. (2013); d_3 the percentage of basin area in karst zone from Caruso et al. (2013); d_4 the forest cover rate and d_5 the urban cover rate both from the Corine Land Cover database (Agency, 2019); d_6 the potential available water reserve from Poncelet (2016); d_7 the high storage capacity basin rate from Finke et al. (1998);

— Observed discharge time series ($\boldsymbol{Q}^*$): Collected by the French Ministry of Environment, covering the period of the forcing data and extracted from the HydroPortail platform.

1.4.3 Mathematical foundations of the proposed framework

This section presents a general mathematical description to support the later explanation of the proposed approaches in the subsequent chapters of this thesis.

Let $\Omega \subset \mathbb{R}^2$ denote a 2D spatial domain, with $x \in \Omega$ representing the spatial coordinate and $t \in]0, T]$ denoting the physical time. The domain is structured according to an 8-direction ("D8") drainage plan (flow direction obtained from terrain elevation map processing), denoted as $\mathcal{D}_\Omega$. Fundamentally, the hydrological model, denoted as $\mathcal{M}$, is a dynamic operator that, based on the drainage plan $\mathcal{D}_\Omega$, projects atmospheric input fields $\boldsymbol{\mathcal{I}}$—such as spatio-temporal rainfall $\boldsymbol{P}(x,t)$ and evapotranspiration $\boldsymbol{E}(x,t)$—onto the modeled state and flux variables $\boldsymbol{U}$, consisting of surface

discharge $\boldsymbol{Q}$, model states $\boldsymbol{h}$, and internal fluxes $\boldsymbol{q}$, such that:

$$\boldsymbol{U}(x,t) = [\boldsymbol{Q}, \boldsymbol{h}, \boldsymbol{q}]\,(x,t) = \mathcal{M}\left(\mathcal{D}_\Omega, \boldsymbol{\mathcal{I}}(x,t); [\boldsymbol{\theta}, \boldsymbol{h}_0]\,(x)\right) \tag{1.30}$$

where $\boldsymbol{\theta}$ and $\boldsymbol{h}_0$ represent the N_θ conceptual parameters and N_h initial states of the hydrological model, which are spatially distributed and need to be calibrated. More specifically, the hydrological operator $\mathcal{M}$ is constructed by partially composing the following components:

$$\mathcal{M} = \mathcal{M}_{hy}\left(\,.\,,\ \mathcal{M}_{rr}\left(\,.\,,\ \mathcal{M}_{snw}(\,.\,)\right)\right) \tag{1.31}$$

where $\mathcal{M}_{snw}$ is the snow module that produces a melt flux $q_{snw\to rr}(x,t)$ feeding the production reservoir of the rainfall-runoff module $\mathcal{M}_{rr}$. The rainfall-runoff module then produces the runoff flux $q_{rr\to hy}(x,t)$, which is subsequently routed through the hydraulic module $\mathcal{M}_{hy}$ to generate the final discharge output (Figure 1.15). Note that the snow module is not included in the case studies.

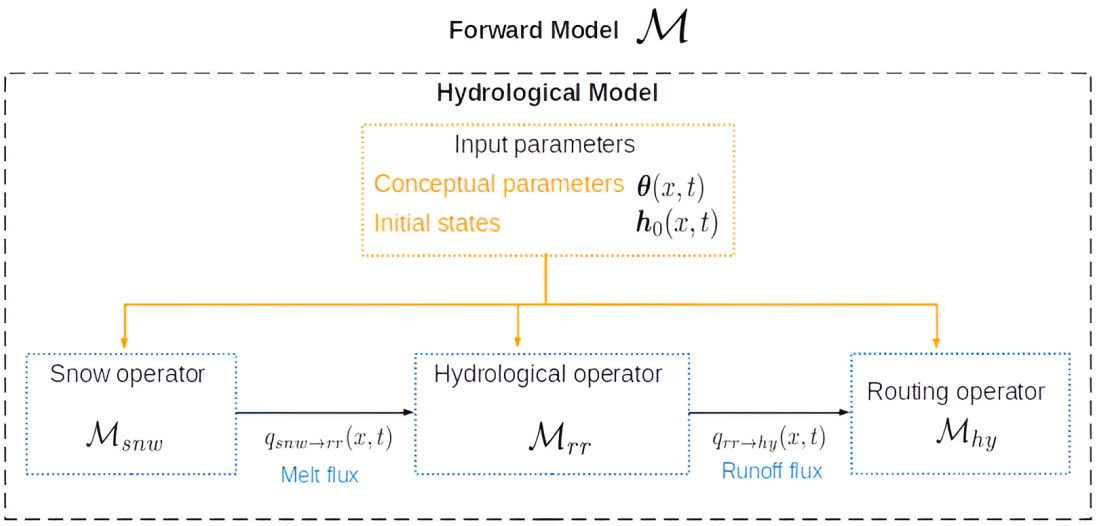


Figure 1.15 – Schematic representation (based on `smash` documentation: Section Forward & Inverse Problems) of the operator composition within the forward model.

Therefore, in this thesis, which focuses on flood modeling in the French Mediterranean region, we use a "zero" snow module (i.e., without snow), effectively neglecting the $\mathcal{M}_{snw}$ operator:

$$\mathcal{M} = \mathcal{M}_{hy}\left(\,.\,,\ \mathcal{M}_{rr}\left(\,.\,\right)\right) \tag{1.32}$$

In general, the inverse problem involves finding the optimal values for the conceptual parameters $\boldsymbol{\theta}$ and/or initial states $\boldsymbol{h}_0$ that minimize a cost function J, which measures the misfit between observed ($\boldsymbol{Q}^*$) and simulated ($\boldsymbol{Q}$) discharges (see Appendix A for commonly used cost functions in hydrological modeling). For simplicity, we present the optimization problem only for the conceptual parameters:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} J\left(\boldsymbol{Q}^*, \boldsymbol{Q} = \mathcal{M}\left(\,.\,, \boldsymbol{\theta}\right)\right) \tag{1.33}$$

Alternatively, the optimization problem can be expressed more concisely as a function of the model parameters $\boldsymbol{\theta}$, as follows:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \tag{1.34}$$

The cost function J is traditionally based on a single hydrological calibration metric, but it can be extended to multi-objective functions that account for hydrological signatures, as presented in

Chapter 2. Moreover, the central contribution of this thesis lies in the integration of a learnable mapping ϕ (comprising two neural networks) into the hydrological model (Figure 1.16) with:

— a descriptors-to-parameters neural network for seamless regionalization of conceptual parameters using physiographic data, which will be presented in Chapter 3;

— a process-parameterization neural network that refines water fluxes in the rainfall-runoff model (Chapter 5), and that is integrated in a numerical resolution scheme (Chapter 6).

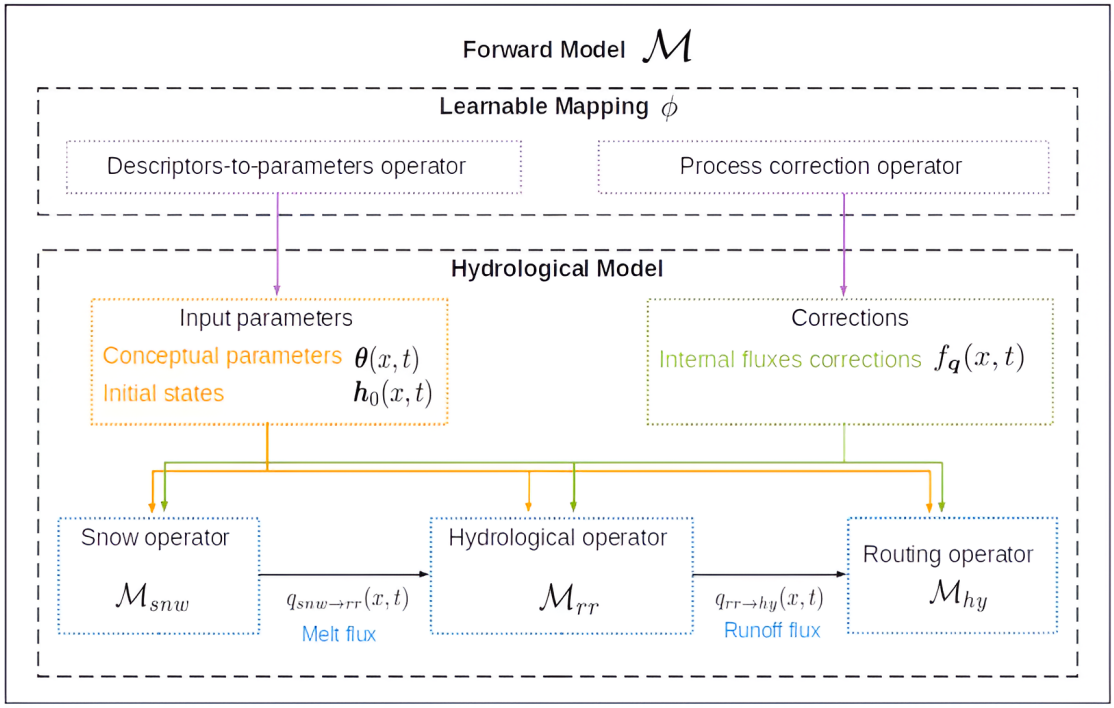


Figure 1.16 – Schematic representation (based on `smash` documentation: Section Forward & Inverse Problems) of the operator composition within the forward model, incorporating a learnable mapping via neural networks.

This learnable mapping uses forcing data $\boldsymbol{\mathcal{I}}$ and physical descriptors $\boldsymbol{D}$ (see Table 1.1) to compute the conceptual parameters $\boldsymbol{\theta}$ (and/or initial states $\boldsymbol{h}_0$) and water flux corrections $\boldsymbol{f}_q$, as briefly expressed in Equation 1.35.

$$[\boldsymbol{h}_0, \boldsymbol{\theta}, \boldsymbol{f}_q] = \phi(\boldsymbol{\mathcal{I}}, \boldsymbol{D}; \boldsymbol{\rho}) \tag{1.35}$$

where $\boldsymbol{\rho}$ represents the hyperparameters of the learnable mapping (i.e., the weights and biases of the neural networks). In this case, the inverse problem in Equation 1.34 needs to be reformulated as an optimization problem with respect to the learnable mapping parameters $\boldsymbol{\rho}$, rather than the conceptual parameters or initial states:

$$\hat{\boldsymbol{\rho}} = \arg\min_{\boldsymbol{\rho}} J(\boldsymbol{\rho}) \tag{1.36}$$

This cost function depends on the sought parameter $\boldsymbol{\rho}$ of the neural network(s) embedded into the hydrological model. All details regarding the calibration process with or without neural networks will be provided in Chapters 2, 3, 4, 5, and 6.

1.4.4 `smash`: Spatially distributed Modeling and ASsimilation for Hydrology platform

A major contribution of this thesis is the implementation and maintenance of the proposed algorithms in `smash`, which has now been developed as open-source Python software built on top of a Fortran core (Colleoni, Huynh, et al., 2025). `smash` has been implemented recently at Service Central VigiCrues and can be used as an operational solver of the French flash flood forecasting system (Piotte et al., 2020).

`smash` contains various hydrological model structures with differing vertical and lateral flow operators as well as spatialized routing schemes. The platform is designed to simulate discharge hydrographs and hydrological states at any spatial location within a gridded domain, and to reproduce the hydrological response of contrasted catchments by leveraging spatially distributed meteorological forcings, physiographic data, and hydrometric observations. Through recent developments, `smash` now integrates hybrid structures and solvers (combinations of AI and numerical models) for learning regionalization and refining model structure errors.

The build system framework of `smash` is illustrated in Figure 1.17. It consists of a Python core for all user interface functions, including both pre- and post-processing, and a Fortran core (with some C files) for high-performance numerical computations. To produce a Python library with binary files that can be installed directly from the package manager, *PyPI*[5], several steps are required. First, Fortran adjoint files are generated from the Fortran sources using the Tapenade automatic differentiation engine (Hascoet et al., 2013), which requires Java. Next, the Fortran code is wrapped for use in Python. *f90wrap* (Kermode, 2020) builds on the capabilities of the popular *F2PY* (Peterson, 2009) utility by generating a simpler Fortran interface to the original Fortran sources that is suitable for wrapping with *F2PY*, along with a higher-level Pythonic wrapper that makes the additional layer transparent to the end user. The entire build system (except for the generation of adjoint files, which remains external for debugging purposes) is managed by *meson*[6], a multi-platform, multi-language open-source build system that enables the generation of `smash` binaries on Linux, macOS, and Windows with relative ease.

Beside the core functionality, `smash` also contains an online documentation (Figure 1.18), that describes the platform comprehensively, containing the four main sections:

- *Getting Started* describes how to install `smash` from the Python package index *PyPI* and set up the necessary environment for running the software;
- *User Guide* provides step-by-step examples from basic to complex applications of `smash` as well as input data conventions and formatting requirements;
- *API Reference* details the different modules and the application programming interface. Modules are documented using the NumPy-style Python docstring format for consistency and ease of access;
- *Math / Num Documentation* details the conceptual and mathematical basis of the forward and inverse modeling problems, their numerical resolution along with optimization and estimation algorithms implemented in the platform.

5. `https://pypi.org`
6. `https://mesonbuild.com`

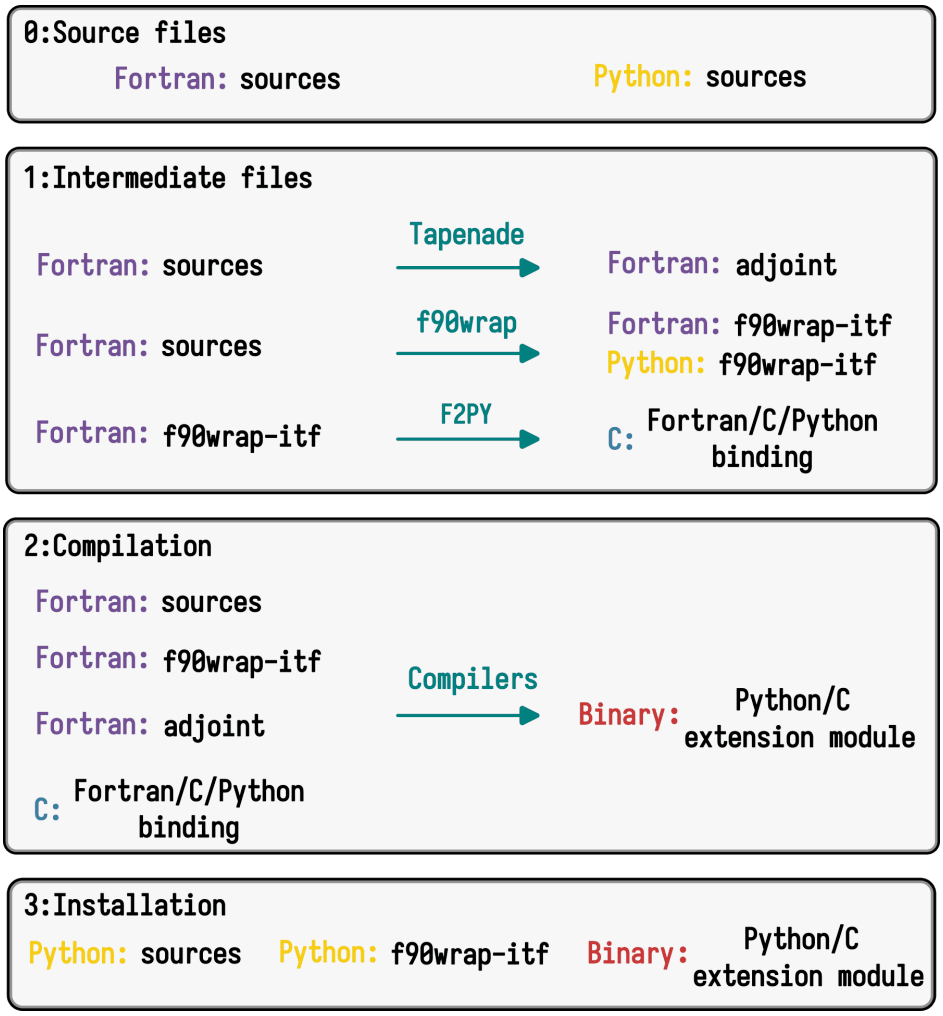


Figure 1.17 – The `smash` build system framework, presented in Colleoni, Huynh, et al. (2025), consists of: (0) Source files written in Fortran and Python; (1) Fortran sources are processed by Tapenade to generate the adjoint code, and wrapped using *f90wrap* to create Python interfaces (*f90wrap-itf*). The *F2PY* tool is then used to generate a Fortran/C/Python binding from the wrapped interfaces; (2) The original Fortran sources, adjoint code, and *f90wrap-itf* are compiled with appropriate compilers to produce a binary Python/C extension module. Finally, in the installation step; (3) The Python module is assembled, combining the original Python sources with the *f90wrap-itf* and the compiled binary Python/C extension module, making the high-performance Fortran code accessible from Python.

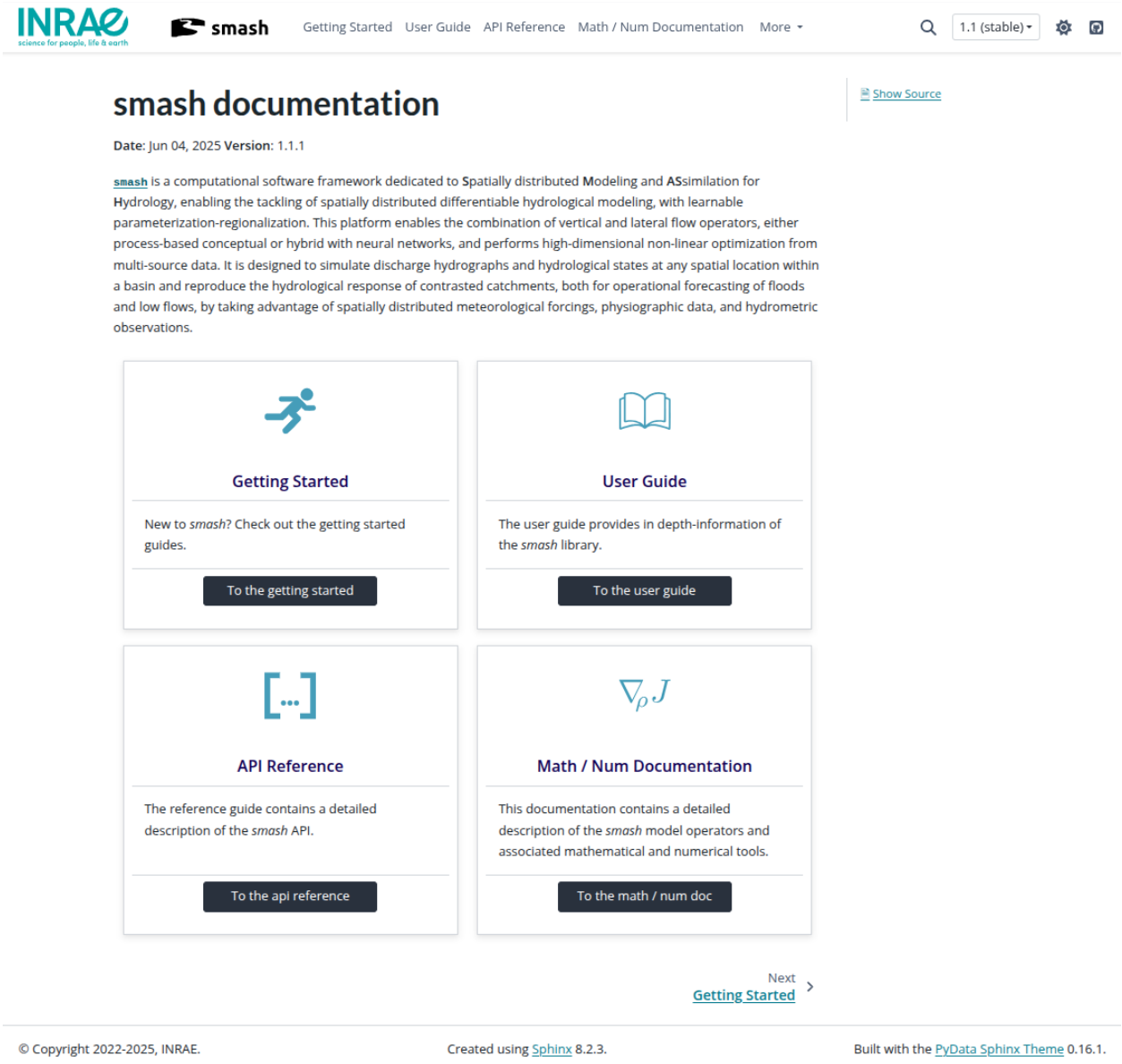


Figure 1.18 – `smash` documentation home page accessible at `https://smash.recover.inrae.fr`.

The remaining chapters of this thesis are organized as follows: Chapter 2 studies the potential of multi-scale hydrological signatures for sensitivity analysis and model calibration. Chapter 3 proposes a novel seamless regionalization algorithm using neural networks within a VDA framework, addressing the challenge of transferring the model parameters from gauged to ungauged watersheds. Chapter 4 employs a Bayesian-weighted estimation method to address parameter estimation in the context of equifinality. The subsequent two chapters propose general frameworks that embed neural networks for internal flux correction within a state-space system (from analytical (Chapter 5) or numerical (Chapter 6) ODE solutions) to improve model accuracy and robustness. Finally, Chapter 7 concludes this thesis and points out future directions.

Chapter 2

Signatures and Sensitivity-Based Multi-Criteria Variational Calibration for Distributed Hydrological Modeling

Contents

2.1 Introduction . . . 37
2.2 Method . . . 38
2.2.1 Overview of the hydrological model studied and optimization algorithms . . 38
2.2.2 Hydrological signatures . . . 40
2.2.3 Hydrograph segmentation . . . 40
2.2.4 Sensitivity analysis . . . 42
2.2.5 Multi-criteria calibration . . . 43
2.3 Catchment information . . . 44
2.4 Sensitivity analysis and selection of signatures for model calibration . . 46
2.5 Performance comparison of classical and signature-based calibration metrics . . . 47
2.5.1 Spatially uniform calibrations with NSGA . . . 48
2.5.2 Spatially distributed calibrations with VDA algorithm . . . 51
2.6 Conclusion . . . 54

This chapter is adapted from Huynh, Garambois, Colleoni, and Javelle (2023) published in *Journal of Hydrology*, with modifications to align with the overall objectives and structure of this thesis. This initial work aimed to develop an approach that leverages automatically computed hydrological signatures for model calibration, both continuously and at the flood scale over large datasets, enabling variational optimization of spatial parameter fields for distributed models.

2.1 Introduction

Numerical hydrological models are used extensively to simulate catchments responses to atmospheric signals and are a key component of floods forecasting systems where accuracy in terms of peak location, amplitude and timing is crucial. As a matter of facts, hydrological models, whatever their complexity and spatialization, consist in more or less empirical representations of flows through watersheds compartments and contain parameters that cannot be inferred directly from the available observations but can only be meaningfully estimated through a calibration procedure (e.g., Gupta, Beven, et al. (2006) and Vrugt, Ter Braak, et al. (2008)). Such procedures aim to improve the model capability in reproducing the available observations of hydrological responses dynamics by optimizing model parameters.

Nevertheless, the whole construction process of a hydrological model is faced with the issue of equifinality: distinct model structures and/or parameter sets can lead to similar (in a sense to be defined) simulations. The equifinality concept has been popularized in hydrology by Beven (1993) while the issues of uncertainty in determining environmental model structures and estimating their parameters were known (e.g., M. B. Beck (1987) and Yeh (1986)). For a given hydrological model structure, the calibration of its parameters is in general an ill-posed inverse problem with non unique solutions. As a consequence, the definition of an optimization algorithm and of a calibration metric is an essential modeling decision. Indeed, it determines how hydrological information is seen and learnt in the calibration process and it can substantially affect the quality and consistency of model simulations.

The present contribution studies the use of hydrological signatures for model calibration, especially for optimization of spatial fields of distributed models (i.e., high-dimensional optimization problem) within a variational data assimilation (VDA) framework (Jay-Allemand, Javelle, et al., 2020), to enhance flood modeling capabilities. Indeed, hydrological signatures are "quantitative metrics that describe statistical or dynamical properties of hydrologic data series, primarily streamflow" allowing to extract "meaningful information about watershed processes" (McMillan, 2021). In this context, the following crucial ingredients are introduced along with relevant literature:

- cost function definition particularly for emphasizing information from flood signatures;
- time-varying signatures computation which requires streamflow signal segmentation and analysis, as well as signatures selection based on their global sensitivity to model parameters which provides valuable guidance for the choice of calibration metrics;
- signature-based calibration approaches linked to multi-criteria calibration.

All these ingredients are integrated in the proposed framework to bring time-varying signatures into global sensitivity analysis and multi-criteria calibration, addressing the following aspects that have received relatively little attention in prior studies:

- the need for an automated segmentation method applicable to large contrasted catchment-floods samples and capable to capture hydrological information at the scale of flash flood events;
- a global analysis of simulated errors across various hydrological signatures and their sensitivity with the model parameters;
- the need for a more intelligent approach to select the Pareto optimal solution in the case of

optimization with multi-objective functions;
— the computation of the cost function based on signatures within a VDA algorithm capable to deal with large spatialized parameter vectors.

2.2 Method

This chapter proposes a calibration strategy that leverages hydrological signatures and their sensitivity analysis, as illustrated in Figure 2.1.

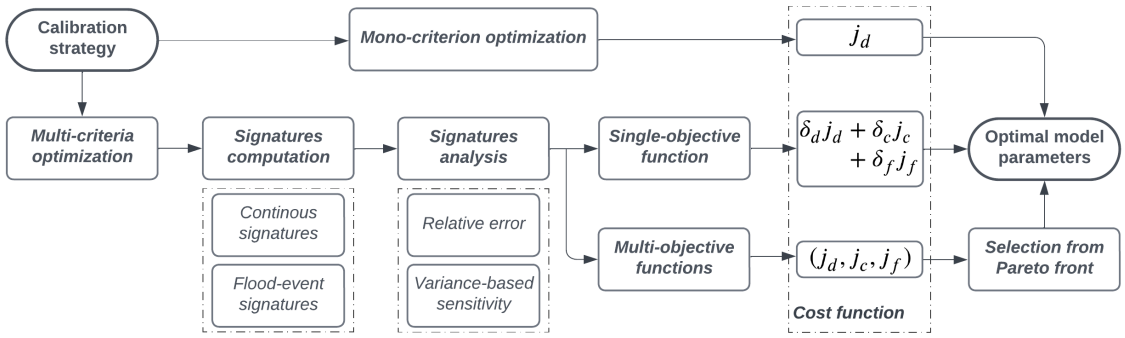


Figure 2.1 – Flowchart of the multi-criteria calibration process using hydrological signatures. The different cost functions are denoted by j_d, j_c and j_f, while the corresponding optimal weights are denoted by δ_d, δ_c and δ_f. The notations used in the cost function will be explained in 2.2.5.

The computations of the signatures are first performed to quantify their sensitivities with the model parameters following Horner (2020). These computations involve performing both whole-period-based analysis to obtain continuous signatures and event-based analysis to capture the most significant events (flood event signatures). Through this analysis, we gain a more meaningful understanding of the parametric sensitivity, not only for discharge but also for other factors that need to be considered as part of the minimization criterion. Furthermore, we evaluate the sensitivity of signature error using variance-based sensitivity analysis (Sobol indices) to determine the most appropriate signatures for multi-criteria optimization. Based on these results, we conduct a multi-criteria optimization with single-objective or multi-objective functions, utilizing suitable hydrological signatures to improve the simulation performance.

2.2.1 Overview of the hydrological model studied and optimization algorithms

In this study, we consider the hydrological model from Equation 1.30 in Chapter 1 using a linear routing module $\mathcal{M}_{hy}$. Since this routing module is a simple linear operator applied to the runoff flux from the rainfall-runoff module $\mathcal{M}_{rr}$, we omit its notation for simplicity. Consequently, the forward model from Equations 1.30 and 1.32 simplifies to:

$$[\boldsymbol{Q}, \boldsymbol{h}, \boldsymbol{q}](x,t) = \mathcal{M}_{rr}\left(\mathcal{D}_\Omega, \boldsymbol{\mathcal{I}}(x,t); \boldsymbol{\theta}(x)\right) \tag{2.1}$$

which focuses solely on the rainfall-runoff process. Here, we use a spatially distributed rainfall-runoff model with six conceptual parameters ($N_\theta = 6$), constructed by applying GR operators (Perrin et al., 2003) at the pixel scale (Figure 2.2). The conceptual hydrological parameters $\boldsymbol{\theta}$ consist of:
— the interception reservoir capacity (c_i in [mm]);
— the production reservoir capacity (c_p in [mm]);
— the main (right) transfer reservoir capacity (c_t in [mm]);

— the left transfer reservoir capacity (c_l in [mm]);
— the non-conservative water exchange parameter (k_{exc} in [min/dt]);
— the linear routing parameter (l_{l_r} in [min]).

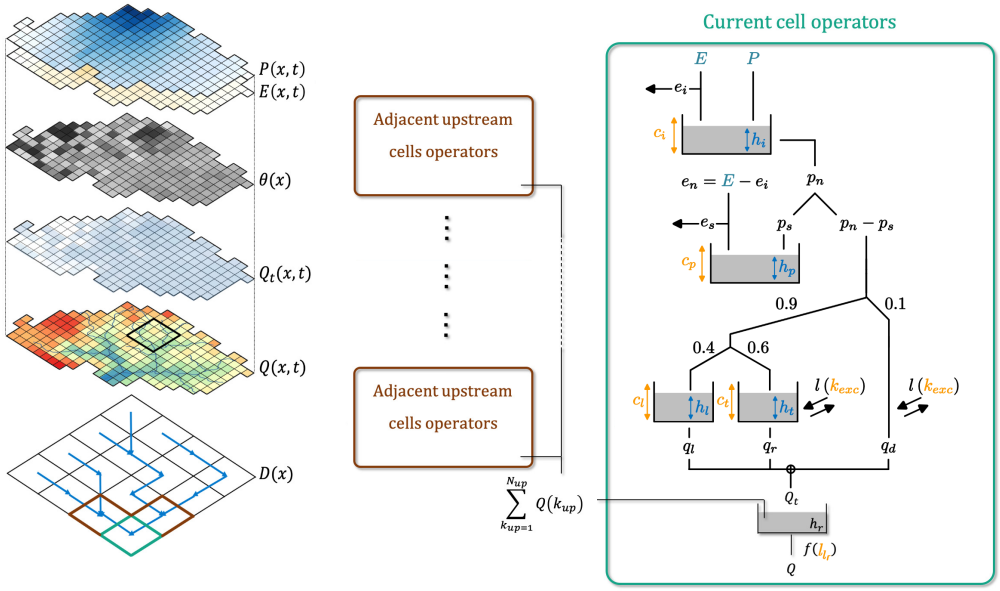


Figure 2.2 – Spatially distributed hydrological model studied in the `smash` platform; (left) Model fields from top to bottom: meteorological inputs, conceptual parameters, internal and output flux maps; (right) Pixel scale and pixel-to-pixel flow operators of the model structure with 6 parameters, where $\boldsymbol{\theta}(x) \equiv (c_i(x), c_p(x), c_t(x), c_l(x), k_{exc}(x), l_{l_r}(x))^T$.

The optimal estimate $\hat{\boldsymbol{\theta}}$ of the model parameters is obtained by solving the inverse problem in Equation 1.34. In practice, bound constraints on the calibrated parameters is included, leading to the following constrained optimization problem:

$$\hat{\boldsymbol{\theta}} = \underset{\bar{\boldsymbol{\theta}}_l \leq \boldsymbol{\theta} \leq \bar{\boldsymbol{\theta}}_u}{\arg\min} J(\boldsymbol{\theta}) \tag{2.2}$$

where $\bar{\boldsymbol{\theta}}_l$ and $\bar{\boldsymbol{\theta}}_u$ denote the lower and upper bounds, which are spatially uniform. This inverse problem is addressed using various global optimization algorithms under a spatially uniform control, resulting in a low-dimensional optimization problem. Examples of applicable optimization algorithms include Step-By-Step (SBS by Michel (1989)) and genetic algorithms such as NSGA (Deb et al., 2002). For high-dimensional optimization, when optimizing a spatially distributed control vector $\boldsymbol{\theta}(x)$, the VDA framework in Jay-Allemand, Javelle, et al. (2020) is employed, where the limited-memory Broyden–Fletcher–Goldfarb–Shanno bound-constrained (L-BFGS-B) algorithm is used, as it is well-suited for high-dimensional problems (Zhu et al., 1997). This algorithm requires the gradient of the cost function with respect to the sought parameters, $\nabla_{\boldsymbol{\theta}} J$, which is obtained by solving the adjoint model. The numerical adjoint model is generated using the automatic differentiation engine Tapenade (Hascoet et al., 2013), adapted to the `smash` source code. Additionally, a uniform first guess (background value) $\bar{\boldsymbol{\theta}}^*$, obtained through a "rapid" optimization of spatially uniform parameters using a global optimization algorithm, serves as the starting point for the optimization problem with high-dimensional control parameters. Given the mildly non-linear nature of the hydrological models considered in this study, this calibration approach is appropriate, with limited sensitivity to prior values, as demonstrated in Jay-Allemand, Javelle, et al. (2020) (see Chapter 4 for methods enhancing the first guess with Bayesian-weighted estimates).

2.2.2 Hydrological signatures

Hydrological signatures, denoted $\mathcal{H}_i$, are introduced to describe and quantify properties of discharge time series, with the aim of analyzing and calibrating hydrological models. These signatures allow the description of various aspects of the rainfall-runoff behavior, such as flow distribution (e.g., based on flow percentiles), flow dynamics (Le Mesnil, 2021), flow separation (Nathan et al., 1990; Lyne et al., 1979), and flow timing, among others. A so-called continuous signature is a signature that can be computed over the entire study period. Flood event signatures on the other hand focus on the behavior of the high-flows that are observed in flash flood events (Figure 2.3).

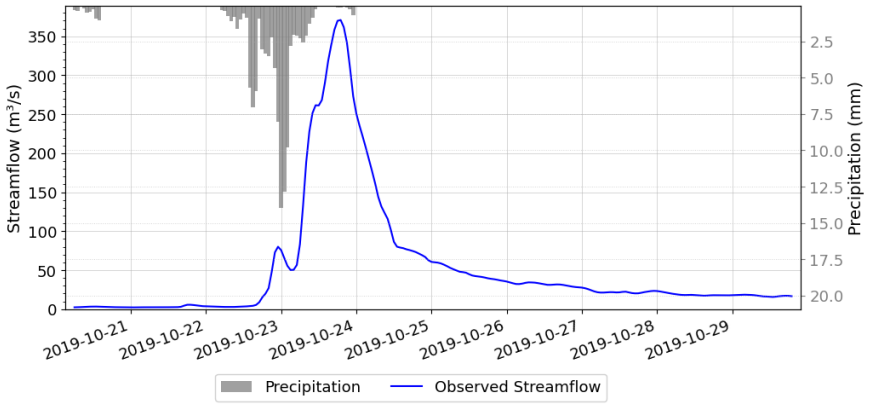


Figure 2.3 – Example of a flood event observed at the outlet gauge of the Aude River in France.

Denote $P(t)$ and $Q(t)$ are the rainfall and runoff at time $t \in \mathbf{U}$, where $\mathbf{U}$ is the study period. Then $Qb(t)$ and $Qq(t)$ are the baseflow and quickflow computed using a classical technique for streamflow separation (please refer to Lyne et al. (1979) and Nathan et al. (1990) for more details). The continuous signatures are calculated over the entire study period as Table 2.1. For an event that occurs within a period $\mathbf{E} \subset \mathbf{U}$, the flood event signatures are calculated as Table 2.2.

Table 2.1 – List of all studied continuous signatures.

Notation	Signature	Description	Formula	Unit
Crc	Runoff coefficients	Coefficient relating the amount of runoff to the amount of precipitation received	$\frac{\int^{t\in\mathbf{U}} Q(t)dt}{\int^{t\in\mathbf{U}} P(t)dt}$	-
Crchf		Coefficient relating the amount of high-flow to the amount of precipitation received	$\frac{\int^{t\in\mathbf{U}} Qq(t)dt}{\int^{t\in\mathbf{U}} P(t)dt}$	-
Crclf		Coefficient relating the amount of low-flow to the amount of precipitation received	$\frac{\int^{t\in\mathbf{U}} Qb(t)dt}{\int^{t\in\mathbf{U}} P(t)dt}$	-
Crch2r		Coefficient relating the amount of high-flow to the amount of runoff	$\frac{\int^{t\in\mathbf{U}} Qq(t)dt}{\int^{t\in\mathbf{U}} Q(t)dt}$	-
Cfp2	Flow percentiles	0.02-quantiles from flow duration curve	$quantile(Q(t), 0.02)$	mm
Cfp10		0.1-quantiles from flow duration curve	$quantile(Q(t), 0.1)$	
Cfp50		0.5-quantiles from flow duration curve	$quantile(Q(t), 0.5)$	
Cfp90		0.9-quantiles from flow duration curve	$quantile(Q(t), 0.9)$	

2.2.3 Hydrograph segmentation

The computation of multi-scale signatures from hydrological time series, typically at flood event scale, requires a segmentation approach. Although the concept of flood event is widely used in

Table 2.2 – List of all studied flood event signatures.

Notation	Signature	Description	Formula	Unit
Eff	Flood flow	Amount of quickflow in flood event	$\int^{t\in\mathbf{E}} Qq(t)dt$	mm
Ebf	Base flow	Amount of baseflow in flood event	$\int^{t\in\mathbf{E}} Qb(t)dt$	mm
Erc	Runoff coefficients	Coefficient relating the amount of runoff to the amount of precipitation received	$\frac{\int^{t\in\mathbf{E}} Q(t)dt}{\int^{t\in\mathbf{E}} P(t)dt}$	-
Erchf		Coefficient relating the amount of high-flow to the amount of precipitation received	$\frac{\int^{t\in\mathbf{E}} Qq(t)dt}{\int^{t\in\mathbf{E}} P(t)dt}$	-
Erclf		Coefficient relating the amount of low-flow to the amount of precipitation received	$\frac{\int^{t\in\mathbf{E}} Qb(t)dt}{\int^{t\in\mathbf{E}} P(t)dt}$	-
Erch2r		Coefficient relating the amount of high-flow to the amount of runoff	$\frac{\int^{t\in\mathbf{E}} Qq(t)dt}{\int^{t\in\mathbf{E}} Q(t)dt}$	-
Elt	Lag time	Difference time between the peak runoff and the peak rainfall	$\arg\max_{t\in\mathbf{E}} Q(t) - \arg\max_{t\in\mathbf{E}} P(t)$	h
Epf	Peak flow	Peak runoff in flood event	$\max_{t\in\mathbf{E}} Q(t)$	mm

hydrology, there is no clear consensus on approaches for flood detection from continuous streamflow time series, as pointed out in Tarasova et al. (2018). Several studies have suggested segmentation algorithms for detecting flood events (refer to the references in Tarasova et al. (2018)). For instance, K. Li et al. (2022) and Astagneau, Bourgin, Andréassian, and Perrin (2021) used simple segmentation methods respectively involving fixed time windows before and after rainfall events or discharge thresholds to detect events. Meanwhile, Tarasova et al. (2018) developed an algorithm incorporating, baseflow separation technique (see also Pelletier et al. (2020)), rainfall attribution methods and an iterative procedure to identify single-peak components of multiple-peak events. In this study, we propose an automated segmentation algorithm, consisting of, peak detection in discharge series, catchment rainfall time series analysis through a combination of rainfall gradients and rainfall energy criterion, which enables a robust determination of flood start time on contrasted catchment-floods, and a classical baseflow separation for determining the end of an event.

The proposed segmentation algorithm is illustrated in Algorithm 1. First, we identify event peak discharges using a peak detection algorithm, which allows for several parameters to be set, such as minimum peak height (mph) or minimum distance between two successive peaks (mpd), among others (Duarte et al., 2021). For instance, we consider events that exceed the 0.995-quantile of the discharge as important events (mph criterion), and events are considered to be distinct if they are separated by at least 12 hours (mpd criterion). Subsequently, we determine the starting and ending dates for each event. The starting date of the event is considered to be the moment when the rain starts to increase dramatically, which is sometime 72 hours before the peak discharge. To calculate this, we compute the gradient of the rainfall and choose the peaks of rainfall gradient that exceed the 0.8-quantile. These peaks correspond to the moments when there is a sharp increase in rainfall. However, we also require an additional criterion called the "energy criterion", which takes into account the "rainfall energy" for a more robust detection of flood start time. The rainfall energy is computed as the sum of squares of the rainfall observed in a 24-hour period, counted from 1 hour before the peak of rainfall gradient. The starting date is the first moment when the rainfall energy exceeds 0.2 of the maximal rainfall energy observed in the 72-hour period before the peak discharge, based on the gradient criterion. Finally, we aim to find the ending date by using baseflow separation. We compute the difference between the discharge and its baseflow from the peak discharge until the end of study period (which lasts for 10 days from the starting date of the event). The ending date is the moment when the difference between the discharge and its baseflow is minimal in a 48-hour period, counted from 1 hour before this moment. Note that these values

are adapted to the basins and flood scales studied.

Algorithm 1 Hydrograph segmentation algorithm

For each catchment, considering 2 time series (T, Q) and (T, P), where: $T = (t_1, ..., t_n)$ is time (by hour), $Q = (q_1, ..., q_n)$ is the discharge, and $P = (p_1, ..., p_n)$ is the rainfall.

1. Detecting peaks that exceed the 0.995-quantile of the discharge, that can be considered as important events: $E = (t_i)_{1 \leq i \leq n}$ s.t. $q_i > Quant_{0.995}(Q)$
2. For each event $t_j \in E$:
   (a) Determining a starting date based on the "rainfall gradient criterion" and the "rainfall energy criterion":
      i. Selecting rainfalls gradient those exceed its 0.8-quantile, considered as the "rainfall events": $RE = (t_k)_{t_k \in (t_j - 72, t_j)}$ s.t. $\nabla P(t_k) > Quant_{0.8}(\nabla P([t_j - 72, t_j]))$
      ii. Defining the rainfall energy function: $f(t_x) = ||(p_x - 1, ..., p_x + 23)||_2$, then the starting date is the first moment the rainfall energy exceeds 0.2 of the maximal rainfall energy: $sd = \min(t_s)_{t_s \in RE}$ s.t. $f(t_s) > 0.2||(f(t_j - 72), ..., f(t_j))||_\infty$
   (b) Determining an ending date based on discharge baseflow $Qb = Baseflow(Q)$: $ed = \arg\min_{t_e} \sum_{t=t_e-1}^{t_e+47} |(Q - Qb)(t)|$ s.t. $t_j \leq t_e \leq sd + 10 \times 24$

Remark. If there exists $m + 1$ $(m > 0)$ consecutive events $(sd_u, ed_u), ..., (sd_{u+m}, ed_{u+m})$ occurring "nearly simultaneously", that means all of these events occur in no more than 10 days: $ed_{u+m} < sd_u + 10 \times 24$, then we merge these $m + 1$ events into a single event (sd_u, ed_{u+m}).

2.2.4 Sensitivity analysis

To perform a calibration process with hydrological signatures, it is important to investigate the sensitivity of simulated signatures with the model parameters, to guide the potential selection of the signatures which should be used to calibrate the model. The sensitivity analysis enables us to examine how the variation of a given output/signature can be apportioned to a variation in model inputs (Saltelli, 2002). If certain signatures are not sensitive with the model parameters, then it may not have any significant impact to optimize an objective function based on these signatures. In this context, we consider a hydrological model $\mathcal{M}$ with N_θ spatially uniform parameters $\bar{\boldsymbol{\theta}} \equiv (\theta_1, ..., \theta_{N_\theta})$. Then the simulated value of a hydrological signature $\mathcal{H}_i$, calculated from the simulated discharges via a discharge-to-signature mapping f_i, is represented as $\mathcal{H}_i^s \equiv f_i \circ \mathcal{M}\left(., \bar{\boldsymbol{\theta}}\right)$. We are interested in Sobol indices called first-order and total-order. The first- (depending on θ_j) and total- (depending on $\theta_{\sim j}$, i.e., all parameters except θ_j) Sobol indices of the simulated signature $\mathcal{H}_i^s$ are respectively defined as follows:

$$S_i^{(1j)} = \frac{\mathbb{V}[\mathbb{E}[\mathcal{H}_i^s|\theta_j]]}{\mathbb{V}[\mathcal{H}_i^s]} = \frac{V_j}{V} \quad \text{and} \quad S_i^{(Tj)} = \frac{\mathbb{E}[\mathbb{V}[\mathcal{H}_i^s|\theta_{\sim j}]]}{\mathbb{V}[\mathcal{H}_i^s]} = 1 - \frac{\mathbb{V}[\mathbb{E}[\mathcal{H}_i^s|\theta_{\sim j}]]}{\mathbb{V}[\mathcal{H}_i^s]} = 1 - \frac{V_{\sim j}}{V}$$

where V_j (respectively, $V_{\sim j}$) is the variance of the expectation of output signature $\mathcal{H}_i^s$ conditioned by the input parameter θ_j (respectively, $\theta_{\sim j}$, i.e., all sampled inputs except θ_j). To estimate these indices, Azzini et al. (2021) proposed a method based on the Saltelli generator (Saltelli, 2002), which is implemented in the *SALib* Python library (Iwanaga et al., 2022; Herman et al., 2017). This method, that is shown to be relatively accurate in a recent benchmark (Puy et al., 2022), allows us to estimate the first-, second- and total-order variance-based sensitivity indices using Monte Carlo simulations. However, in our specific application with high-dimensional parameter spaces,

we have encountered significant challenges in estimating the second-order variance-based sensitivity indices due to their computationally intensive nature (Saltelli, 2002; Campolongo et al., 2011). To achieve accurate results, a large number of Monte Carlo simulations are required, which can be time-consuming and computationally demanding. Therefore, for the purpose of this study, we focus on estimating the first- and total-order Sobol indices, which provide a sufficiently efficient means of capturing information about interaction effects while retaining an acceptable computational cost.

2.2.5 Multi-criteria calibration

First, we define cost function parts corresponding respectively to classical metrics, continuous signatures and event based signatures. Let us consider a classical objective function j_d, which is the dominant criterion (or the most constrained criterion) in case of multi-criteria optimization, an objective function j_c combining continuous-signatures-based cost functions, and j_f combining flood-event-signatures-based cost functions. Then, the cost function to be minimized, denoted J, can be defined as Equation 2.3.

$$J \equiv \begin{cases} \delta_d j_d + \delta_c j_c + \delta_f j_f \text{ for single-objective optimization,} \\ (j_d, j_c, j_f) \text{ for multi-objective optimization} \end{cases} \tag{2.3}$$

where $\delta_d, \delta_c, \delta_f$ are the corresponding optimization weights in the first case. Keep in mind that we take into account the use of signatures in both cases but the first case is a single-objective optimization while the second is a multi-objective optimization.

Then we detail how each cost function part is computed from signatures. For each signature $\mathcal{H}_i$, let $\mathcal{H}_i^o$ and $\mathcal{H}_i^s$ denote the observed and simulated values, respectively. The set of continuous and flood event signatures denoted N_c and N_f, respectively. Then, the components j_d, j_c and j_f can be defined as follows:

- $j_d \equiv 1 - NSE$ or $1 - KGE$ (refer to Appendix A for common calibration/evaluation metrics used in hydrological modeling). This metric j_d is considered as a constraining objective function for selecting an optimal solution from non-inferior solutions in case of multi-objective optimization (see Appendix B.3).
- $j_c \equiv \begin{cases} \sum_{\mathcal{H}_i \in N_c} \sigma_{\mathcal{H}_i} j_c^{\mathcal{H}_i}, \text{ for single-objective multi-criteria optimization;} \\ (\{j_c^{\mathcal{H}_i}\}_{\mathcal{H}_i \in N_c}), \text{ for multi-objective optimization} \end{cases}$
  where $j_c^{\mathcal{H}_i} \equiv \left| \frac{\mathcal{H}_i^s}{\mathcal{H}_i^o} - 1 \right|$ is the objective function based on continuous signature $\mathcal{H}_i$ and $\sigma_{\mathcal{H}_i}$ is the corresponding optimization weight of $\mathcal{H}_i$ in case of single-objective function.
- $j_f \equiv \begin{cases} \sum_{\mathcal{H}_i \in N_f} \sigma_{\mathcal{H}_i} j_f^{\mathcal{H}_i}, \text{ for single-objective multi-criteria optimization;} \\ (\{j_f^{\mathcal{H}_i}\}_{\mathcal{H}_i \in N_f}), \text{ for multi-objective optimization.} \end{cases}$
  In this case, and in the context of global optimization in time, $j_f^{\mathcal{H}_i} \equiv \frac{1}{N_E} \sum_{e=1}^{N_E} \left| \frac{\mathcal{H}_{i,e}^s}{\mathcal{H}_{i,e}^o} - 1 \right|$ defines the scalar objective function related to flood signature $\mathcal{H}_i \in N_f$ over the N_E events selected with the segmentation method described in Algorithm 1. Otherwise, to perform a season-based optimization on flood event signatures, we can compute for the events occurring in the selected season. For example, for a Spring-based optimization:

$$j_{f,spring}^{\mathcal{H}_i} \equiv \frac{1}{\dim \mathcal{Y}_1} \sum_{e \in \mathcal{Y}_1} \left| \frac{\mathcal{H}_{i,e}^s}{\mathcal{H}_{i,e}^o} - 1 \right| \text{ s.t. } \forall e \in \mathcal{Y}_1 \subset \{1, ..., N_E\}, \mathcal{H}_{i,e} \text{ occurs in Spring.}$$

Finally, these cost functions enable to formulate, after the single objective calibration problem, the following multi-objectives calibration problems. The optimization problems taking into account

signatures via the cost function defined in Equation 2.3 can be developed as Equation 2.4 for a single-objective optimization, and as Equation 2.5 for a multi-objective optimization.

$$\min_{\boldsymbol{\theta}\in\mathcal{O}\subset\mathbb{R}^n} \delta_d j_d(\boldsymbol{\theta}) + \delta_c \sum_{\mathcal{H}_i\in N_c} \sigma_{\mathcal{H}_i} \left| \frac{\mathcal{H}_i^s(\boldsymbol{\theta})}{\mathcal{H}_i^o(\boldsymbol{\theta})} - 1 \right| + \delta_f \sum_{\mathcal{H}_i\in N_f} \sigma_{\mathcal{H}_i} \frac{1}{N_E} \sum_{e=1}^{N_E} \left| \frac{\mathcal{H}_{i,e}^s(\boldsymbol{\theta})}{\mathcal{H}_{i,e}^o(\boldsymbol{\theta})} - 1 \right| \tag{2.4}$$

$$\min_{\boldsymbol{\theta}\in\mathcal{O}\subset\mathbb{R}^n} \left(j_d(\boldsymbol{\theta}), \left\{ \left| \frac{\mathcal{H}_i^s(\boldsymbol{\theta})}{\mathcal{H}_i^o(\boldsymbol{\theta})} - 1 \right| \right\}_{\mathcal{H}_i\in N_c}, \left\{ \frac{1}{N_E} \sum_{e=1}^{N_E} \left| \frac{\mathcal{H}_{i,e}^s(\boldsymbol{\theta})}{\mathcal{H}_{i,e}^o(\boldsymbol{\theta})} - 1 \right| \right\}_{\mathcal{H}_i\in N_f} \right) \tag{2.5}$$

While the minimization problem with single-objective function 2.4 is accessible for both global and distributed calibration methods, performing a multi-objective optimization as problem 2.5 is sophisticated for distributed calibration considering a spatially distributed control vector adapted to a high-dimensional hydrological optimization problems, and requiring a lot of cost gradient information. When using multi-objective optimization approaches in global calibration, it is possible to find a set of feasible solutions rather than a single optimal solution, as is the case in single-objective optimization (cf. Appendix B). This is achieved through the use of a non-dominated sorting genetic algorithm (NSGA), which is detailed in Appendix B.2. As a result, a set of non-inferior solutions, also known as a Pareto front, can be obtained (see Appendix B.1). To select the optimal solution from the Pareto front, a method is proposed and depicted in Appendix B.3.

Note that the objective functions j_c and j_f related to continuous and flood signatures have also been implemented in Fortran. This implementation imposes strict positivity of their components ($j_c^{\mathcal{H}_i}$ and $j_f^{\mathcal{H}_i}$) numerically to ensure that the total cost J remains convex and differentiable. The numerical adjoint model has been also re-derived as needed by the variational calibration algorithm. The cost function based flood event signatures j_f can be computed thanks to a temporal mask of corresponding flood events selected by the segmentation algorithm, implemented in the Python routines, and passed to the Fortran routines via the wrapped interface.

2.3 Catchment information

A relatively large dataset of catchment-floods mostly located in the French mediterranean region is used. This dataset stems from Jay-Allemand (2020) and contains time series of hydro-meteorological variables and time invariant catchment attributes for four high rainfall-flow areas in France, identified as study areas of the PICS research project [1]. It encompasses 141 catchments including 23 outlet gauges, which are mostly located in the French Mediterranean region (Figure 2.4). This is a subset of a larger dataset of 4,190 French catchments from INRAE-HYCAR research unit (Brigode et al., 2020; Delaigue et al., 2020). The hydrological model inputs consist of observation data, covering a period of about 13 years (2006 to 2019), that includes hourly distributed discharge and rainfall (see Section 1.4.2 in Chapter 1). Note that observation data, rainfall grids and discharge time series, over the selected catchments have few missing data as detailed in Table 2.3, so that it can be neglected when performing the computations and analysis in this study. Table 2.3 contains catchment information such as the river name, surface, code, number of upstream gauges, and missing rates in the outlet gauges.

1. https://pics.ifsttar.fr

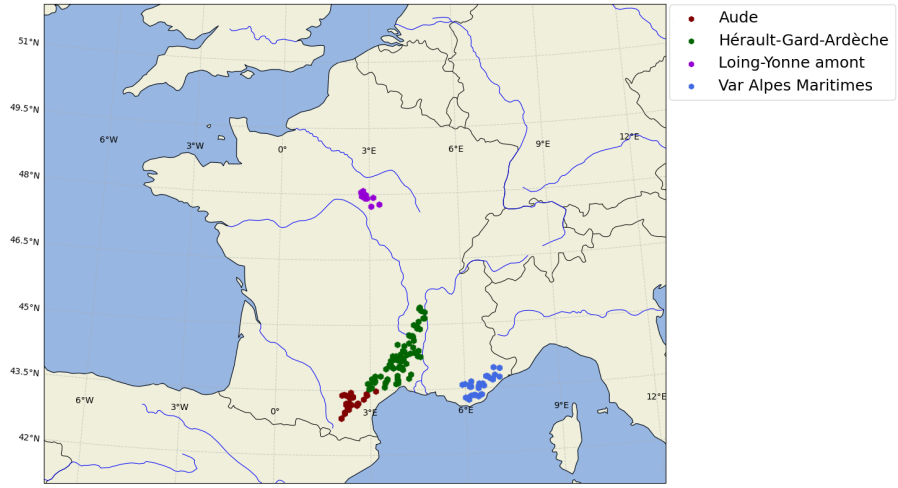


Figure 2.4 – Spatial distribution of 141 catchments of the PICS dataset, consisting of 23 outlet gauges and 118 upstream gauges on the map of France with four regions denoted by different colors.

Table 2.3 – Overview of 23 outlet gauges in the PICS dataset. The columns (left to right) represent the gauge code, river name, surface area, missing rate of rainfall (resp. discharge) at the outlet gauge (for the period 2006–2019), and the number of upstream gauges.

Code	River name	Surface (km^2)	Missing rates (%)	Total upstream gauges
H3201010	Le Loing	2302	0.14 (3.68)	8
V3524010	La Cance	381	0.14 (4.31)	3
V3744010	Le Doux	621	0.14 (4.02)	2
V4154010	L'Eyrieux	649	0.14 (7.38)	3
V5064010	L'Ardèche	2264	0.14 (4.22)	9
V5474015	La Cèze	1112	0.14 (3.76)	6
V7164015	Le Gardon	1093	0.14 (16.62)	10
Y1232010	L'Aude	1828	0.14 (3.74)	11
Y1364010	Le Fresquel	935	0.14 (3.74)	4
Y1415020	L'Orbiel	242	0.14 (3.74)	2
Y1564010	L'Orbieu	589	0.14 (3.77)	3
Y1605050	La Cesse	251	0.14 (4.64)	1
Y2332015	L'Hérault	2208	0.14 (7.22)	12
Y2584010	L'Orb	1336	0.14 (4.04)	11
Y3204040	Le Lez	168	0.14 (15.55)	3
Y3444020	Le Vidourle	503	0.14 (7.97)	4
Y3534010	Le Vistre	496	0.14 (4.42)	1
Y4624010	Le Gapeau	535	0.14 (3.79)	6
Y5312010	L'Argens	2512	0.14 (5.08)	10
Y5444010	La Giscle	201	0.14 (9.96)	2
Y5534030	La Siagne	492	0.14 (5.30)	5
Y5615030	Le Loup	289	0.14 (3.79)	1
Y6434010	L'Estéron	442	0.14 (7.70)	1

2.4 Sensitivity analysis and selection of signatures for model calibration

To start with, the relative error is analyzed between observed signatures and simulated ones with a model calibrated using SBS algorithm and spatially uniform parameters. Table 2.4 shows that some hydrological signatures with a significant simulation error such as: $Cfp2$, $Cfp10$, $Cfp50$, Elt and Epf that could be better constrained with a signature-based calibration process as investigated in next subsection (hydrological signatures with corresponding notations are presented in Table 2.1 and Table 2.2).

Table 2.4 – Relative error between simulated and observed signatures of the same model structure calibrated either with $1 - NSE$ or $1 - KGE$ by SBS algorithm for global optimization. The values (in the form of . [., .]) in each case represent respectively the median, mean and standard deviation of a signature over gauged catchments.

Notation	Signature type	Relative error on simulated signature	
		Cal. with j_d^{NSE}	Cal. with j_d^{KGE}
Crc	Continuous runoff coefficients	0.14 [0.28, 0.38]	0.16 [0.3, 0.46]
Crchf		0.24 [0.35, 0.35]	0.26 [0.4, 0.45]
Crclf		0.15 [0.3, 0.44]	0.15 [0.33, 0.54]
Crch2r		0.23 [0.4, 0.68]	0.22 [0.38, 0.69]
Cfp2	Flow percentiles	0.72 [3.99, 21.14]	0.76 [5.99, 29.98]
Cfp10		0.52 [2.64, 8.8]	0.52 [2.87, 9.42]
Cfp50		0.29 [0.49, 0.85]	0.2 [0.52, 0.99]
Cfp90		0.21 [0.37, 0.96]	0.18 [0.38, 0.99]
Eff	Flood flow	0.23 [0.32, 0.31]	0.19 [0.31, 0.37]
Ebf	Base flow	0.22 [0.33, 0.39]	0.22 [0.33, 0.41]
Erc	Flood event runoff coefficients	0.2 [0.28, 0.26]	0.18 [0.27, 0.26]
Erchf		0.23 [0.32, 0.31]	0.19 [0.31, 0.37]
Erclf		0.22 [0.33, 0.39]	0.22 [0.33, 0.41]
Erch2r		0.12 [0.19, 0.2]	0.13 [0.2, 0.24]
Elt	Lag time	0.48 [0.96, 1.25]	0.46 [0.82, 1.1]
Epf	Peak flow	0.28 [0.38, 0.35]	0.25 [0.36, 0.41]

Next, we survey the global sensitivity of these signatures with the model parameters. We considered over 10,000 spatially uniform sets of the 6 model parameters, sampled using Saltelli generator (Saltelli, 2002), to estimate the total-order Sobol indices across 23 gauged catchments (catchments downstream outlets of the dataset). Based on the results presented in Table 2.5, it can be observed that the non conservative water exchange parameter k_{exc} and the transfer parameter c_t exhibit the highest sensitivities to the studied signatures, both in terms of first-order and total-order. Our analysis suggests that these two parameters have the most significant impact on the output signatures as a result of their interactions with other inputs. This is in coherence with highest sensitivities found for soil depth and subsurface flow parameters of an event flash flood model found in Garambois, Roux, Larnier, Castaings, et al. (2013) and Garambois, Roux, Larnier, Labat, et al. (2015a) on some catchments of the present set. Conversely, we found that parameters such as the interception c_i and the production of runoff c_p have little to no impact on the simulated signatures. We also observed that continuous signatures exhibit lower sensitivities than flood-event signatures in both first-order and total-order effects. Furthermore, constraining hydrological model by flood

Table 2.5 – Median across gauged catchments of first- (respectively, total-) order variance-based sensitivity indices of the studied signatures to the model parameters.

	c_i	c_p	c_t	c_l	l_{l_r}	k_{exc}
Crc	-0.0 (0.0001)	-0.0004 (0.0004)	0.1336 (1.2998)	0.0006 (0.0002)	-0.0 (0.0)	0.1167 (1.3778)
Crchf	0.0038 (0.0103)	0.0268 (0.1155)	0.3739 (0.8506)	0.0153 (0.0123)	0.0367 (0.1513)	0.2245 (0.7919)
Crclf	-0.0 (0.0001)	-0.0004 (0.0003)	0.1299 (1.3018)	0.0006 (0.0001)	-0.0 (0.0)	0.1142 (1.3844)
Crch2r	-0.0004 (0.0017)	0.0193 (0.0255)	0.1014 (0.1426)	0.1099 (0.2055)	0.1833 (0.2449)	0.3984 (0.5481)
Cfp2	0.0014 (0.056)	0.0002 (0.0024)	0.1283 (1.6008)	0.0 (0.0)	0.0 (0.001)	-0.0026 (1.2871)
Cfp10	-0.0 (0.0001)	-0.0002 (0.0001)	0.128 (1.3353)	0.0002 (0.0)	0.0 (0.0)	0.092 (1.3922)
Cfp50	-0.0001 (0.0001)	-0.0001 (0.0001)	0.1267 (1.315)	0.0005 (0.0001)	0.0 (0.0)	0.1043 (1.3933)
Cfp90	-0.0002 (0.0001)	-0.0006 (0.0015)	0.1329 (1.2483)	0.001 (0.0006)	-0.0001 (0.0002)	0.1512 (1.3817)
Eff	0.0002 (0.0059)	0.0699 (0.1939)	0.306 (0.7807)	0.0242 (0.022)	0.0321 (0.1389)	0.1872 (0.7303)
Ebf	-0.0001 (0.001)	0.0014 (0.0159)	0.144 (1.1914)	0.0018 (0.0019)	-0.0002 (0.0056)	0.162 (1.3146)
Erc	-0.0001 (0.0015)	0.0076 (0.0314)	0.18 (1.1633)	0.0028 (0.0031)	-0.0001 (0.0011)	0.1705 (1.2433)
Erchf	0.0002 (0.0059)	0.0699 (0.1939)	0.306 (0.7807)	0.0242 (0.022)	0.0321 (0.1389)	0.1872 (0.7303)
Erclf	-0.0001 (0.001)	0.0014 (0.0159)	0.144 (1.1914)	0.0018 (0.0019)	-0.0002 (0.0056)	0.162 (1.3146)
Erch2r	0.0057 (0.0099)	0.0123 (0.0426)	0.0873 (0.2124)	0.0256 (0.0552)	0.4387 (0.5797)	0.1171 (0.2255)
Elt	-0.0002 (0.0116)	-0.0004 (0.0293)	0.0043 (0.087)	0.0009 (0.0048)	0.8832 (0.953)	0.0127 (0.0568)
Epf	-0.0008 (0.0026)	0.0357 (0.1235)	0.2505 (0.9199)	0.0081 (0.0074)	0.1099 (0.2632)	0.1257 (0.8049)

event signatures along with a classical calibration metric (e.g., $1 - NSE$ or $1 - KGE$), which is based primarily on continuous records of streamflow, is ideal to balance the model between the global score and the performance on flood events. We select for example the peak flow, denoted as Epf, which is one of flood event signatures having both significant relative error and high sensitivity, to perform multi-critera calibration methods. Note that multi-criteria optimization methods with multiple signatures are absolutely reachable but will not be shown in this study for sake of brevity and simplify results analysis.

Furthermore, it is worth mentioning that global sensitivity analysis can be performed with local derivatives based approaches. A link between global Sobol indices and local derivatives has been proposed by Sobol et al. (2010) (refer also to Lamboni et al. (2013)). Global sensitivity matrices in three dimensions (sample size, parameters number, time) and sensitivity statistics, based on local derivatives computed by finite differences have been proposed in Gupta and S. Razavi (2018) and S. Razavi et al. (2019) for geophysical models and applied to HBV-SASK lumped hydrologic model. Note that the VDA algorithm upgraded in the present work uses accurate local (in parameter space) cost function gradients, global in time and spatially distributed, computed with the adjoint method. Such method enables to compute accurate spatial sensitivity maps even for high-dimensional parameter spaces (e.g., Monnier et al. (2016)) and deepening sensitivity analysis with the differentiable and spatially distributed hydrological model, along with accounting for sensitivity indices into the VDA algorithm, is a very interesting direction intentionally left for further research.

2.5 Performance comparison of classical and signature-based calibration metrics

In this section, we compare the performance of different models using both uniform and distributed optimization methods with different calibration metrics, including signature-based ones. For spatially uniform calibration methods, we aim to compare different calibration metrics includ-

ing classical single-objective optimization (CSOO), signature-based single-objective optimization (SSOO) and signature-based multi-objective optimization (SMOO). For spatially distributed calibration methods, two strategies selected for comparison are CSOO and SSOO. In both spatially uniform or distributed calibration scenarios, the models are calibrated on 23 outlet gauges of the PICS data on the calibration period 2006-2013. The validation of calibrated models performances is done in space and time following the three setups:

— on 23 outlet gauges on the period 2013-2019 (temporal validation, T_Val);
— on 118 upstream gauges on the period 2006-2013 (spatial validation, S_Val);
— on 118 upstream gauges on the period 2013-2019 (spatio-temporal validation, S-T_Val).

2.5.1 Spatially uniform calibrations with NSGA

We first perform global calibrations using NSGA with (i) classical single-objective functions; (ii) multi-criteria single-objective functions; and (iii) multi-objective functions. Table 2.6a displays the mean of different objective functions for calibration and validation (with 3 validation metrics), and for 3 optimization methods (CSOO, SSOO and SMOO) with various cost functions. In CSOO, we interpret that the model calibrated with $j_d^{KGE} = 1 - KGE$ produces a better result on the peak flow j_f^{Epf}, compared to the one calibrated with $j_d^{NSE} = 1 - NSE$. This explains why KGE criterion is more robust than NSE for constraining a hydrological model, since it is built on the decomposition of NSE (Gupta, Kling, et al., 2009), which emphasizes relative importance of several hydrological features. This finding is consistent with that of Mizukami, Rakovec, et al. (2019). The authors calibrated daily models over numerous US catchments with multiple metrics, including NSE, weighted KGEs, annual peak flow bias (APFB), and they found that KGE resulted in better estimates of annual peak flows than NSE. Additionally, the best reproduction of annual peak flows was achieved with APFB, but this was at the expense of other high flow metrics.

Using event signatures in addition to classical continuous metrics in SSOO, we found that simulated peak flow is highly improved in terms of relative error (about 15-18 times and 1.4-1.7 times on average, respectively, for calibration and temporal validation) while classical calibrated metrics are significantly deteriorated (about 1.4-1.6 times and 1.4-1.5 times on average, respectively, for calibration and temporal validation). This may arise from imbalances between global score and performance in simulating flood event signature. To address this issue, careful consideration of the optimization weights assigned to objective functions is necessary in order to achieve a balance between model performance on short and long-term series. It should be noted that this approach can be time-consuming, as it requires numerous simulations to determine the appropriate optimization weights for the objective functions, typically using a L-curve approach. Alternatively, the use of global calibration algorithms, which do not require gradient information and can be solved using lower-dimensional optimization problems, can also address these imbalances through the application of a multi-objective optimization approach. This approach offers the advantage of keeping acceptable levels of deterioration of NSE and KGE while significantly improving the simulation of peak flow as shown by multi-objective SMOO results in Figure 2.5 and 2.6.

However, this global multi-objective optimization algorithm is not capable to deal with high-dimensional control vectors and the spatially uniform parameter setup here (under-parameterization) led to unsatisfactory results in spatial and spatio-temporal validation metrics. Therefore, a distributed calibration approach, such as using the VDA algorithm accounting for signatures, could improve the model performances. This approach maintains the same optimization weights as described above, and its performance will be evaluated in the subsequent section.

As shown in Figure 2.7, the corresponding optimal parameters obtained using various optimization strategies are presented. Based on the preliminary analysis, it is evident that the distribution over studied catchments of l_{l_r} has an important difference when performing traditional calibration

Table 2.6 – Calibration, temporal, spatial, and spatio-temporal validation metrics for calibration methods with (a) spatially uniform parameters (CSOO, SSOO, SMOO), and (b) spatially distributed parameters, optimal fit for $cost = 0$. The mean of calibration and validation cost values for different objective functions are displayed for each calibration metric.

Method	Calibration metric	$\overline{j_d^{NSE}}$				$\overline{j_d^{KGE}}$				$\overline{j_f^{Epf}}$			
		Cal	T_Val	S_Val	S-T_Val	Cal	T_Val	S_Val	S-T_Val	Cal	T_Val	S_Val	S-T_Val
(a) Calibration with Spatially Uniform Parameters													
CSOO	j_d^{NSE}	0.274	0.277	0.901	0.616	0.239	0.369	0.687	0.736	0.279	0.324	0.387	0.357
	j_d^{KGE}	0.352	0.330	1.048	0.795	0.183	0.323	0.665	0.721	0.267	0.280	0.379	0.344
SSOO	$j_d^{NSE}/2 + j_f^{Epf}/2$	0.447	0.418	1.056	0.889	0.377	0.476	0.759	0.853	0.014	0.189	0.346	0.372
	$j_d^{KGE}/2 + j_f^{Epf}/2$	0.551	0.431	1.259	0.956	0.335	0.443	0.777	0.833	0.017	0.209	0.337	0.358
SMOO	$\{j_d^{NSE}, j_f^{Epf}\}$	0.341	0.351	1.020	0.845	0.271	0.420	0.703	0.803	0.087	0.215	0.336	0.391
	$\{j_d^{KGE}, j_f^{Epf}\}$	0.456	0.409	1.163	0.821	0.243	0.368	0.683	0.724	0.048	0.182	0.316	0.389
(b) Calibration with Spatially Distributed Parameters													
CSOO	j_d^{NSE}	0.221	0.244	0.655	0.596	0.233	0.355	0.553	0.597	0.274	0.334	0.381	0.376
	j_d^{KGE}	0.239	0.231	0.802	0.702	0.140	0.292	0.617	0.701	0.226	0.295	0.365	0.364
SSOO	$j_d^{NSE}/2 + j_f^{Epf}/2$	0.251	0.241	0.831	0.639	0.231	0.305	0.586	0.612	0.183	0.298	0.392	0.383
	$j_d^{KGE}/2 + j_f^{Epf}/2$	0.297	0.245	0.964	0.671	0.190	0.300	0.617	0.647	0.152	0.271	0.376	0.387

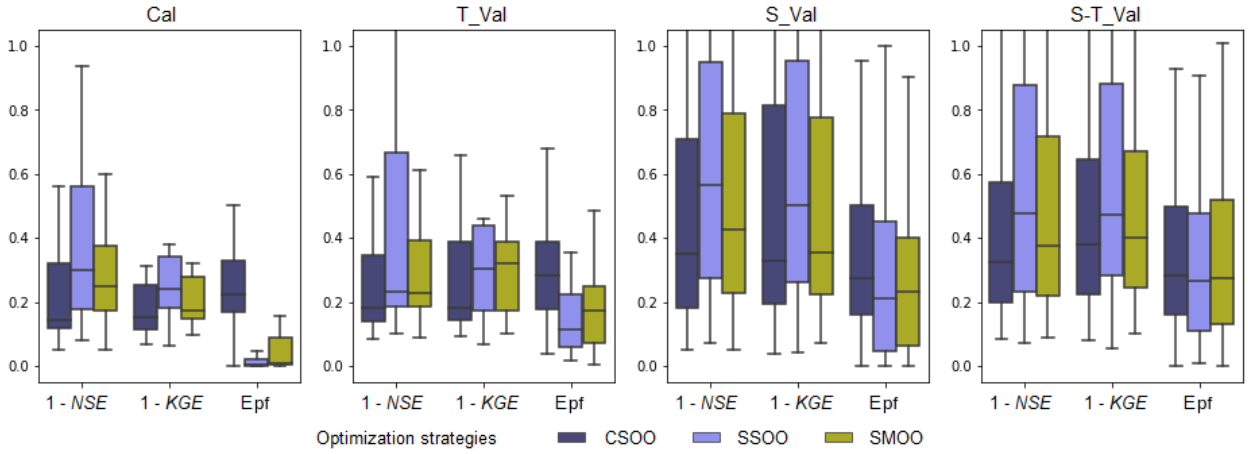


Figure 2.5 – Comparison, with spatially uniform parameters, of calibration and validation metrics (optimal fit for $cost = 0$) for three optimization approaches (CSOO, SSOO, SMOO) by constraining $1-NSE$ in case of global algorithms (SBS or NSGA). From left to right: calibration (Cal), temporal validation (T_Val), spatial validation (S_Val) and spatio-temporal validation (S-T_Val).

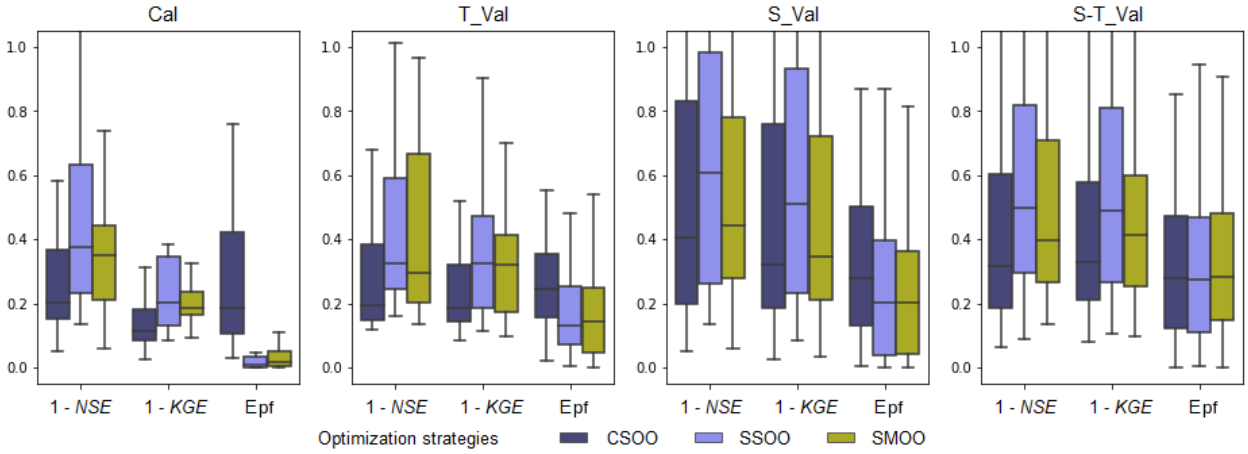


Figure 2.6 – Comparison, with spatially uniform parameters, of calibration and validation metrics (optimal fit for $cost = 0$) for three optimization approaches (CSOO, SSOO, SMOO) by constraining $1-KGE$ in case of global algorithms (SBS or NSGA). From left to right: calibration (Cal), temporal validation (T_Val), spatial validation (S_Val) and spatio-temporal validation (S-T_Val).

(CSOO) and multi-criteria calibration methods (SSOO and SMOO). We recall that l_{l_r} is the routing parameter in the conceptual design, so it plays a crucial role in the production of the peak flow Epf. Additionally, the sensitivity analysis in Table 2.5 has indicated that l_{l_r} is one of the three parameters explaining most of the sensitivity of the peak flow.

The above result on the importance of lateral flow components in a flood hydrological model is in coherence with existing works, for example as shown in Garambois, Roux, Larnier, Castaings, et al. (2013) on few catchments-flood events used in the present study, in addition to high sensitivity to subsurface flow parameter (see also Douinot et al. (2018)), the temporal sensitivity of kinematic wave compound friction parameters in a distributed flash flood model increases with flood magnitude. Improving hydraulic meaningfulness of hydrological models is an important topic since it can improve floods discharge modeling in high resolution catchment-flood models (e.g., Bout et al. (2018), Z. Li et al. (2021), and G. Kirstetter et al. (2021) with shallow water models and sim-

plifications) but also improve internal state-flux coherence and realism as required for instance to assimilate remote sensing observables of river surface such as height and width (e.g., Paiva et al. (2011), Pujol, Garambois, Finaud-Guyot, et al. (2020), and Pujol, Garambois, and Monnier (2022)).

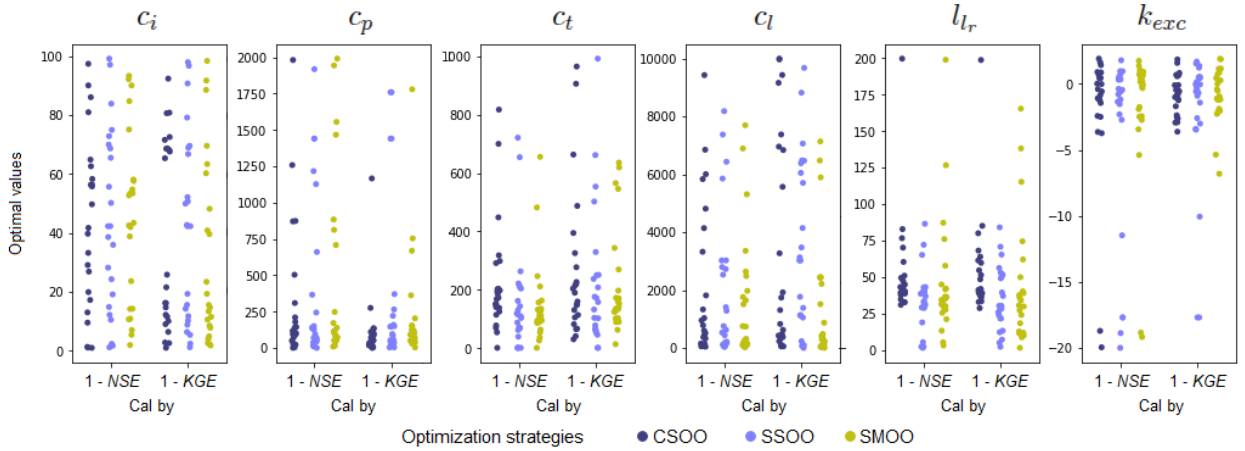


Figure 2.7 – Analysis of spatially uniform calibrated parameters over the whole catchment sample. In each scatterplot, the first column present the distribution of a parameter for 3 optimization strategies (CSOO, SSOO, SMOO) using j_d^{NSE}, whereas the strategies in the second column use j_d^{KGE} as the dominant (or constrained) objective function.

2.5.2 Spatially distributed calibrations with VDA algorithm

Now, spatially distributed calibrations with the VDA algorithm using multi-criteria cost functions, including signatures, are performed. We employ SSOO technique for a distributed calibration using L-BFGS-B algorithm provided a first guess by SBS algorithm. In overall, all of obtained scores in Table 2.6b are significantly enhanced compared to the uniform calibration method, thanks to spatially distributed control vectors granting more flexibility to reproduce observed discharge. Instead of a sharp decline of j_f^{Epf} as above, this relative error slightly decreases about 1.5 times (from about 0.25 down to 0.16) in calibration and from about 0.32 down to 0.28 in temporal validation, but instead, the scores (NSE and KGE) are slightly reduced in calibration and have an inappreciable deterioration in temporal validation. So in this case, we do not have imbalances between the model performances on short and long-term series when employing SSOO. We observe clearly in Figure 2.8 and 2.9 that the error of simulated pick flow is significantly reduced while the deterioration level of the scores remains tolerable, particularly in calibration and temporal validation.

Ultimately, the scoring metrics are computed on 111 flood events picked from 23 outlet gauges (by segmentation method depicted in Algorithm 1) on the calibration period. The results plotted in Figure 2.10 show that, in distributed calibration, the score of constrained calibration metric is not decreased but even improved from 0.80 (respectively, 0.71) up to 0.83 (respectively, 0.78) in median for NSE (respectively, KGE). It indicates that the optimum of the model parameters has moved to another location that produces a better performance in simulating flood events by slightly reducing the scores in simulating the low-flow.

Regarding to the parameter space, Table 2.7 presents statistical quantities, including mean, median and standard deviation, of the spatially uniform parameter sets obtained using 4 calibration metrics for the studied catchments. Comparing to the spatially distributed optimal parameters in Table 2.8, we interpret that the mean of distributed parameters over all catchments in the 4 cases (corresponding to 4 calibration metrics) is globally coherent to the distribution of the first

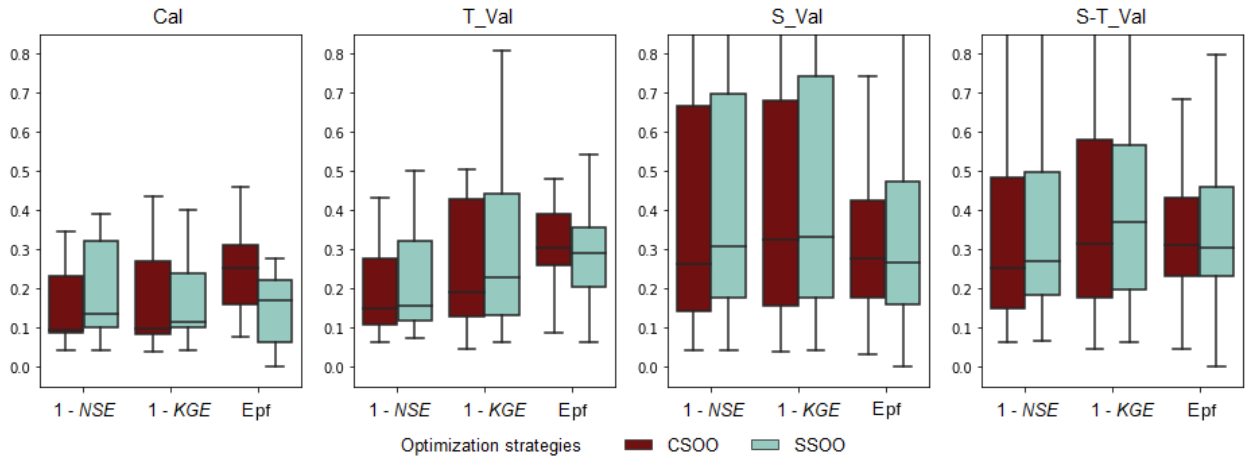


Figure 2.8 – Comparison, with spatially distributed parameters, of calibration and validation metrics (optimal fit for $cost = 0$) for two optimization approaches (CSOO, SSOO) by constraining $1 - NSE$ in case of distributed calibration. From left to right: calibration (Cal), temporal validation (T_Val), spatial validation (S_Val) and spatio-temporal validation (S-T_Val).

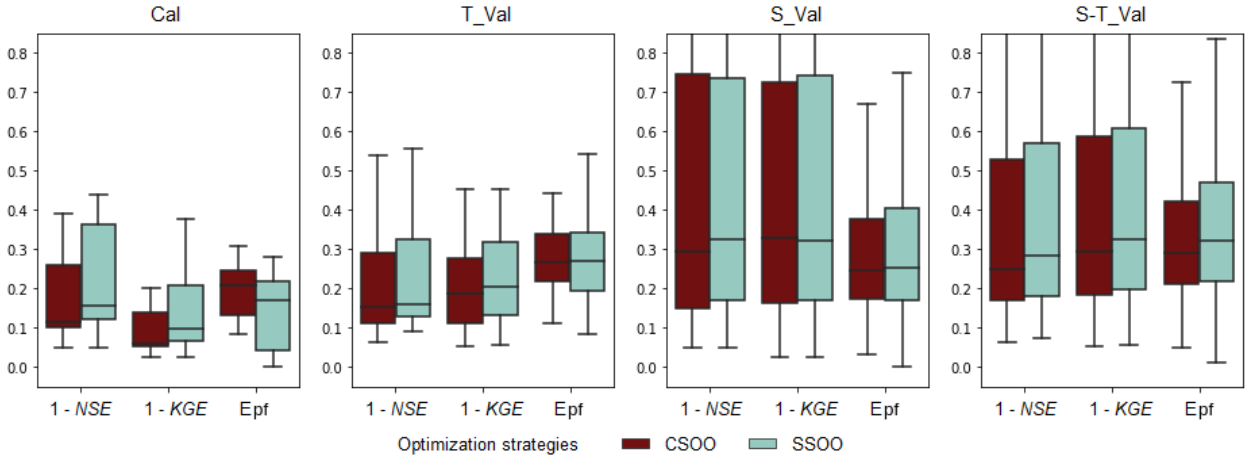


Figure 2.9 – Comparison, with spatially distributed parameters, of calibration and validation metrics (optimal fit for $cost = 0$) for two optimization approaches (CSOO, SSOO) by constraining $1 - KGE$ in case of distributed calibration. From left to right: calibration (Cal), temporal validation (T_Val), spatial validation (S_Val) and spatio-temporal validation (S-T_Val).

guess. Several parameters are almost spatially uniform (e.g., the non conservative water exchange parameter k_{exc} has a small distributed deviation in median (respectively, in average) over all catchments 0.01 (respectively, 0.05) (calibrated with j_d^{NSE}) compared to its distributed mean in median (respectively, in average) -0.59 (respectively, -4.98)). Conversely, the transfer parameter c_l has a great distributed deviation (in median over all catchments) 193.72 compared to its distributed average 347.87, that also has a massive difference to its distributed median 114.79. Figure 2.11 illustrates the spatially distributed optimal parameters at the largest catchment (the Argens River), for a distributed calibration with $j_d^{KGE}/2 + j_f^{Epf}/2$.

Reducing the over-parameterization in distributed hydrological models calibration problems through spatial constrains while enhancing regional parameters consistency remains a key issue, especially for flash flood prediction at ungauged locations (e.g., classical post-regionalization in

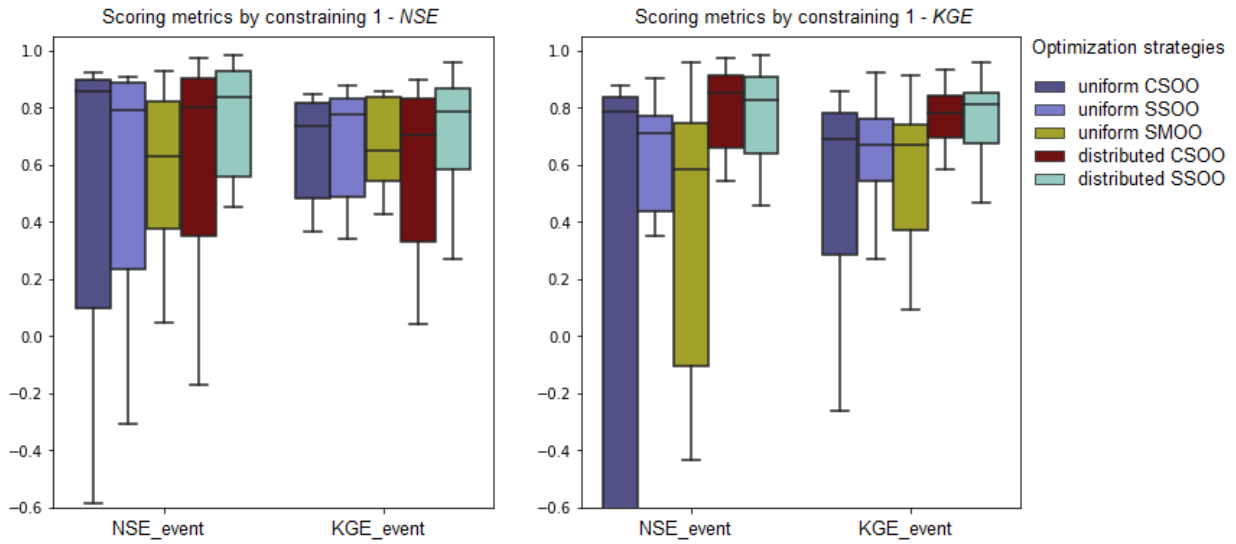


Figure 2.10 – Comparison of scoring metrics computed on 111 events picked from 23 outlet gauges on the calibration period 2006-2013 for the five optimization strategies, by constraining $1 - NSE$ (left) and $1 - KGE$ (right), optimal fit for $score = 1$.

Table 2.7 – Uniform optimal parameters calibrated by SBS algorithm with 4 calibration metrics for each catchment on its outlet gauge. The values (in the form of . [., .]) in each case represent respectively the median, mean and standard deviation of the optimal parameter values over all catchments of the dataset.

	Calibration metric			
	j_d^{NSE}	$j_d^{NSE}/2 + j_f^{Epf}/2$	j_d^{KGE}	$j_d^{KGE}/2 + j_f^{Epf}/2$
c_i	14.71 [20.3, 26.22]	16.93 [20.83, 26.48]	17.6 [27.15, 33.07]	17.27 [30.17, 35.83]
c_p	169.99 [291.17, 434.58]	146.04 [310.14, 505.68]	151.87 [286.79, 483.33]	141.56 [289.69, 466.99]
c_t	171.76 [286.6, 269.5]	162.66 [313.49, 304.83]	266.32 [431.2, 355.04]	267.21 [436.83, 360.62]
c_l	347.87 [812.15, 1274.12]	250.42 [1366.73, 2789.22]	383.51 [1413.93, 2749.35]	262.89 [1337.96, 2795.16]
l_{l_r}	41.32 [52.63, 34.05]	40.94 [50.97, 30.97]	41.33 [51.2, 30.29]	40.24 [50.2, 27.58]
k_{exc}	-0.59 [-4.98, 8.21]	0.0 [-3.81, 7.34]	-0.0 [-3.62, 7.31]	-0.0 [-3.28, 6.39]

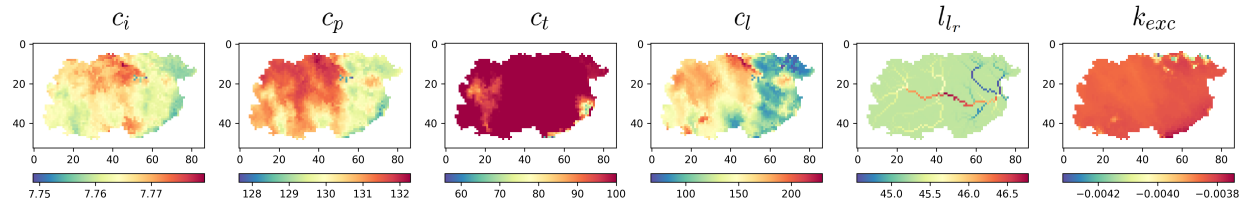


Figure 2.11 – Spatially distributed optimal parameters for the Argens River basin, obtained by minimizing $j_d^{KGE}/2 + j_f^{Epf}/2$.

Garambois, Roux, Larnier, Labat, et al. (2015a) on French Mediterranean flash floods). This issue can be tackled with calibration approaches accounting for physiographic descriptors through regularizations (e.g., De Lavenne et al. (2019) and Jay-Allemand, Demargne, et al. (2024) in multi-gauges

Table 2.8 – Analysis of spatially distributed parameter sets of the models corresponding to 4 calibration metrics. First, spatial median ($\tilde{.}$), average ($\bar{.}$) and standard deviation ($\sigma_.$) for each parameter field are calculated for each catchment, then their median, mean and standard deviation over all catchments are represented in the form of . [., .].

	Calibration metric			
	j_d^{NSE}	$j_d^{NSE}/2 + j_f^{Epf}/2$	j_d^{KGE}	$j_d^{KGE}/2 + j_f^{Epf}/2$
$\tilde{c_i}$	15.45 [20.22, 26.34]	10.91 [20.14, 26.75]	17.6 [27.19, 33.16]	17.3 [30.32, 36.05]
$\overline{c_i}$	14.71 [20.3, 26.22]	16.93 [20.83, 26.48]	17.6 [27.15, 33.07]	17.27 [30.17, 35.83]
σ_{c_i}	0.22 [0.91, 1.46]	0.07 [1.13, 4.16]	0.13 [0.57, 1.28]	0.05 [0.82, 3.09]
$\tilde{c_p}$	161.81 [286.05, 435.48]	145.79 [314.19, 518.0]	156.65 [288.75, 476.53]	148.27 [296.94, 485.79]
$\overline{c_p}$	169.99 [291.17, 434.58]	146.04 [310.14, 505.68]	151.87 [286.79, 483.33]	141.56 [289.69, 466.99]
σ_{c_p}	38.52 [60.58, 59.64]	8.95 [37.57, 44.15]	31.08 [53.49, 57.35]	12.03 [46.82, 102.47]
$\tilde{c_t}$	174.6 [287.44, 270.08]	158.48 [317.5, 311.03]	266.09 [429.0, 353.73]	267.12 [447.27, 372.32]
$\overline{c_t}$	171.76 [286.6, 269.5]	162.66 [313.49, 304.83]	266.32 [431.2, 355.04]	267.21 [436.83, 360.62]
σ_{c_t}	13.88 [28.84, 35.82]	3.23 [25.3, 57.2]	5.68 [24.28, 60.74]	1.45 [24.39, 74.56]
$\tilde{c_l}$	114.79 [675.7, 1276.32]	127.09 [1322.22, 2806.59]	180.63 [1332.45, 2784.42]	146.96 [1322.17, 2803.08]
$\overline{c_l}$	347.87 [812.15, 1274.12]	250.42 [1366.73, 2789.22]	383.51 [1413.93, 2749.35]	262.89 [1337.96, 2795.16]
σ_{c_l}	193.72 [355.92, 433.62]	34.67 [139.21, 252.41]	69.91 [222.54, 388.17]	31.82 [61.87, 81.06]
$\tilde{l_{l_r}}$	41.37 [52.08, 34.42]	41.37 [52.08, 34.42]	41.37 [52.08, 34.42]	41.37 [52.08, 34.42]
$\overline{l_{l_r}}$	41.32 [52.63, 34.05]	40.94 [50.97, 30.97]	41.33 [51.2, 30.29]	40.24 [50.2, 27.58]
$\sigma_{l_{l_r}}$	4.66 [6.01, 5.04]	1.45 [5.17, 10.17]	3.01 [5.31, 10.29]	1.34 [5.08, 13.82]
$\tilde{k_{exc}}$	-0.59 [-4.98, 8.21]	0.0 [-3.72, 7.42]	0.0 [-3.61, 7.31]	-0.0 [-3.27, 6.39]
$\overline{k_{exc}}$	-0.59 [-4.98, 8.21]	0.0 [-3.81, 7.34]	-0.0 [-3.62, 7.31]	-0.0 [-3.28, 6.39]
$\sigma_{k_{exc}}$	0.01 [0.05, 0.09]	0.0 [0.17, 0.73]	0.02 [0.05, 0.09]	0.0 [0.08, 0.34]

calibration problems) or through pre-regionalization mappings, such as the multi-scale parameter regionalization approach (MPR) from Samaniego, Kumar, and Attinger (2010), used for example in Mizukami, Clark, et al. (2017). In addition to exploiting the information of multi-scale signatures in calibration with the present VDA algorithm, the use of a pre-regionalization scheme, i.e., "strong constrains" in the forward model in form of a mapping between physiographic covariables and conceptual hydrological parameter fields, represent an interesting perspective for future research.

2.6 Conclusion

In this study, we enhanced the calibration process of the conceptual distributed hydrological model `smash` for Mediterranean floods by incorporating hydrological signatures and various multi-criteria optimization strategies. First, we computed and analyzed both continuous signatures and flood event signatures. Subsequently, we used sensitivity analysis to select appropriate signatures for constraining the model. Finally, we performed signatures-based multi-criteria optimization approaches, which demonstrated their robustness and reliability in improving simulated peak flood events without significantly compromising the NSE and KGE. Notably, for distributed calibration,

the model constrained by the signature performed better in simulating flood events and achieved higher NSE and KGE scores compared to the model calibrated without using signatures. These results highlight the superiority of signature-based calibration approaches, particularly in flash flood prediction. Furthermore, we compared the parameter spaces of different models to provide insights into the optimal transition from traditional calibration approaches to signature-based calibration methods.

Our proposed calibration strategy addresses the need for an intelligent approach to model calibration in the presence of multiple objectives and complex hydrological processes. This approach offers a new perspective on the hydrological calibration process, accounting for the incorporation of classical discharge metrics and multi-scale signatures, that can provide a more comprehensive assessment of the model performance. As a potential perspective of improvement, the method could be reinforced via the use of multi-source information such as from remotely sensed data products and of multi-gauge streamflow series in regionalization problems. The segmentation algorithm could be tested on larger flood samples, also including catchment rainfall moments (Zoccatelli et al., 2011; Emmanuel et al., 2015) describing rainfall patterns for floods analysis (e.g., Garambois, Larnier, Roux, et al. (2014) and Saharia et al. (2021)) and in order to prepare learning sets for training hybrid flood modeling-correction approaches. Building on these insights, future work will aim to address the issue of equifinality, previously mentioned in the introduction section. The VDA algorithm could be upgraded with Bayesian elements, accounting for signatures and sensitivity. We will also develop better spatial constraints by mapping physiographic descriptors to parameters fields (pre-regionalization). The effectiveness of new approaches could be evaluated and analyzed over large datasets with signatures thanks to the segmentation algorithm. The proposed method could be easily applied to new spatially distributed model structures and hypothesis testing.

Chapter 3

Seamless Regionalization Using Neural Networks within a Differentiable High-Resolution Hydrological Model

Contents

3.1 Introduction 57
3.2 Forward-inverse algorithms within a regionalization framework 59
3.2.1 Forward model with regionalization 60
3.2.2 Calibration cost function accounting for multiple observation sites 61
3.2.3 The variationnal data assimilation algorithm 62
3.2.3.1 Optimization algorithm for polynomial regionalization 62
3.2.3.2 Optimization algorithm for neural network-based regionalization . 63
3.3 Data and numerical experiment 64
3.3.1 Study area and experimental design 64
3.3.2 Regional learning performance 66
3.3.3 Interpretation of the learning process and inferred parameters 73
3.4 Discussions 76
3.5 Conclusion 77

This chapter is based on Huynh, Garambois, Colleoni, Renard, Roux, et al. (2024) published in *Water Ressources Research*, with modifications to align with the overall objectives and structure of this thesis. Since spatially distributed hydrological models require constraints to address over-parameterization issues, as discussed in Chapter 2, the goal here is to develop a regionalization approach, aiming to integrate information from physical descriptors into conceptual spatialized models while leveraging the spatially distributed gradients of the `smash` numerical model, particularly enabling the use of neural networks.

3.1 Introduction

When calibrating a hydrological model, the optimization problem is typically an ill-posed inverse problem faced with the equifinality (Beven, 2001) of feasible solutions, which can be further interpreted as model structural equifinality and spatial equifinality in the context of spatially sparse observations compared to model controls (see for example discussions in Garambois, Larnier, Monnier, et al. (2020)). Most calibration approaches enable the estimation of spatially uniform model parameters for a single gauged catchment, but this leads to piecewise constant and discontinuous parameters fields for adjacent catchments. Moreover, parameter sets determined through calibration are not transferable to ungauged locations although the latter represent the majority of the global land surface (Fekete et al., 2007; Hannah et al., 2011). Therefore, prediction in ungauged basins remains a "grand challenge" (Sivapalan, 2003) in hydrology (Hrachowitz, Savenije, et al., 2013), which hinders the development of effective high-resolution models adapted to the simulation of hydrological extremes in a context of high data uncertainty (e.g., for Mediterranean flash floods in Garambois, Roux, Larnier, Labat, et al. (2015b) and Jay-Allemand, Demargne, et al. (2024)).

The estimation of hydrological model parameters in ungauged regions is performed using a so-called regionalization approach that exploits and transfers hydrological information from gauged to ungauged catchments by utilizing various descriptors of catchment physical properties (see reviews in Blöschl et al. (2013), Samaniego, Kumar, and Attinger (2010), Hrachowitz, Savenije, et al. (2013), and H. E. Beck, M. Pan, et al. (2020)). The most widely used approach in early regionalization studies involves independent catchment-by-catchment calibrations, followed by multiple regression or interpolation techniques to transfer the calibrated parameter sets from gauged to ungauged locations (Abdulla et al., 1997; Seibert, 1999; Parajka, Merz, et al., 2005; T. Razavi et al., 2013; Parajka, Viglione, et al., 2013) and can be called post-regionalization (Samaniego, Kumar, and Attinger, 2010). This approach presumes that the variability of calibrated model parameters through the catchments is related, for instance, to spatial proximity (Widén-Nilsson et al., 2007; Oudin, Andréassian, et al., 2008), and physical or climatic similarity (Oudin, Kay, et al., 2010; H. E. Beck, Dijk, et al., 2016). However, post-regionalization approaches are often limited to lumped parameters, thus ignoring within-catchment variabilities (see reviews in Samaniego, Kumar, and Attinger (2010) and T. Razavi et al. (2013)), except when calibrated parameters correspond to tuning factors of physical pedotranfer functions or hydraulic frictions correspondence tables (see Garambois, Roux, Larnier, Labat, et al. (2015b) for details). The identification of transfer functions in post-regionalization is complicated by the uncertainty of estimated parameter sets, while spatial proximity is mostly applicable to densely gauged river networks and regions (Oudin, Andréassian, et al., 2008; Reichl et al., 2009). Moreover, incorporating a statistical learning process, especially unsupervised learning approaches, in the post-regionalization step can exacerbate existing issues such as parameter biases induced by data measurement errors (Kavetski et al., 2006). A regionalized calibration simultaneously exploiting the information of multiple gauges, within spatial clusters defined a priori from descriptors, is performed in Huang et al. (2019) over Norway using climatic similarity. The parameters calibrated over multiple gauges of a climatic zone are applied to ungauged

catchments of the same zone. This approach does not account for hydrological heterogeneity within the catchments or within the regional clusters determined by physical similarity, which can have a major impact on forecasting, in particular for extreme floods (Garambois, Roux, Larnier, Labat, et al., 2015b; Jay-Allemand, Demargne, et al., 2024).

The simultaneous regionalization approach involves optimizing a transfer function between physical descriptors and model parameters (cf. Hundecha et al. (2004), Götzinger et al. (2007), Bastola et al. (2008), and Samaniego, Kumar, and Attinger (2010)). In this case and contrarily to post-regionalization, the descriptors-to-parameters mapping is the first trainable operator of the forward hydrological model. This approach enables overcoming most of the aforementioned problems and has been applied in several studies. For instance, it has been used for regionalizing semi-distributed models such as HBV in Hundecha et al. (2004) or in Götzinger et al. (2007). A multiscale parameter regionalization (MPR) method, which combines descriptor maps, spatial upscaling functions, and regionalization transfer functions in the form of multivariate mappings from descriptors, was proposed by Samaniego, Kumar, and Attinger (2010). This method, implemented within a spatially distributed multiscale hydrological model (mHM), provides consistent (seamless) parameter and flux fields across scales (Samaniego, Kumar, Thober, et al., 2017), while also imposing a spatial regularization effect through a strong constraint in the forward model (upscaling laws and regionalization transfer functions). Such a regularization is needed when working with spatially distributed hydrological models and spatially sparse discharge data leading to overparameterized optimization problems. In the above studies, state of the art optimization or sampling algorithms are used, typically the shuffle complex evolution algorithm (SCE) (Duan et al., 1992) in Mizukami, Clark, et al. (2017), the distributed evolutionary algorithms (DEAP) (Fortin et al., 2012) in H. E. Beck, M. Pan, et al. (2020), or the GLUE framework with a random sampling approach in Lane et al. (2021). Those algorithms are applicable with low-dimensional controls only, which limits the affordable number of descriptors and the spatialization of regional transfer parameters (that are lumped), and especially, which limits the affordable complexity of the regionalization operator.

This chapter proposes a modeling approach called HDA-PR (Hybrid Data Assimilation and Parameter Regionalization), relied on seamless regional optimization algorithms for learning complex transfer functions between physical descriptors and conceptual hydrological parameters that are spatially distributed. It is designed to exploit the informative content of massive heterogeneous datasets over large spatio-temporal computational domains, and is thus well-suited for solving high-dimensional inverse problems. The approach relies on the combination of the following ingredients:

— Learnable regionalization functions via the introduction into the direct hydrological model of an explicit tunable mapping between heterogeneous physical descriptors and spatially distributed conceptual parameters. This mapping allows estimating parameter values while imposing a constraint on their spatial variability, via the use of physical descriptors and a priori knowledge. Multivariate polynomial regressions and neural networks are employed to learn such a complex non-linear descriptors-to-parameters mapping.

— A differentiable spatially distributed hydrological model into which the regionalization operators have been implemented. This enables the computation of accurate, spatially-distributed gradients of the calibration cost (objective) function, with respect to the sought regionalization parameters, which can be of high dimension. Obtaining accurate gradients for such high-dimensional parameters is crucially needed for optimization algorithms.

The approach aims at ensuring that the hybrid data assimilation algorithm, which integrates an explainable learning process, produces results that can be physically interpreted (Larnier and Monnier, 2020; Höge et al., 2022; Fablet et al., 2021; Althoff et al., 2021). It is able to enhance calibration scores with deep learning from large heterogeneous datasets while maintaining their physical interpretability. The research questions investigated are:

— Does the embedding of neural network-based regionalization transfer functions within a spa-

tially distributed differentiable hydrological model enable using variational data assimilation (VDA) to extract relevant information from physical descriptors and discharge time series at multiple gauges? And does it provide effective spatial constraints to avoid spatial overparameterization in the inverse problem, leading to effective and interpretable regional maps of conceptual hydrological parameters for modeling gauged and ungauged catchments?

— How do neural networks compare, in terms of modeling performances in calibration and spatio-temporal extrapolation, to a simple multi-gauge calibration approach with spatially uniform model parameters or to a linear multivariate regression on physical descriptors?

3.2 Forward-inverse algorithms within a regionalization framework

This section presents the forward model and inverse algorithms of the proposed HDA-PR method. An algorithm flowchart is provided in Figure 3.1 to help in global understanding.

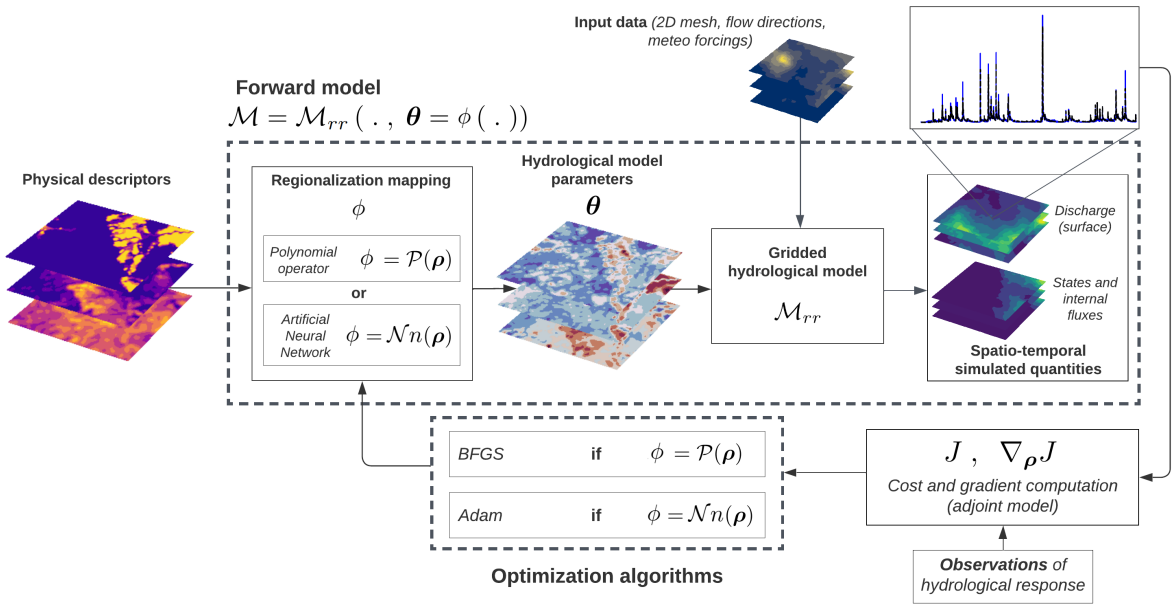


Figure 3.1 – Flowchart of the forward-inverse algorithm used in HDA-PR. The forward hydrological model is a gridded model (spatio-temporal regular grid at 1 km^2 and 1 h) using GR operators (Perrin et al., 2003).

First, the differentiable forward model consists in: (i) a parsimonious and robust spatially distributed GR-like conceptual hydrological model structure, composed of GR hydrological operators ("Génie Rural" lumped models cf. Perrin et al. (2003)) applied at pixel scale for runoff generation, and a simple pixels-to-pixel routing scheme, the whole hydrological model being differentiable (Jay-Allemand, Javelle, et al., 2020; Colleoni, Garambois, et al., 2022); and (ii) regionalization operators, consisting of either multivariate polynomial regressions or neural networks, for mapping descriptors onto hydrological model parameters. The calibration cost function adapted to multi-site (and potentially multi-source) observations is then defined. The inverse optimization algorithms, that use the spatially distributed gradients of the cost function with respect to model parameters, are capable of dealing with high-dimensional inverse problems such as encountered with tunable parameters of regionalization descriptors-to-parameters mappings.

The core strength of HDA-PR is the use of differentiable descriptors-to-parameters transfer functions, especially in the form of neural networks, and the capability to automatically compute accu-

rate cost gradients. The latter enables the use of gradient-based variational optimization algorithms with high-dimensional regional parameter vectors. The method is applicable to any differentiable forward model as well as to multi-source heterogeneous datasets, and hence constitutes a powerful data assimilation framework.

3.2.1 Forward model with regionalization

From the classical hydrological model in Equation 1.32 in Chapter 1, this study introduces a regionalization operator ϕ to estimate the hydrological parameters $\boldsymbol{\theta}$ and initial states $\boldsymbol{h}_0$:

$$\mathcal{M} = \mathcal{M}_{rr}\left(\,.\,,\,[\boldsymbol{\theta}, \boldsymbol{h}_0] = \phi\left(\,.\,\right)\right) \tag{3.1}$$

Note that, as in Chapter 2, this study employs a linear routing module; however, for simplicity, its notation is omitted. Thus, the forward model can be written as:

$$\boldsymbol{U}\left(x,t\right) = \left[\boldsymbol{Q}, \boldsymbol{h}, \boldsymbol{q}\right]\left(x,t\right) = \mathcal{M}_{rr}\left(\mathcal{D}_{\Omega}, \boldsymbol{\mathcal{I}}(x,t); \left[\boldsymbol{\theta}, \boldsymbol{h}_0\right](x)\right) \tag{3.2}$$

In this study, the distributed rainfall-runoff model $\mathcal{M}_{rr}$ is a parsimonious spatialized GR structure (Perrin et al., 2003) with four parameters ($N_\theta = 4$). This model is similar to the 6-parameter structure presented in Chapter 2 (see Figure 2.2), but with only one transfer reservoir instead of two. Additionally, the capacity of the interception reservoir is automatically adjusted using a flux-matching approach (Ficchì et al., 2019). Thus, the model parameters to be calibrated are:

$$\boldsymbol{\theta}\left(x\right) = \left(c_p(x), c_t(x), k_{exc}(x), l_{l_r}(x)\right)^T$$

where the four spatially varying parameter fields are the production reservoir capacity (c_p in [mm]), the transfer reservoir capacity (c_t in [mm]), the non-conservative water exchange flux parameter (k_{exc} in [mm/dt]), and the linear routing parameter (l_{l_r} in [min]).

While learning both model parameters and initial states is possible, this study focuses solely on learning the conceptual hydrological parameters. To constrain and provide physical interpretability to the spatial distribution of these model parameters $\boldsymbol{\theta}(x)$, several physical descriptors, denoted as $\boldsymbol{D}(x)$, are introduced as inputs to the regionalization operator ϕ:

$$\boldsymbol{\theta}\left(x\right) = \phi(\boldsymbol{D}\left(x\right); \boldsymbol{\rho}),\ \forall x \in \Omega \tag{3.3}$$

with $\boldsymbol{D}$ the N_D-dimensional vector of physical descriptor maps covering Ω, and $\boldsymbol{\rho}$ the vector of tunable regionalization parameters that is defined below.

Two types of regionalization operators are used in HDA-PR (see Figure 3.1):

1. A set $\mathcal{P}$ of multivariate polynomial regression operators for each parameter of the forward hydrological model (Equation 3.2):

$$\begin{aligned} \boldsymbol{\theta}(x, \boldsymbol{D}, \boldsymbol{\rho}) &\coloneqq \mathcal{P}\left(\boldsymbol{D}(x), \boldsymbol{\rho}\right) \equiv \left[\left(\theta_k(x, \boldsymbol{D}, \rho_k)\right)_{k=1}^{N_\theta}\right]^T,\ \forall x \in \Omega; \\ \theta_k(x, \boldsymbol{D}, \rho_k) &\coloneqq s_k\left(\alpha_{k,0} + \sum_{d=1}^{N_D} \alpha_{k,d} D_d^{\beta_{k,d}}(x)\right),\ \forall k \in [1..N_\theta] \end{aligned} \tag{3.4}$$

with $s_k(z) = l_k + (u_k - l_k)/\left(1 + e^{-z}\right),\ \forall z \in \mathbb{R}$, a transformation based on a Sigmoid function with values in $]l_k, u_k[$, thus imposing bound constraints in the direct hydrological model such that $l_k < \theta_k(x) < u_k,\ \forall x \in \Omega$. The lower and upper bounds l_k and u_k, associated to each parameter field θ_k of the hydrological model (Equation 3.2) are assumed spatially uniform for simplicity here. The regional control vector to be estimated in this case is:

$$\boldsymbol{\rho} \equiv \left[\left(\rho_k\right)_{k=1}^{N_\theta}\right]^T \equiv \left[\left(\alpha_{k,0}, \left(\alpha_{k,d}, \beta_{k,d}\right)_{d=1}^{N_D}\right)_{k=1}^{N_\theta}\right]^T \tag{3.5}$$

2. An ANN denoted $\mathcal{N}n$, consisting of a multilayer perceptron, aimed at learning the descriptors-to-parameters mapping such that:

$$\boldsymbol{\theta}(x, \boldsymbol{D}, \boldsymbol{\rho}) \coloneqq \mathcal{N}n\left(\boldsymbol{D}(x), \boldsymbol{W}, \boldsymbol{b}\right), \forall x \in \Omega \tag{3.6}$$

where $\boldsymbol{W}$ and $\boldsymbol{b}$ are respectively weights and biases of the neural network composed of N_L dense layers. The architecture of the neural network and the forward propagation is detailed in Appendix C and Equation C.2. Note that an output layer consisting of a scaling transformation based on the Sigmoid function (cf. Equation C.1) enables the imposition of bound constraints on the k^{th}-hydrological parameters, i.e., $l_k < \theta_k(x) < u_k$, $\forall x \in \Omega$. The regional control vector in this case is:

$$\boldsymbol{\rho} \equiv [\boldsymbol{W}, \boldsymbol{b}]^T \equiv \left[(\boldsymbol{W}_j, \boldsymbol{b}_j)_{j=1}^{N_L}\right]^T \tag{3.7}$$

For each regionalization operator (Equation 3.4 or 3.6), the regional calibration problem consists in optimizing (in a sense defined below) the regionalization control $\boldsymbol{\rho}$ (Equation 3.5 or 3.7) that can be of relatively high dimension since it is proportional to the number of descriptors (N_D), the number of model parameters (N_θ), and the degree of spatialization of the regional controls. Optimization algorithms adapted to the high-dimensional problems of interest, taking advantage of accurate spatially distributed gradients computation with the adjoint of the forward model, are detailed thereafter. Importantly, note that by definition of the mathematical model (differentiable hydrological model, cf. Jay-Allemand, Javelle, et al. (2020), combined with embedded neural networks or polynomial regionalization functions which are differentiable) and given the numerical implementation rules followed, the forward numerical model is differentiable. This is a necessary condition for computing cost gradients with respect to spatially distributed hydrological parameters and obtain those of regional controls, as needed for solving the optimization problem. This is a key idea and property of our proposed algorithms.

The numerical resolution of the ordinary differential equation (ODE)-based operator of the forward hydrological model (Equation 3.2) relies on an explicit expression of its solution, approximated on the regular mesh $\mathcal{T}_\Omega$ of constant step dx with a fixed time step dt. All physical descriptors are mapped onto model grid for simplicity here.

Note that adding an upscaling operator after the regionalization scheme (as done in Samaniego, Kumar, and Attinger (2010)) is feasible in HDA-PR under the condition that it is differentiable (at least numerically), and is a potentially interesting topic for further research, as is improving observation operators. In both cases one could use algebraic expressions or neural networks in the HDA-PR assimilation framework.

3.2.2 Calibration cost function accounting for multiple observation sites

A calibration cost function is defined to measure the misfit between simulated and observed discharge time series, respectively denoted $\boldsymbol{Q}^* = (Q^*_{g=1..N_G})^T$ and $\boldsymbol{Q} = (Q_{g=1..N_G})^T$. In order to measure the discrepancy between observed and simulated quantities from multiple observation sites, we consider the cost function:

$$J(\boldsymbol{Q}^*, \boldsymbol{Q}) = \sum_{g=1}^{N_G} w_g j(Q^*_g, Q_g) \tag{3.8}$$

where $\sum_{g=1}^{N_G} w_g = 1$, and $j(Q^*_g, Q_g) = 1 - NSE(Q^*_g, Q_g)$ a local quadratic metric "at the station". Here we choose $1-NSE$, with NSE being the Nash-Sutcliffe efficiency, as the calibration metric (see Appendix A.1). This NSE-based calibration cost function is a quadratic differentiable and convex

function, involving the response of the direct model. It depends on the control vector $\boldsymbol{\rho}$ through the direct model $\mathcal{M}$ (Equation 3.1) composed of the regionalization operator ϕ (Equation 3.3) and the direct hydrological model $\mathcal{M}_{rr}$ (Equation 3.2). The multi-site calibration corresponds to $N_G > 1$ while $N_G = 1$ is the classical single-gauge calibration. For $N_G > 1$, the weighting w_g is defined such that $\sum_{g=1}^{N_G} w_g = 1$ (we consider $w_g = 1/N_G$ in this study).

3.2.3 The variationnal data assimilation algorithm

The VDA aims at estimating the tunable parameter $\boldsymbol{\rho}$ of the descriptors-to-parameters transfer function ϕ embedded into the hydrological model $\mathcal{M}_{rr}$ and predicting hydrological parameters $\boldsymbol{\theta}(x), \forall x \in \Omega$ from physical desciptors maps $\boldsymbol{D}(x)$, by minimizing the discrepancy between the modeled and observed discharges at multiple gauges. The cost function J to optimize depends on the regional control $\boldsymbol{\rho}$ through the response $\boldsymbol{U}(\boldsymbol{\rho}) = \mathcal{M}_{rr}(.;\, \boldsymbol{\theta} = \phi(\boldsymbol{\rho}))$ of the forward model that combines the hydrological model $\mathcal{M}_{rr}$ and the regionalization operator ϕ. The VDA inverse problem is written as the following convex optimization problem of the control vector $\boldsymbol{\rho}$:

$$\hat{\boldsymbol{\rho}} = \arg\min_{\boldsymbol{\rho}} J\left(\boldsymbol{Q}^*, \mathcal{M}_{rr}(., \phi(., \boldsymbol{\rho}))\right) = \arg\min_{\boldsymbol{\rho}} J\left(\boldsymbol{U}(\boldsymbol{\rho})\right) \tag{3.9}$$

The calibration of this regional control $\boldsymbol{\rho}$ aims to (i) reduce the misfit between observed and simulated discharges at spatially sparse gauging stations, as evaluated by Equation 3.8, while (ii) determining the hydrological parameter maps $\boldsymbol{\theta}(x)$ for discharge modeling at any ungauged sites, thereby benefiting from the information extracted from physical descriptors $\boldsymbol{D}(x)$, and spatial constraints induced by the regional transfer functions, whose parameters $\boldsymbol{\rho}$ are being optimized. The regionalization operator can be expressed as either (i) a multi-polynomial mapping $\phi \equiv \mathcal{P}$ (Equation 3.4), or (ii) an artificial neural network $\phi \equiv \mathcal{N}n$ (Equation 3.6). For both formulations of the regionalization operator, the regional control vector $\boldsymbol{\rho}$ to optimize is large, and gradient-based optimization methods adapted to high-dimensional inverse problems are employed and detailed thereafter.

Note that the inverse problem 3.9 is a VDA optimization problem in the sense that it seeks to optimally combines observations (here discharge series $Q^*_{g=1..N_G}$ at N_G multiple gauges) with a model (here of Equation 3.1). While the NSE-based cost function defined earlier is not built upon explicit probabilistic assumptions, it can be shown (Kavetski et al., 2006) that the minimization in Equation 3.9 is equivalent to maximizing the posterior density $p(\boldsymbol{\rho}|\boldsymbol{Q}^*)$ of the parameter $\boldsymbol{\rho}$ given observations $\boldsymbol{Q}^*$ under the assumptions of independent and identically distributed Gaussian errors at each gauge, spatially independent errors and no prior information. This probabilistic interpretation highlights a set of assumptions that could certainly be improved upon, and opens the way for more advanced probabilistic models that would explicitly recognize the various sources of uncertainty affecting the model and the surrounding data (forcings and responses). This is further discussed in section 3.4.

3.2.3.1 Optimization algorithm for polynomial regionalization

In this case, the forward model includes the polynomial descriptors-to-parameters mapping (Equation 3.4), i.e., $\phi \equiv \mathcal{P}$ and the regional control vector is:

$$\boldsymbol{\rho} \coloneqq \left[\alpha_{k,0}, (\alpha_{k,d}, \beta_{k,d})\right]^T, \forall (k,d) \in [1..N_\theta] \times [1..N_D]$$

The optimization problem, represented in Equation 3.9, is solved using the L-BFGS-B algorithm (Zhu et al., 1997), which is specially adapted to the high-dimensional parameter space. In this study, there are no bound constraints on the values of $\alpha_{k,.}$, whereas the exponents $\beta_{k,d}$ are simply sought between 0.5 and 2. This algorithm requires the gradient of the cost function with respect

to the sought parameters $\nabla_{\boldsymbol{\rho}} J$. This gradient is computed by a single run of the adjoint model, which is obtained by automatic differentiation using the Tapenade engine (Hascoet et al., 2013). The entire process is implemented in the `smash` Fortran source code, where the full forward model $\mathcal{M} \equiv \mathcal{M}_{rr}\left(., \mathcal{P}\left(.\right)\right)$ is a composition of both the hydrological model and the polynomial descriptors-to-parameters mapping.

The background value $\boldsymbol{\rho}^*$, used as a starting point for the optimization (see Chapter 4 for methods improving the background value), is set using a spatially uniform solution $\bar{\boldsymbol{\theta}}^*$, which is obtained by a simple global optimization algorithm (Michel, 1989) of the inverse problem (Equation 3.9) where $\mathcal{M} \equiv \mathcal{M}_{rr}$ and $\boldsymbol{\rho} \coloneqq \bar{\boldsymbol{\theta}}$, as follows:

$$\boldsymbol{\rho}^* \equiv \left[\alpha_{k,0} = s_k^{-1}\left(\bar{\theta_k}^*\right), \left(\alpha_{k,d} = 0, \beta_{k,d} = 1\right)\right]^T, \forall (k,d) \in [1..N_\theta] \times [1..N_D]$$

where $s_k^{-1}(z) = \ln\left(\frac{z-l_k}{u_k-z}\right)$ is the inverse Sigmoid.

The termination criterion is determined based on the satisfaction of at least one of the following criteria:

— Maximum number of iterations;
— Cost function criterion: $\frac{J^{(i)}-J^{(i+1)}}{\max\left(|J^{(i)}|,|J^{(i+1)}|,1\right)} \leq \epsilon \times 10^6$ (e.g., $\epsilon \approx 2.22 \times 10^{-16}$);
— Gradient criterion: $||\nabla_{\boldsymbol{\rho}} J^{(i)}||_\infty \leq 10^{-12}$

where $J^{(i)}$, $||\nabla_{\boldsymbol{\rho}} J^{(i)}||_\infty$ are respectively the cost value and its projected gradient at iteration i, and ϵ represents the machine precision.

3.2.3.2 Optimization algorithm for neural network-based regionalization

In this case, the forward model includes a descriptors-to-parameters mapping performed with a neural network, i.e., $\phi \equiv \mathcal{N}n$ and the regional control vector is $\boldsymbol{\rho} \coloneqq [\boldsymbol{W}, \boldsymbol{b}]^T$. The optimization problem (Equation 3.9) can typically be solved using the Adam optimization algorithm (Kingma et al., 2014), an efficient stochastic gradient descent algorithm able to adapt the learning rate based upon the first and the approximation of the second moments of the gradients for fast convergence, and only requiring the first order gradients of the cost function. In the present case, the cost function writes as:

$$J\left(\boldsymbol{U}\left(\boldsymbol{\rho}\right)\right) = J\big(\boldsymbol{Q}^*, \mathcal{M}_{rr}(.\ ,\ \boldsymbol{\theta} = \mathcal{N}n(\boldsymbol{D}, \boldsymbol{\rho}))\big) \tag{3.10}$$

This formulation of the cost function highlights its dependency on the forward model $\mathcal{M} \equiv \mathcal{M}_{rr}\left(., \mathcal{N}n\left(.\right)\right)$, which is composed of two components in its numerical implementation: (i) an ANN implemented in Python, which produces the output $\boldsymbol{\theta}$ used as input by (ii) the hydrological model $\mathcal{M}_{rr}$ implemented in Fortran. In order to optimize J, its gradients with respect to $\boldsymbol{\rho}$ are required. The main technical difficulty here is to achieve a "seamless flow of gradients" through backpropagation. To overcome this, we divide the gradients into two parts and apply the chain rule with analytical derivation and numerical code differentiation (cf. hybrid VDA course in Monnier (2021) and references therein). First, $\nabla_{\boldsymbol{\theta}} J$ can be computed via the automatic differentiation applied to the Fortran code corresponding to $\mathcal{M}_{rr}$. Then, $\nabla_{\boldsymbol{\rho}} \boldsymbol{\theta}$ is simply obtained by analytical calculus applicable given the explicit architecture of the ANN, consisting of a multilayer perceptron. Finally, the two gradients can be combined as $\nabla_{\boldsymbol{\rho}} J = \nabla_{\boldsymbol{\theta}} J . \nabla_{\boldsymbol{\rho}} \boldsymbol{\theta}$. The background value $\boldsymbol{\rho}^*$ in this case is randomly initialized using a specific method, which will be discussed later. The termination criterion is determined by a specified number of training epochs in the optimization algorithm. A detailed explanation of the network architecture, backward propagation, and the optimization process can be found in Appendix C.

3.3 Data and numerical experiment

3.3.1 Study area and experimental design

The performance of HDA-PR in terms of discharge modeling but also in terms of its capability to extract information from data to estimate conceptual model parameters is analyzed over the French Mediterranean region. The performance of three regionalization methods is compared: a simple multi-gauge calibration approach for spatially uniform model parameters (Uniform), and two variants of HDA-PR. This first variant (Multi-linear) uses linear multivariate regression on physical descriptors. The second variant (ANN) uses a multilayer perceptron, which corresponds to a more complex mapping.

The evaluation is performed on a flash-flood-prone area known as the Mediterranean arc (and called ArcMed hereafter), situated in the South of France, as indicated in Figure 3.2. Covering

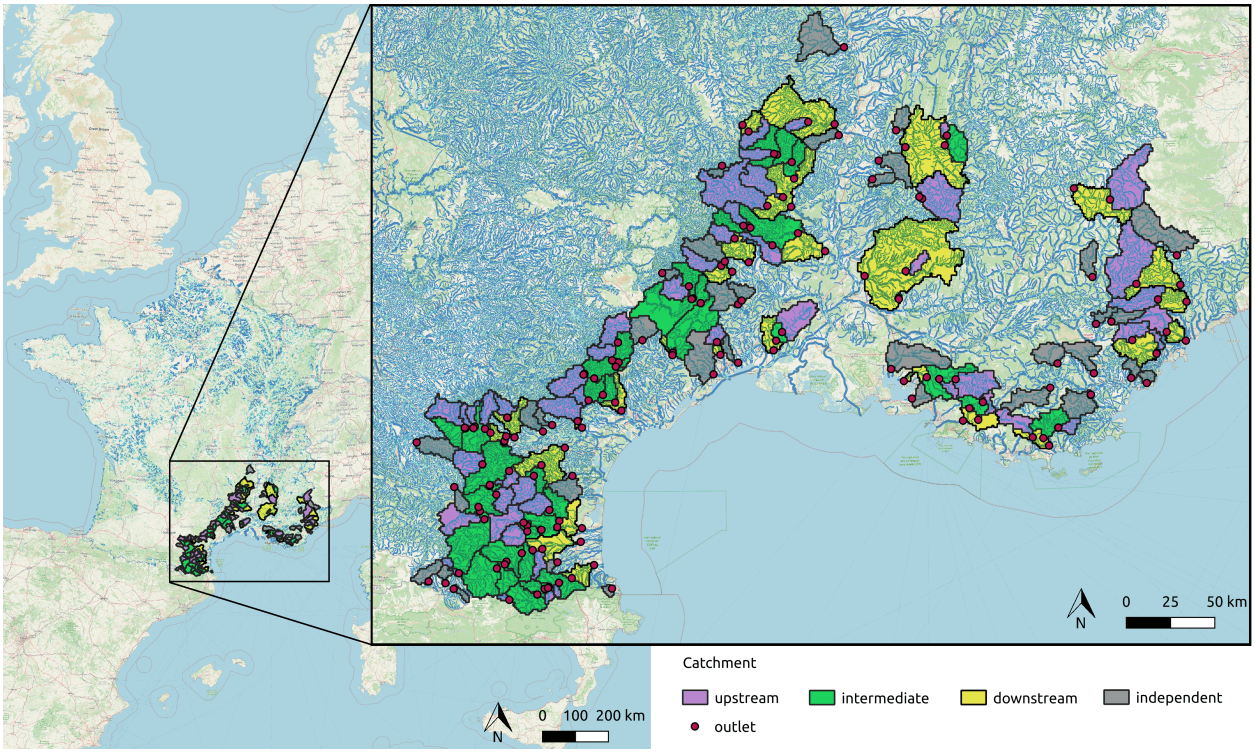


Figure 3.2 – Map of France highlighting the ArcMed study area, covering 150,000 km^2 (100,000 km^2 excluding sea), comprising 126 catchments categorized as 38 catchments located upstream, 33 intermediate catchments, 24 catchments positioned downstream, and 31 independent catchments, representing a total drainage area of 26,000 km^2.

an approximate land area of 100,000 km^2, ArcMed comprises 126 flash-flood-prone catchments, including both nested and independent ones, representing about 26,000 km^2 of combined drainage area. These catchments have been selected based on the availability of long time series with high-quality observed flow data and minimal anthropogenic impacts. ArcMed is known for its diverse hydrological properties and contrasted catchment behaviors, including steep topography and very heterogeneous soils and bedrock (e.g., Garambois, Roux, Larnier, Labat, et al. (2015b)). The area is affected by intense rainfall events that trigger non-linear flash flood responses, presenting a very challenging modeling scenario, in particular due to the significant presence of karstic zones. This evaluation dataset is quite extensive considering the high-resolution, which is made necessary

by the fast and small-scale non-linear processes characterizing the flash floods occurring in the area. The resulting spatio-temporal computational domain is quite large ($> 26,000$ pixels $\times\, 35,000$ time steps). By comparison, regionalization studies in the literature often make use of larger regions and more numerous catchments, but also of much coarser models typically running at a daily time step and coarser spatial resolution.

A set of 7 physical descriptors (see Table 1.1 in Chapter 1) available throughout Metropolitan France is used following Odry (2017) and Jay-Allemand, Demargne, et al. (2024). Note that this setup is sufficient to assess the regionalization performance of the proposed algorithms while keeping the present study concise. The issue of selecting the most relevant information for multi-source observations of hydrological responses and the most adequate descriptors layers is intentionally left for future research since it requires additional complementary modules to the proposed framework. It is worth noting that prior to the optimization process, all descriptors are standardized between 0 and 1 through min-max scaling.

To evaluate the performance of HDA-PR, we employ two spatial cross-validation setups which involve (i) calibrating the model on 38 upstream gauges (upstream calibration) and (ii) calibrating the model on 24 downstream gauges (downstream calibration). The upstream calibration poses a greater challenge due to the smaller catchments' areas, resulting in a lower integrative effect on non-linear hydrological processes and potentially more hydrological variabilities within and between basins. Consequently, this setup represents a more demanding interpolation scenario compared to the downstream calibration, which involves larger basins, as demonstrated later with calibration results. These two calibration setups, along with a temporal validation scheme based on a two-period split, will facilitate the study of information extraction from discharge time series and from physical descriptors. For each calibration setup, we apply the different multi-site regional calibration methods using the set of gauges from the calibration catchments, while the remaining gauges from the validation catchments are used for spatial and spatio-temporal validation purposes. The weighting in Equation 3.8 can be set as $w_g = \frac{1}{N_G}$, representing the average cost over multiple gauges, since the gauges used for calibration in each setup, have the same number of observations (up to sporadic missing values), and share the same nature (either downstream or upstream), thus have similar information accumulation along the flow paths. The chosen calibration metric is the NSE, computed using data from multiple gauges over a four-year period P1 (August 2016 to July 2020) with a one-year warm-up period (August 2016 to July 2017). The following calibration methods are compared:

- Local calibrations for each gauge, both with spatially uniform (i.e., $\boldsymbol{\rho} \equiv \bar{\boldsymbol{\theta}}$) and full spatially distributed controls (i.e., $\boldsymbol{\rho} \equiv \boldsymbol{\theta}(x)$), which are respectively under- and over-parameterized hydrological optimization problems. These approaches represent reference or benchmark performances, denoted as "Uniform (loc)" and "Distributed (loc)".
- Multi-gauge regional calibration approaches with:
  - lumped model parameters (i.e., $\boldsymbol{\rho} \equiv \bar{\boldsymbol{\theta}}$), representing "level 0" regionalization, denoted as "Uniform (reg)";
  - a multivariate linear mapping (i.e., $\boldsymbol{\rho} \equiv [\alpha_{k,0}, (\alpha_{k,d}, 1)]^T$), referred to as "Multi-linear (reg)", which represents the classical regionalization mapping with the same complexity as the transfer functions of H. E. Beck, M. Pan, et al. (2020) and without upscaling laws;
  - a multilayer perceptron (i.e., $\boldsymbol{\rho} \equiv [\boldsymbol{W}, \boldsymbol{b}]^T$), denoted as "ANN (reg)", representing the core novelty of the HDA-PR framework.

Note that the multivariate linear mapping above is a particular case of the polynomial mapping presented in section 3.2.1, obtained by forcing exponents $\beta_{k,d}$ to one. General polynomial mappings have been studied but are not further considered in this study for the sake of conciseness. To ensure robust validation, model performances are assessed in terms of spatial, temporal, and spatio-temporal validations, with a temporal validation period P2 covering two years from August 2020 to July 2022. Various evaluation metrics, including multiple hydrological signature-based metrics

presented in Chapter 2, are also employed.

3.3.2 Regional learning performance

In this section, the regional learning performance is analyzed over the whole ArcMed region. Before a thorough evaluation of performance is performed, Figure 3.3 provides several examples of observed and simulated discharges at randomly selected gauges, and compares them to observations. For these examples, using lumped model parameters $\bar{\boldsymbol{\theta}}$ leads to a poor performance in simulating discharges. By contrast, the other two regional learning methods result in improved performance, in both calibration and validation catchments. Given the complexity and heterogeneity of the region, it is unsurprising that lumped model parameters regionalization is unable to accurately reproduce such contrasted hydrological responses.

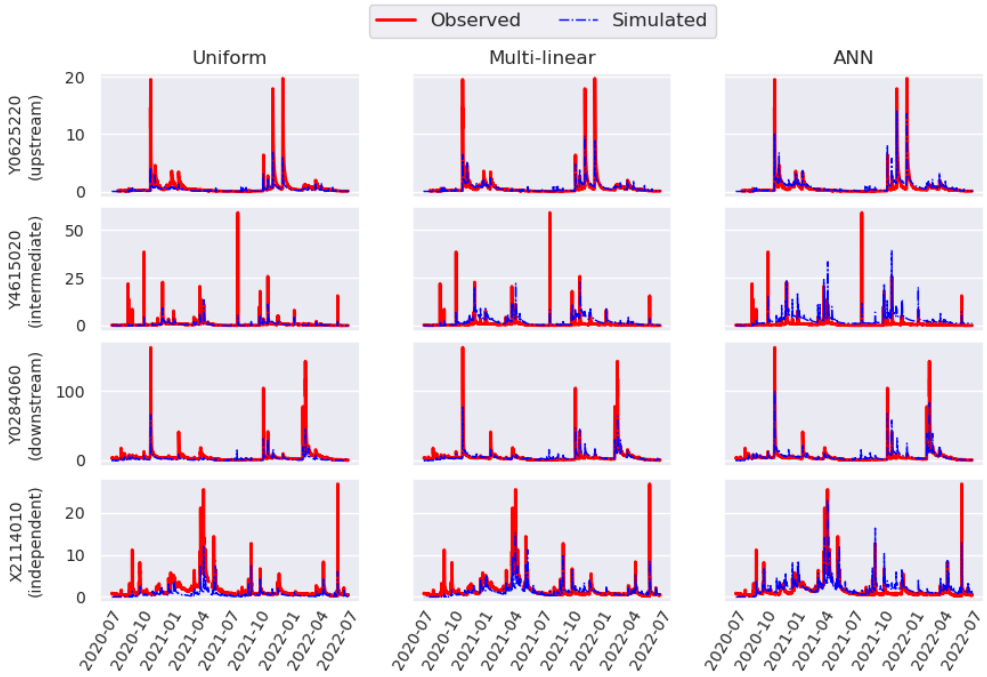


Figure 3.3 – Observed and simulated discharges (in m^3/s) at several locations during the validation period P2, using three multi-gauge regional calibration methods (columns) within the first calibration setup, where upstream catchments are used for calibration on P1. The first row hence corresponds to temporal validation assessments, while the subsequent three rows correspond to spatio-temporal validation assessments.

Figure 3.4 focuses on global NSE scores (see Figure 3.5 for KGE scores), i.e., the calibration metric, over the whole ArcMed region, for the two local calibration methods (spatially uniform and spatially distributed calibrations) and the three regionalization methods (Uniform, Multi-linear, and ANN). Overall, it suggests that the regionalization methods incorporating information from physical descriptors and imposing model parameters spatialization (Multi-linear and ANN) lead to superior performance when compared to the spatially Uniform baseline. This trend is particularly noticeable in the more challenging calibration-validation scenario using only upstream gauges for calibration. Furthermore, the enhanced efficiency of ANN-based regionalization is evident from global performance comparisons against the Multi-Linear approach, considering both upstream and downstream basin sets used in regional optimization.

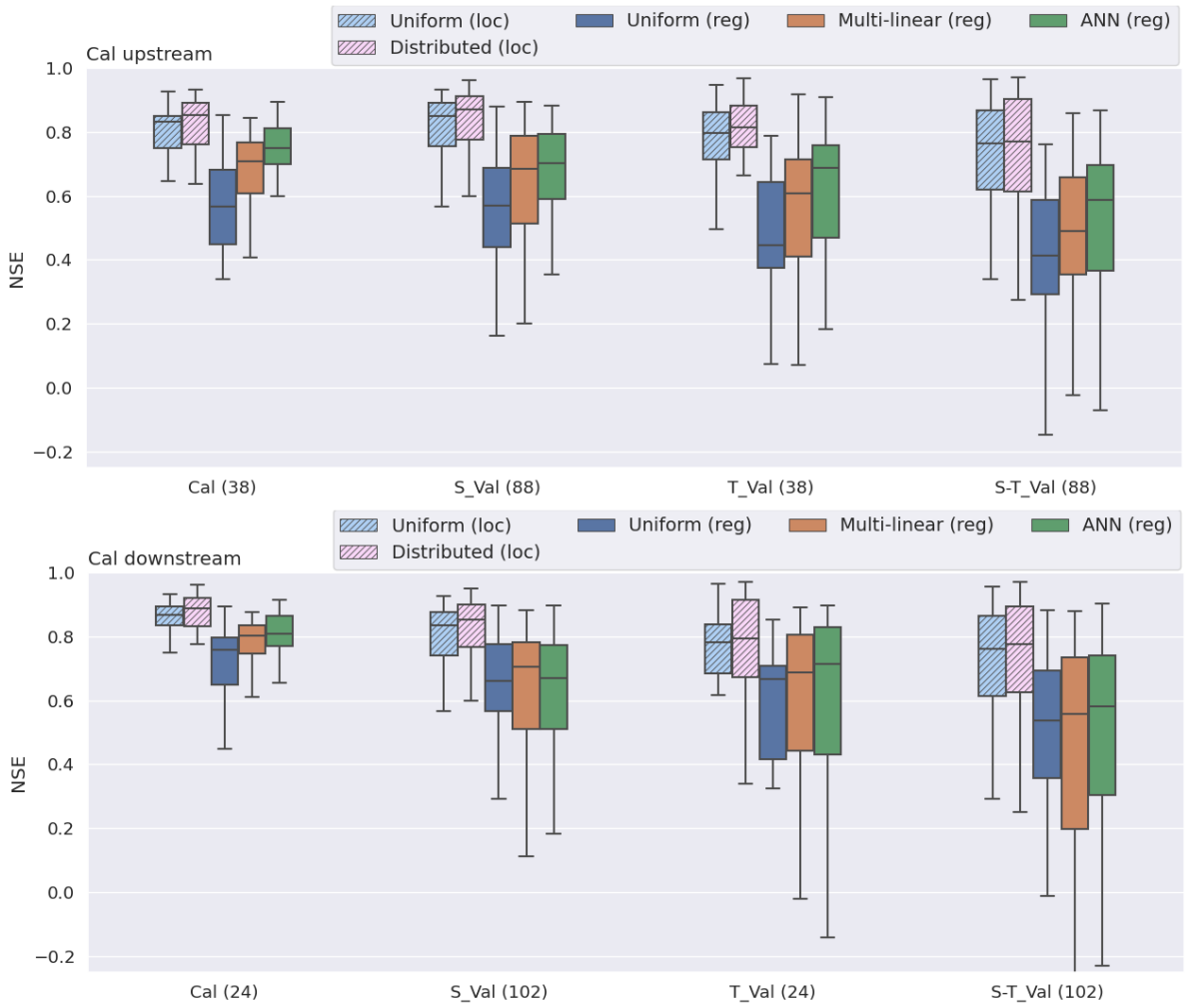


Figure 3.4 – Boxplots of NSE scores (optimal value = 1) across calibration and validation catchments for both calibration setups which are upstream (top) and downstream (bottom), compared to reference solutions obtained by local calibration methods (Uniform (loc) and Distributed (loc)). From left to right: results are displayed for calibration catchments on period P1 ("Cal"), validation catchments on P1 for spatial validation ("S_Val"), calibration catchments on P2 for temporal validation ("T_Val"), and validation catchments on P2 for spatio-temporal validation ("S-T_Val"). The numbers in parentheses indicate the count of catchments included in each boxplot.

In calibration ("Cal"), median NSE scores for the upstream calibration on 38 catchments are 0.7 and 0.75 for Multi-linear and ANN regionalization methods, representing a marked improvement compared to 0.57 for the Uniform approach. By contrast, for the downstream calibration, the median NSE scores are much closer: 0.8 and 0.81 for Multi-linear and ANN regionalization methods versus 0.76 for the Uniform approach. This indicates that the improvement obtained with the former two regionalization methods is smaller for the downstream calibration than for the upstream one. Moreover, the NSE performance for the Uniform approach is already quite close to that of the reference local calibrations in the downstream case, leaving little room for improvement with Multi-linear and ANN regionalization methods. Overall, these results suggest that the multi-gauge calibration problem poses a more challenging interpolation issue on upstream gauges than on downstream gauges. Thus, the regional calibration problem incorporating descriptors is more

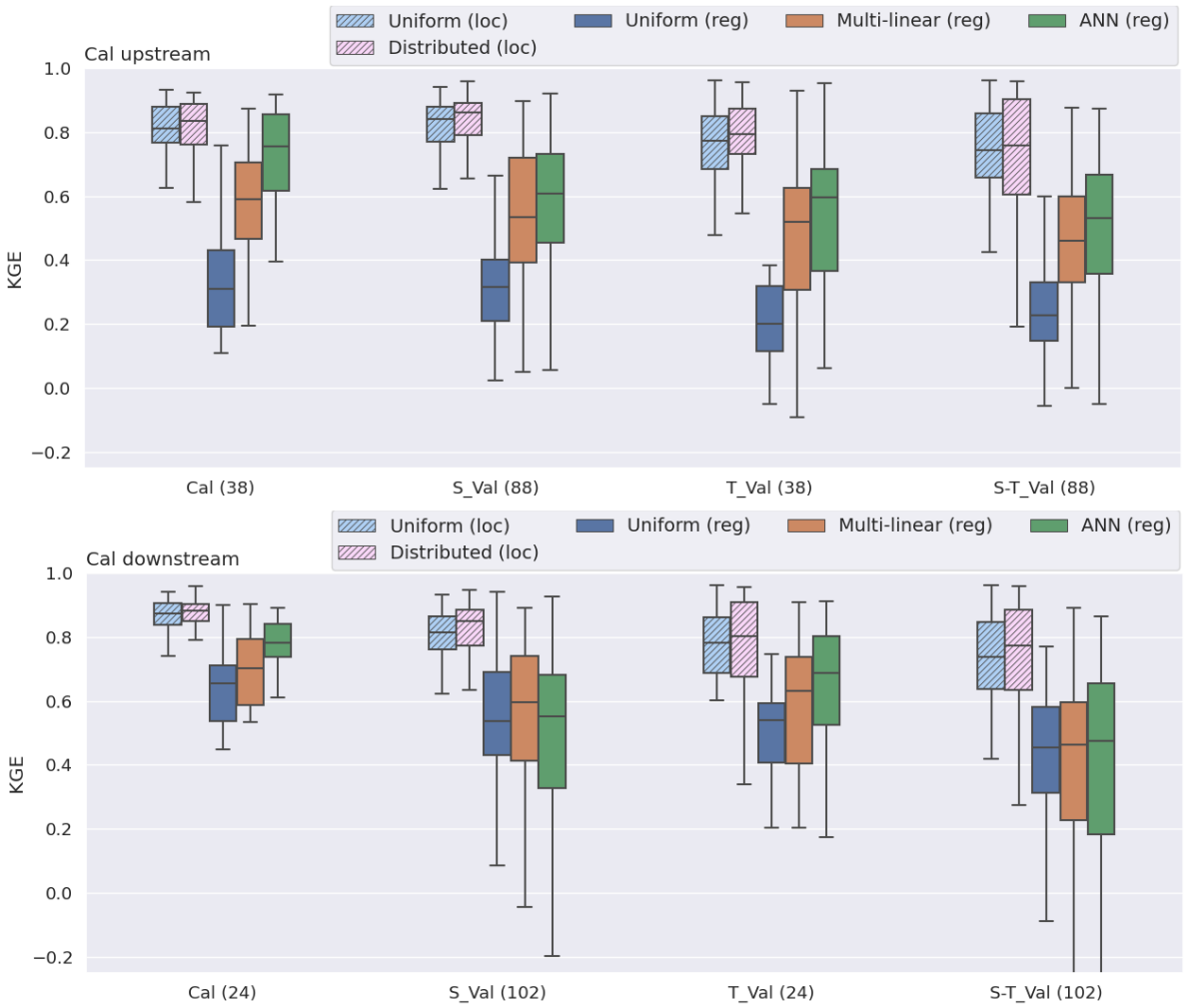


Figure 3.5 – Boxplots of KGE scores (optimal value = 1) across calibrated (with NSE cost function) and validation catchments for both calibration setups (upstream (top) and downstream (bottom)), compared to reference solutions obtained by local calibration methods (Uniform (loc) and Distributed (loc)). From left to right: results are displayed for calibration catchments on period P1 ("Cal"), validation catchments on P1 for spatial validation ("S_Val"), calibration catchments on P2 for temporal validation ("T_Val"), and validation catchments on P2 for spatio-temporal validation ("S-T_Val"). The numbers in parentheses indicate the count of catchments included in each boxplot.

demanding on smaller-sized catchments with lower integrative effects and potentially higher non-linear responses across basins. In terms of temporal validation ("T_Val") for both upstream and downstream cases, the same ranking as in calibration is observed with the three regionalization methods. Specifically, the ANN achieves a median NSE score of 0.69 (resp. 0.71) compared to 0.6 and 0.44 (resp. 0.69 and 0.68) for Multi-linear and Uniform in the upstream calibration setup (resp. downstream calibration setup). In spatial validation ("S_Val"), which assesses the potential of regionalization methods for ungauged basins, the ANN demonstrates the best performance in the upstream calibration setup, while similar performances (similar medians and interquartile ranges) of the three regionalization methods are observed in the downstream setup. Moving on to the most complex evaluation for the extrapolation capability of HDA-PR, spatio-temporal validation ("S-T_Val"), results consistently show the efficiency of the ANN, improving the NSE scores by 0.11 (resp. 0.18) in upstream setup and by 0.02 (resp. 0.04) in downstream setup compared to Multi-linear

(resp. Uniform). In summary, multi-linear regression leads to slightly better median performances in spatial validation, while ANN performs best in temporal and spatio-temporal validations in the downstream calibration setup. However, in the upstream calibration case, the ANN demonstrates superior performance in calibration and for all validation scenarios. This result highlights the capability of ANN to outperform (at least in terms of scoring metrics up to this point) classical regionalization approaches using lumped model parameters and multi-linear regression in the most challenging extrapolation scenario (i.e., upstream calibration).

We now delve into a more detailed analysis of NSE performances in validation catchments categorized by their nature (upstream, downstream, intermediate, or independent) for both calibration setups: upstream and downstream, as illustrated in Figure 3.6 (see Figure 3.7 for KGE scores).

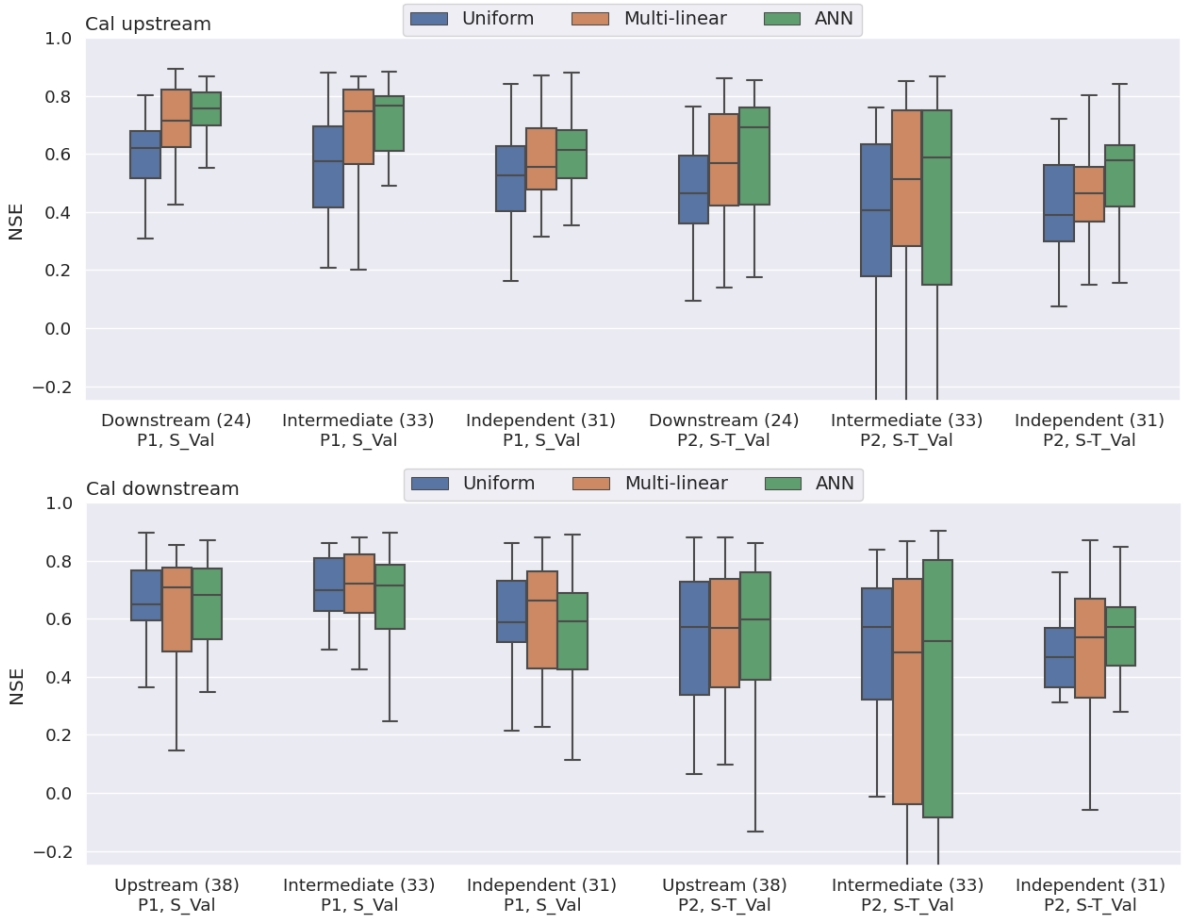


Figure 3.6 – Comparison of NSE scores (optimal value = 1) across validation catchments categorized by their nature (upstream, downstream, intermediate, or independent) for both calibration setups (upstream (top) and downstream (bottom)). From left to right: results are displayed for validation catchments on period P1 for spatial validation ("S_Val") and validation catchments on P2 for spatio-temporal validation ("S-T_Val"). The numbers in parentheses indicate the count of catchments included in each boxplot.

In the downstream calibration setup, no clear ranking appears. Regarding the upstream calibration setup, the median NSE is systematically higher with ANN regionalization, with less spread (interquartile range) in all basin classes except for the 33 intermediate catchments, which show a slightly smaller first quartile compared to Multi-linear. Moreover, significantly higher performances are achieved with ANN in spatio-temporal validation. This is particularly notable on a relatively

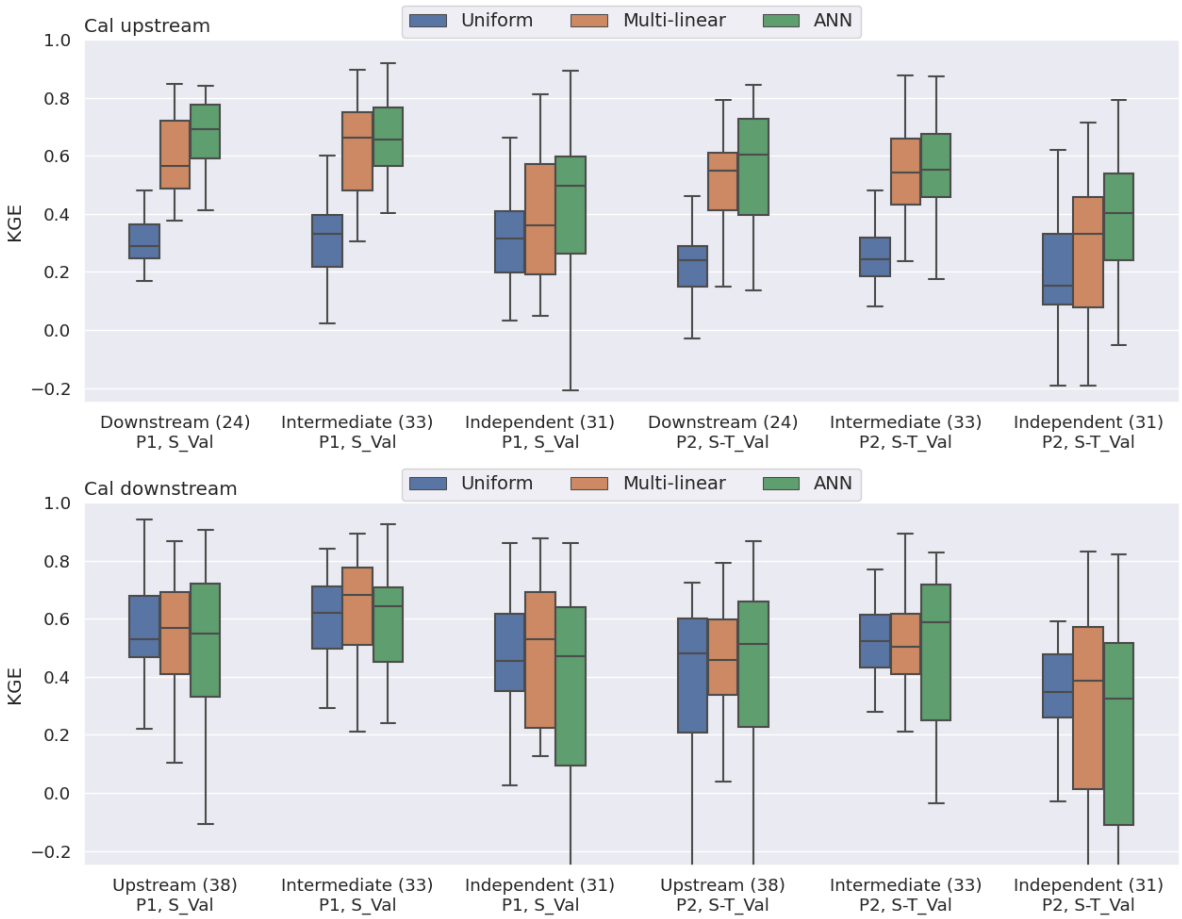


Figure 3.7 – Comparison of KGE scores (optimal value = 1) across validation catchments categorized by their nature (upstream, downstream, intermediate, or independent) for both calibration setups (upstream (top) and downstream (bottom)). From left to right: results are displayed for validation catchments on period P1 for spatial validation ("S_Val") and validation catchments on P2 for spatio-temporal validation ("S-T_Val"). The numbers in parentheses indicate the count of catchments included in each boxplot.

limited number of smaller-sized catchments in the calibration, indicating an effective extraction of information from both discharge data and descriptors. This effectiveness becomes evident when the model is tested on the validation catchments, which are more numerous, may or may not be nested and have varied areas. Overall, the performances obtained with both upstream and downstream descriptor-based calibrations show the relevance of regionalization with ANN. Indeed, ANN performs best in the upstream setup. ANN performance is not always the highest in the downstream setup, but it remains very similar to other regionalization approaches. The complex and highly-parameterized nature of the ANN hence does not seem to result in a deterioration of performances in validation, even when this complexity may not be necessary.

To quantify how much the ANN-based regionalization approach improves the model performance compared to multi-linear regression and spatially uniform parameters regionalization, we compute the improvement rate in terms of NSE of the ANN relative to the other methods. We focus on the upstream calibration setup and evaluate improvement rates of ANN versus the two simpler approaches (Uniform and Multi-linear) in spatial and spatio-temporal validation, as illustrated in the maps of Figure 3.8. Comparing to Uniform, the ANN yields positive improvement rates at more

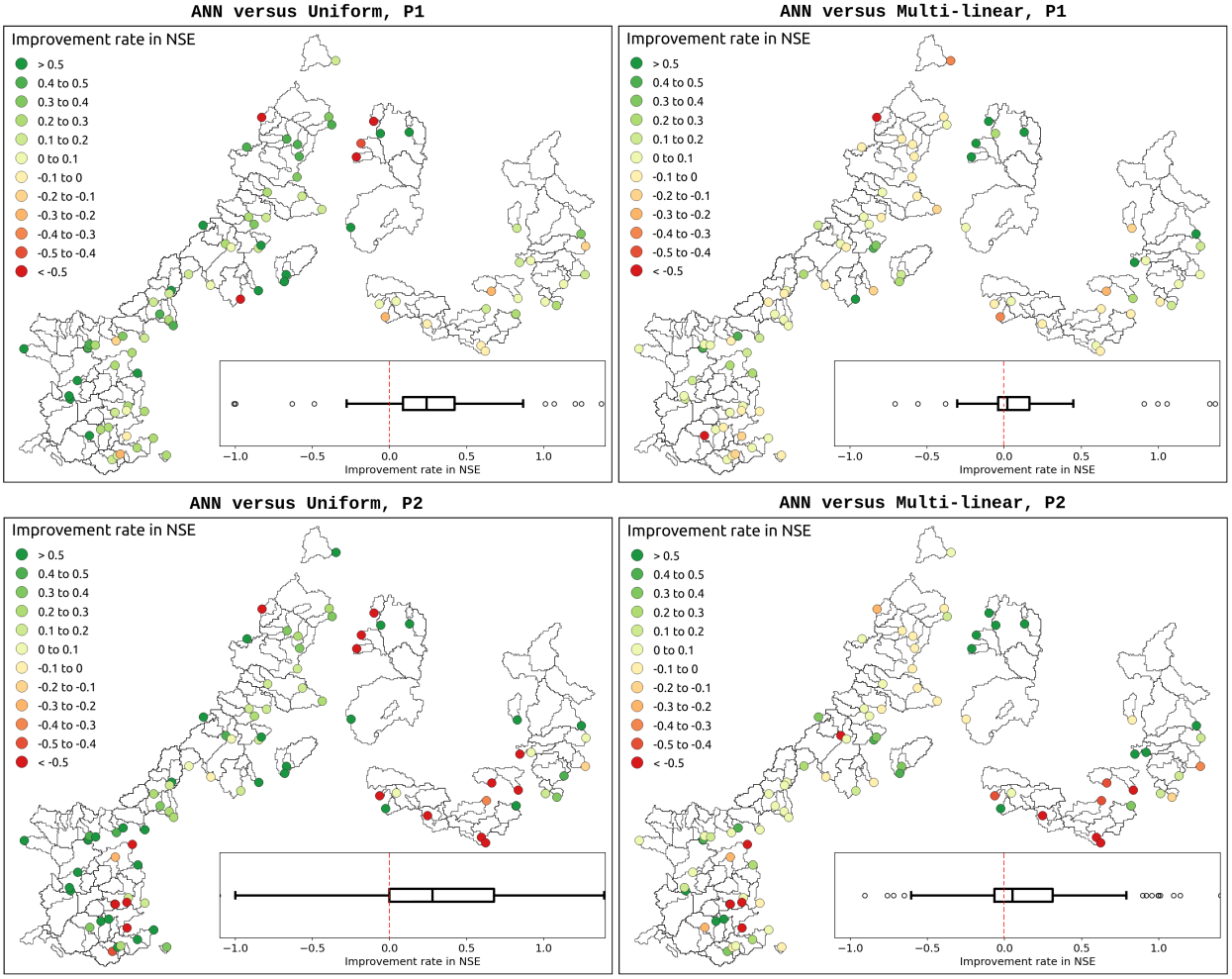


Figure 3.8 – Comparison of NSE improvement rates in spatial and spatio-temporal validation for HDA-PR with ANN mapping versus uniform (left) and multi-linear (right) mappings, utilizing models calibrated with upstream catchments. Results are shown across validation catchments positioned downstream, intermediate, and independent for spatial validation on period P1 (top) and spatio-temporal validation on period P2 (bottom). The improvement rate at a catchment i is calculated as $r^i = \frac{NSE^i_{\text{ANN}} - NSE^i_{\square}}{|NSE^i_{\square}|}$, where $\square$ represents either multi-linear or uniform mapping.

than 80% of validation catchments on P1, with a median improvement rate of around 0.25 (see the boxplot in the top left panel of Figure 3.8). Transitioning to P2, the variation of improvement rates among catchments becomes more significant. Catchments that exhibited positive improvement rates during P1 generally maintain positive or enhanced rates, whereas some of those with negative rates either worsen or remain unchanged. Although the median improvement rate of NSE reaches 0.3, the interquartile range and whiskers visibly expand, as indicated by the boxplot in the bottom left panel of Figure 3.8. Regarding the improvement rates of ANN versus Multi-linear, we observe similar trends in the variation of improvement rates among catchments across the two periods. While the number of catchments with high improvement rates is generally reduced, more than 65% of catchments still exhibit positive rates on both periods. Nevertheless, it is apparent that some catchments (situated in the southeast region of the map and known for their complex hydrological behaviours including karstic effects and more difficulties in hydrological modeling) display negative

improvement rates of the ANN versus both Uniform and Multi-linear approaches. This discrepancy may stem from difficulties in extrapolating effective model parameters from the considered physical descriptors at these locations with the ANN. This could be improved by using other descriptors and/or stronger constraints, for example, on the sensitive exchange parameter k_{exc}.

In order to obtain a more robust evaluation criterion adapted to flood modeling, we consider validation in terms of multiple evaluation metrics based on hydrological signatures for flood events, which are computed via an automated segmentation algorithm proposed in Chapter 2. This evaluation is based on the relative error of three flood event signatures: flood runoff coefficient (Erc), flood flow (Eff), and peak flow (Epf), computed on a total of 1,737 flood events selected from the entire study area over a 6-year period from August 2016 to July 2022. In the upstream calibration setup, the two non-uniform regionalization methods, and in particular the ANN, demonstrate their ability to outperform the uniform regionalization (using lumped model parameters) in both calibration and validation, as illustrated in Figure 3.9. For example, the relative errors (median over flood events)

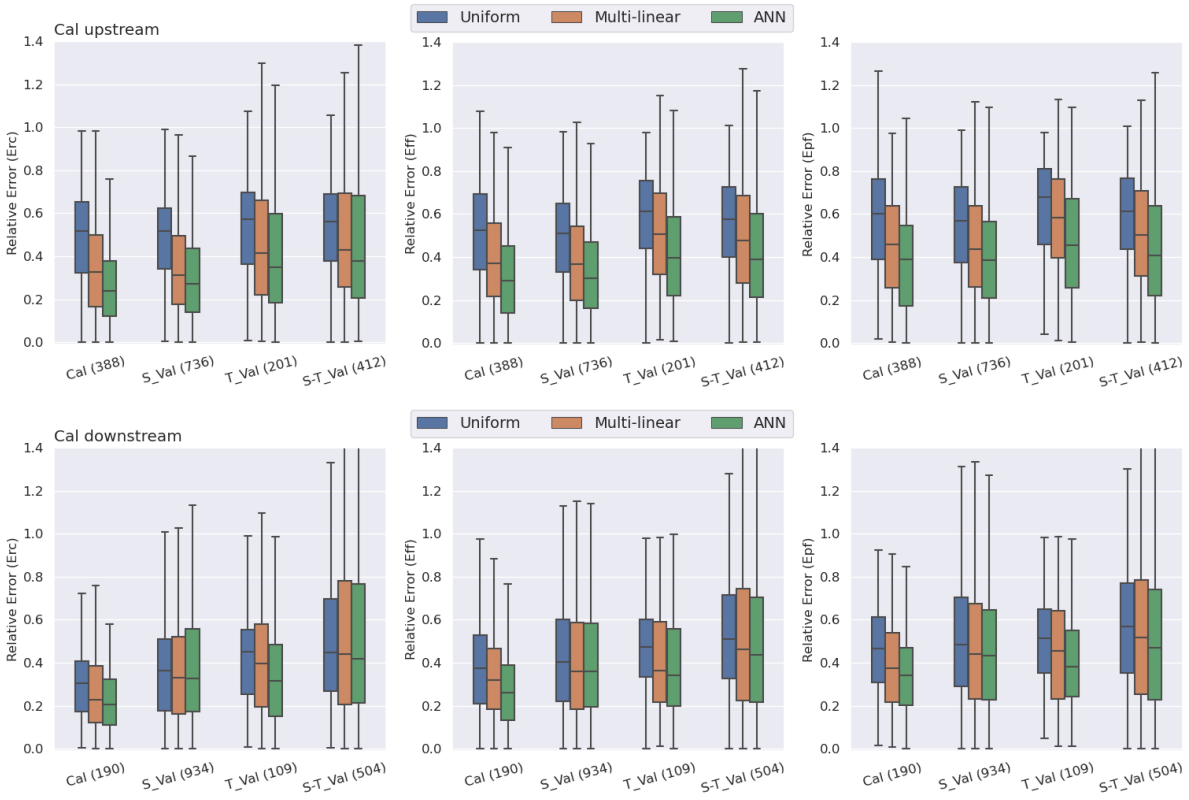


Figure 3.9 – Relative error (optimal value = 0) of three flood event signatures—flood runoff coefficient (Erc), flood flow (Eff), and peak flow (Epf)—for both calibration setups: upstream (top) and downstream (bottom). The evaluation is based on a total of 1,737 flood events, with 1,124 selected during the calibration period P1 and 613 selected during the validation period P2, across 126 catchments. From left to right: results are displayed for calibration catchments on P1 ("Cal"), validation catchments on P1 for spatial validation ("S_Val"), calibration catchments on P2 for temporal validation ("T_Val"), and validation catchments on P2 for spatio-temporal validation ("S-T_Val"). The numbers in parentheses indicate the count of flood events included in each boxplot.

of simulated peak flows using ANN are around 0.4, compared to over 0.6 (for the uniform baseline) and 0.5 (for Multi-linear) in spatio-temporal validation (upstream calibration setup). In the downstream calibration setup, although the ANN does not markedly outperform the Multi-linear

and Uniform regionalizations, it still achieves the best scores in terms of flood event signatures in both calibration and validation. For example, in spatio-temporal validation of the peak flow, ANN achieves a median relative error of 0.47, while Uniform and Multi-linear have median relative errors of 0.58 and 0.52. It is noteworthy that these flood signatures-based metrics were not included in the cost function during the calibration process, which further supports the robustness and power of the regionalization methods, particularly the one based on ANN. It thus underscores the potential of HDA-PR for enhancing flash flood forecasting systems.

3.3.3 Interpretation of the learning process and inferred parameters

A key feature of the proposed HDA-PR algorithm pertains to the differentiability of the forward spatially distributed model, including the ANN or Multi-Linear regionalization mappings (cf. section 3.2.1), which allows obtaining the gradient $\nabla_{\boldsymbol{\theta}} J$ of the cost function with respect to the hydrological model parameters. This gradient facilitates the computation of $\nabla_{\boldsymbol{\rho}} J$ necessary for optimizing the regional control vector $\boldsymbol{\rho}$ using algorithms tailored for high-dimensional inverse problems (cf. section 3.2.3), while maintaining a "physical" interpretation. Spatial maps of the cost gradient with respect to hydrological model parameters (which also represents the local sensitivity of the model parameters), at a background value $\boldsymbol{\theta}^* = \phi(., \boldsymbol{\rho}^*)$ corresponding to the control value $\boldsymbol{\rho}^*$ at the first iteration of the optimization process (cf. section 3.2.3.1 and section 3.2.3.2), are shown in Figure 3.10: $\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}^*)(x) = \left(\frac{\partial J(\boldsymbol{\theta}^*)}{\partial c_p}, \frac{\partial J(\boldsymbol{\theta}^*)}{\partial c_t}, \frac{\partial J(\boldsymbol{\theta}^*)}{\partial k_{exc}}, \frac{\partial J(\boldsymbol{\theta}^*)}{\partial l_{l_r}} \right)(x)$). The figure displays the

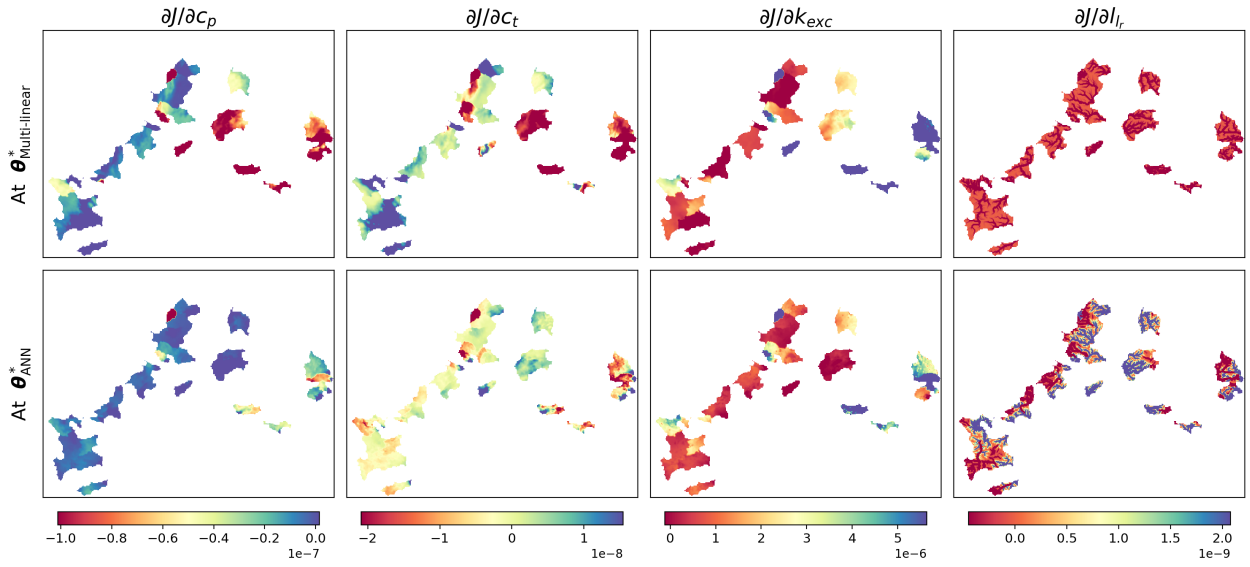


Figure 3.10 – Spatially distributed cost gradients $\nabla_{\boldsymbol{\theta}} J\left(\boldsymbol{\theta}^*_{\square}\right)$ to conceptual hydrological parameters used in the optimization process (of $\boldsymbol{\rho}$ using $\nabla_{\boldsymbol{\rho}} J = \nabla_{\boldsymbol{\theta}} J . \nabla_{\boldsymbol{\rho}} \boldsymbol{\theta}$) showed for downstream calibration (for the sake of visibility) at initial points $\boldsymbol{\theta}^*_{\text{Multi-linear}} := \mathcal{P}(., \boldsymbol{\rho}^*_{\text{Multi-linear}})$ and $\boldsymbol{\theta}^*_{\text{ANN}} := \mathcal{N}n(., \boldsymbol{\rho}^*_{\text{ANN}})$, where $\boldsymbol{\rho}^*_{\text{Multi-linear}}$ (resp. $\boldsymbol{\rho}^*_{\text{ANN}}$) is the background value of the multi-linear regression (resp. ANN) mapping. Note that the cost gradient values are obtained for the active cells within the gauged partition, i.e., cells contributing to the simulated discharge at gauges (downstream here) where the observation cost function J is evaluated.

maps for the downstream calibration setup, chosen for enhanced visualization compared to smaller basin sizes utilized in the upstream calibration setup. For both regionalization mappings (ANN or Multi-Linear), the initial cost gradient (sensitivity) at different points $\boldsymbol{\theta}^*$ in model parameter space for each mapping, indicates a high sensitivity to the non-conservative exchange parameter

k_{exc}, which is coherent with global and local sensitivity analysis from Chapter 2 on a comparable model structure without regionalization mapping. Interestingly, the gradient maps (for the first optimization iteration) of the forward model containing descriptors-to-parameters mappings show the spatial trends that will be obtained in the following optimization iteration, since a positive gradient should result in a parameter increase and conversely (if $\nabla_{\boldsymbol{\rho}}\boldsymbol{\theta}$ is of the same sign as $\nabla_{\boldsymbol{\theta}}J$ plotted here since $\nabla_{\boldsymbol{\rho}}J = \nabla_{\boldsymbol{\theta}}J.\nabla_{\boldsymbol{\rho}}\boldsymbol{\theta}$). Note that due to the high dimension of $\boldsymbol{\rho}$ and its non-physical meaning for ANN weights and biases, $\nabla_{\boldsymbol{\rho}}\boldsymbol{\theta}$ cannot be plotted in a simple manner and, more importantly, cannot be "physically" interpreted like $\nabla_{\boldsymbol{\theta}}J$. Moreover, these maps of $\nabla_{\boldsymbol{\theta}}J$ interestingly contain the footprint of physical descriptor patterns (along with the footprint of model structure and multi-site discharge observations and cost function formulation); they indicate sensitivity "hot spots", which could be useful in future works for tailoring descriptor processing and choice, as well as for studying other information sources, tailoring cost functions, parameter bounds, and regularizations of the forward-inverse problem.

When calibrating a model with gradient-based optimization algorithms, it is also important to discuss the descent of the cost function. This analysis enables understanding how optimization algorithms converge towards the global or local minimum of the cost function, and identifying potential trade-offs between model flexibility and overparameterization, thereby completing validation results. The descent of the cost function J is represented in Figure 3.11. It is apparent that the cost func-

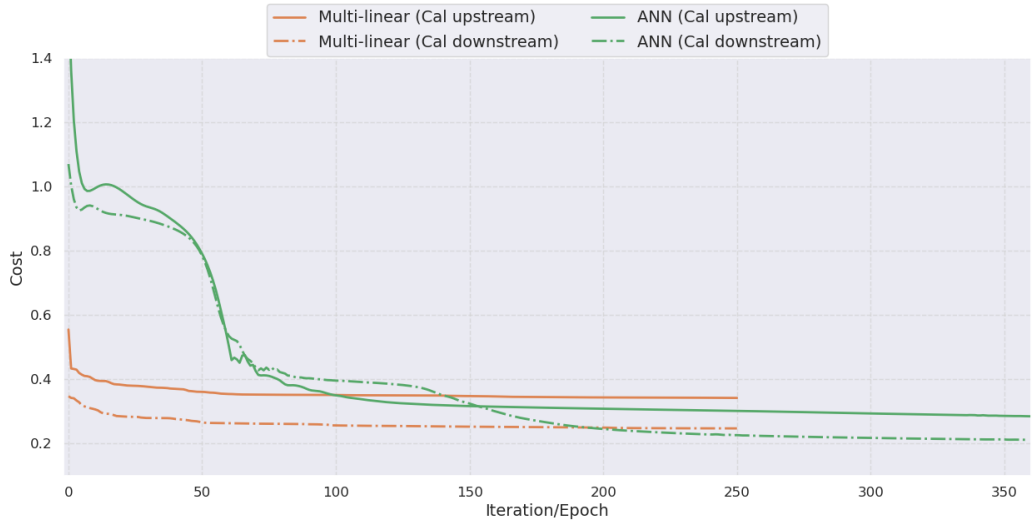


Figure 3.11 – The descent of the cost function $J = 1 - NSE$ for both Multi-linear and ANN approaches across both calibration setups.

tions in the Multi-linear case start from a more optimal point than the ANN (approximately 0.37 in downstream calibration setup and 0.58 in upstream calibration setup) since they use a uniform background solution obtained by a global optimization method, as mentioned in section 3.2.3.1. Furthermore, they converge after around 200 iterations and remain monotonous throughout the optimization process. The ANN, with a significantly larger number of parameters (see Table C.1 in Appendix), can achieve a lower cost despite starting from a higher cost. Moreover, the ANN cost function in Figure 3.11 is not monotonous throughout training epochs: it shows localized increases for several instances within the first 100 epochs. This result can be attributed to the complexity of the regionalization mapping in each case, suggesting that the surface of the cost function is more complex for ANN than for multi-linear mapping. Moreover, the Adam algorithm used to optimize the ANN enables the exploration of different paths to locate the optimum, therefore avoiding get-

ting stuck in local minima, where significant changes occur in the control vector space (biases and weights). This property could be essential in tackling equifinality and reach robust global optimum even with different starting points.

Finally, Figure 3.12b shows the optimized parameter maps obtained with the two non-uniform regionalization methods across both calibration setups, that can be physically interpreted. In the

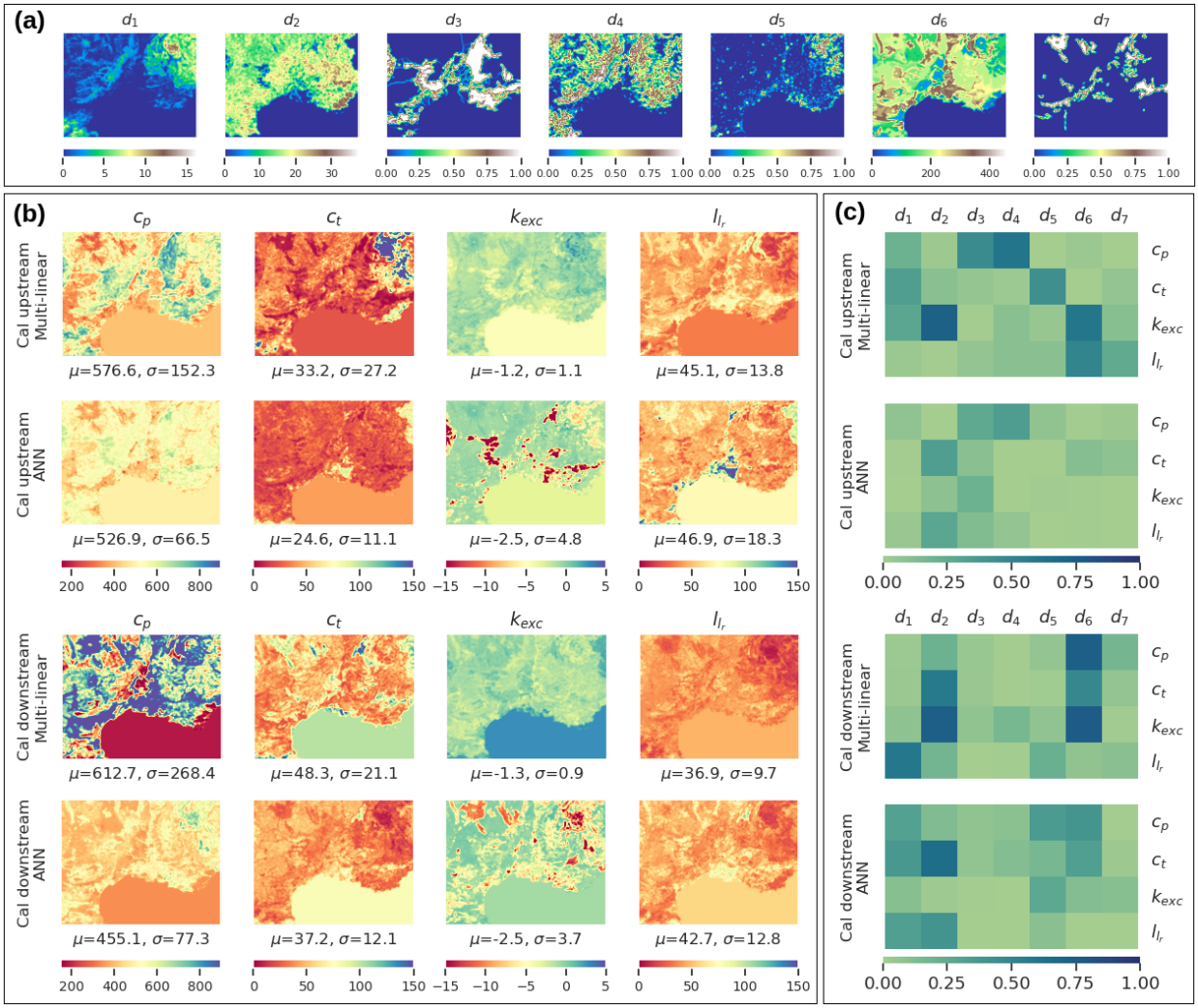


Figure 3.12 – Analysis of input descriptors and output model parameters for two regionalization methods (Multi-linear and ANN) across both calibration setups (upstream and downstream): (a) Spatial distribution of physical descriptors (d_1-d_7), details provided in Table 1.1; (b) Spatial distribution of calibrated hydrological parameters (c_p, c_t, k_{exc}, l_{l_r}) with μ and σ indicating their spatial average and standard deviation over the active cells within the spatial domain Ω; (c) Linear correlation between descriptors and parameters over the active cells within the spatial domain Ω.

downstream calibration setup, the spatial patterns of the drainage density descriptor (d_2) and the potential water reserve descriptor (d_6) in Figure 3.12a are easily observable in the distributed parameter maps in the multi-linear regression case, as evidenced by strong correlations between these descriptors and the capacity of the production store (c_p), the capacity of the transfer store (c_t), and the non-conservative water exchange (k_{exc}). Such correlations can be quantified using one-to-one parameter-descriptor correlation matrices as depicted in Figure 3.12c. Correlations are smaller in the ANN case than in the Multi-linear case: this suggests that the parameter maps

derived with the former go beyond a simple linear combination of descriptors, and hence include some degree of non-linearity.

Overall, the obtained regional hydrological parameter maps, together with the performances in interpolation and spatio-temporal extrapolation cases (as shown in section 3.3.2), demonstrate the relevance of the intrinsic spatial constraint introduced by the descriptors-to-parameters mappings. These mappings result in a regularizing effect for the overparameterized calibration problem in distributed hydrological modeling from sparse discharge data.

3.4 Discussions

Learning the spatial variability of conceptual hydrological parameters may be difficult to achieve with simple regionalization methods (e.g., based on multi-linear regression). However, while complex regional mappings can reduce the misfit between observed and simulated hydrological responses, their ability to produce physically interpretable results may be questioned. The various regionalization mappings define the way information is transferred from physical descriptors to hydrological parameters. It is important to ensure that HDA-PR effectively uses the physical patterns provided by descriptor maps to constrain the estimation of meaningful parameter maps. To address this question, it is worth noting that the "safest" approach is to use multi-linear regression, which corresponds to a simple weighted average of the descriptor patterns. The case of multi-polynomial regression (that was not included in this study), the risk of losing physical properties may arise when the polynomial degree is unbounded. To mitigate this risk, we propose imposing bounds on the polynomial degree, $0.5 \leq \beta_{k,d} \leq 2$. However, ANNs pose the most complicated scenario, where the control vector (that is, the weights and biases) consists of numerous parameters that are difficult to physically constrain. Our hands-on experience indicates that a multilayer perceptron with two or three hidden layers is sufficient for learning the parameters of a parsimonious conceptual distributed hydrological model without under- or over-extracting the physical information of the input descriptors. Note that the number of neurons in each layer must be reasonable, and should not exceed $\sqrt{N_D.N_x}$ based on our experiments. Next, to alleviate the vanishing gradient problem inherent to the ANN, we employed several techniques commonly used in the machine learning community. First, we applied Xavier initialization (Glorot et al., 2010) to the weights, maintaining a reasonable magnitude of the gradients. Second, we used the ReLU activation function or its variants in the hidden layers, enabling the gradient to flow more freely through the network. Third, we varied the number of hidden layers between 2 and 4, striking a balance between network flexibility and exacerbation of the vanishing gradient problem. Ultimately, we used a relatively high initial learning rate between 0.003 and 0.005 to prevent the gradients from shrinking excessively during training.

While the ANN approach showed superior performance in the upstream calibration setup, its performance did not exhibit significant improvement in the downstream scenario, where the Uniform approach already yielded quite satisfactory results compared to local calibration references. It is worth noting that while "extreme" setups such as upstream-only and downstream-only calibrations are useful in the academic context of this study, mixed setups that combine both, as well as include independent catchments, are likely to be employed in practice. For example, a calibration setup with a random selection of catchments will be tested in Chapter 4 and Chapter 5 for a subset of the 126 catchments presented in this study. Furthermore, the performance of calibration setups that use nested gauges could be improved by adapting the cost function for better selecting and weighting information between calibration gauges and flow signal parts (e.g., with differentiable flood signatures-based algorithm for calibration in Chapter 2), as well as by accounting for data and structural uncertainties in this cost function (Kuczera et al., 2010; Renard, Kavetski, Kuczera, et al., 2010).

Regarding regionalization over larger areas, such as for large basins or at country scales, an increased flexibility in the regional mapping might be needed to deal with significant physical heterogeneity. For instance, one may wish to use different transfer functions for different clusterings of the spatial domain (e.g., into sub-regions or hydrological response units (HRU)). This can be achieved through the use of spatialized regional controls, for example as done in regional calibration for catchment clusters determined with a similarity measure (Huang et al., 2019). In the proposed HDA-PR framework, mappings that vary with regions or HRU can be obtained by including indicators of such regions/HRUs in the list of descriptors. This additional flexibility would certainly be necessary to circumvent the rigidity of the multi-linear mapping, but maybe not for a flexible mapping such as the ANN one. Another interesting research avenue is to develop an automatic identification of effective physical descriptors from large databases as well as identification of optimal spatial flexibility for constructing effective regional data assimilation approaches.

Finally, an important research avenue is the development of a probabilistic framework to quantify the uncertainties surrounding the application of HDA-PR. Several challenges need to be addressed for this purpose, some of them being related to the spatialized nature of the hydrologic model, while others are more specific to the large dimensionality induced by the regionalization mappings, in particular the ANN one. First, accounting for the uncertainty affecting the forcing and response data used in calibration requires a modification of the cost function based on data error models. Such models need to be parsimonious (e.g., to represent spatialized rainfall errors using a controlled number of parameters, Mustafa et al. (2018)), yet recognize the specificities of some data acquisition procedures (for instance, the partly systematic nature of rating curve errors affecting streamflow, Horner et al. (2018)). Second, since the structural errors made by the hydrologic model are varying in space, a probabilistic model describing structural uncertainty would need to be regionalized, as is the hydrologic model itself. Third, estimation tools need to be able to cope with the high dimensionality of the estimation problem. For instance, the use of Markov Chain Monte Carlo methods will certainly require methods that can take advantage of the differentiable implementation of HDA-PR (Hoffman et al., 2014) and of parallel computing capabilities (Laloy et al., 2012; Syed et al., 2022). Alternatively, the search of regionalization mappings that retain the flexibility provided by ANNs while being more parsimonious is another promising perspective.

3.5 Conclusion

A Hybrid Data Assimilation and Parameter Regionalization (HDA-PR) approach has been introduced in this study. We investigated the potential of incorporating learnable regionalization mappings, including multivariate polynomial regressions and neural networks, into a differentiable high-resolution hydrological model. To the best of our knowledge, we present the first implementation of ANNs within this context, enabling a seamless regionalization in hydrology. Effective optimization algorithms capable of performing high-dimensional optimizations from multi-source data have been obtained with:

— effective regional transfer functions of adaptable complexity, enabling the use of information from heterogeneous data sources, with learning of a non-linear multivariate mapping in the case of the ANN, and providing effective spatial constraint of varied flexibility;
— a differentiable forward hydrological model, embedding the regional mappings, that enables accurate computation of spatially distributed gradients of the multi-gauge cost function - which is crucially needed in the context of sparse observations (i.e., cost evaluation locations), and relatively small gradient values;
— optimization algorithms, adapted to high-dimensional problems, with seamless flow of cost gradients, especially when combined with physical descriptors, which efficiently enhance the

transferability of geophysical properties from gauged to ungauged locations.

HDA-PR has been thoroughly evaluated with two calibration setups in the French Mediterranean region with a high-resolution spatio-temporal hydrological modeling approach using multi-gauge discharge and descriptors maps. The results obtained on both calibration setups, and especially on the most challenging calibration scenario using only upstream gauges, highlight the effectiveness of HDA-PR that utilizes physical descriptors, surpassing the performance of a uniform regionalization method with lumped model parameters. Moreover, the ANN exhibited superior performance compared with multi-linear regression, that is, an approach with the same complexity as transfer functions used in previous studies on descriptors-based regionalization with a seamless calibration scheme. The NSE scores of HDA-PR with ANN in temporal validation surpass 0.7 for both calibration setups, which is a fairly good performance compared to the reference benchmarks of around 0.78 (obtained in local calibration), thereby establishing its remarkable capability in challenging modeling scenarios (i.e., upstream calibration) as well as its capacity to collapse to a simpler mapping in less challenging ones (i.e., downstream calibration). Various flood event signatures are also used as validation metrics to demonstrate the robustness of HDA-PR, where the two regionalization methods using descriptors outperform the uniform regionalization method. Particularly, the ANN-based mapping yields the best performance for all validation scenarios based on flood event signatures. Interestingly, the ANN enables extracting information from physical descriptors and learning non-linear multivariate descriptors-to-parameters mappings while providing better model controllability than a linear mapping for complex calibration cases. The exploration of more complex dataset and neural networks architecture, on top of other differentiable flow models, represent a very promising axis for further research.

This research and the proposed algorithms open several perspectives. Immediate work focuses on: (i) the testing and improvement of HDA-PR for application at national scales and on other continents, such as its operational application to estimate the regional hydrological model (at a 15-min temporal resolution) for the national flash flood early warning system called "Vigicrues Flash" in France (Javelle et al., 2019; Piotte et al., 2020); (ii) study of effective descriptor selection along with multi-gauge cost functions explicitly accounting for data uncertainties, and optimal spatial clustering of regional controls, for example into HRU; (iii) information selection with a signature-based cost function (Chapter 2) and spatial weighting strategies for multi-site discharge over river networks; (iv) implementation of a framework to quantify predictive uncertainty at both gauged and ungauged sites.

HDA-PR can be extended to state and composite parameters-states optimization which could be very interesting for multi-scale data assimilation and real time model correction from multi-source and multi-site data. Adding a learnable feature extraction layer fom images (CNN) or time series (LSTM) on top of the regionalization transfer functions would enable the exploration of larger and richer databases including time varying data. Finally, HDA-PR is transposable to regionalization of differentiable integrated hydrological-hydraulic networks models (e.g., Pujol, Garambois, and Monnier (2022)) and could be used to explore regionalization potential from cocktails of in-situ and satellite data, including the forthcoming SWOT (Surface Water and Ocean Topography satellite mission) observations of water surfaces variabilities of worldwide larger rivers. In general, its applicability extends beyond hydrological models and can be adapted to other geophysical models.

Chapter 4

A Bayesian-Weighted Estimation Approach for Spatially Distributed Models under Equifinality Issues

Contents

4.1 Introduction **80**
4.2 Method **80**
4.2.1 Revisiting regionalization and calibration framework 81
4.2.2 From low-dimensional to high-dimensional Bayesian-weighted estimates . . 82
4.3 Results **83**
4.3.1 Numerical experiment 83
4.3.2 Model performance and analysis 84
4.4 Discussion **86**
4.5 Conclusion **87**

This chapter is based on Huynh, Garambois, Colleoni, Renard, and Roux (2023), a conference paper presented at *Colloque SHF, prévision des crues-inondations, Toulouse, 2023*), with significant methodological improvements and modifications to align with the overall objectives and structure of this thesis. This study aimed to investigate an original likelihood and Bayesian inspired estimation approach from Gejadze et al. (2022) for parameter optimization in context of equifinality.

4.1 Introduction

Hydrological modeling confronts fundamental challenges in parameter estimation within ill-posed inverse problems, a complexity eloquently captured by the concept of equifinality (Beven, 2006). This phenomenon challenges the traditional paradigm of seeking a singular, optimal model parameterization. The equifinality thesis reveals two critical dimensions of model uncertainty: structural and spatial equifinality (Beven, 2006). Structural equifinality, as demonstrated by Gupta, Kling, et al. (2009), emerges from multiple model configurations producing equivalent predictive outcomes. Spatial equifinality introduces additional complexity by revealing how spatially distributed parameters can yield similar model performances despite significant local variations (Sivapalan, 2003). Gradient-based optimization algorithms in variational data assimilation (VDA) frequently encounter computational challenges due to multiple local minima in the cost function. Dennis et al. (1977) and subsequent hydrological modeling studies by Arsenault et al. (2014) have highlighted optimization algorithms' sensitivity to initial parameter estimates (also known as "first guess", "prior", or "background" value). L. Pan et al. (1998) and Skahill et al. (2006) proposed methodological approaches to mitigate convergence to local optima by exploring multiple parameter configurations. The persistent challenge of parameter non-uniqueness is particularly pronounced in distributed hydrological models, where traditional calibration approaches often fail to adequately represent parameter uncertainty and model structural limitations as demonstrated by Clark et al. (2008) and Hrachowitz, Fovet, et al. (2014).

In Chapter 3, efficient regionalization algorithms have been proposed for spatially distributed hydrological modeling, utilizing descriptors-to-parameters mappings through neural networks and multivariate regressions within a VDA framework. This approach transforms ill-posed inverse problems by introducing spatial constraints and regularization. Despite these advancements, significant challenges persist in parameter estimation. The inherent sensitivity to initial parameter estimates introduces substantial uncertainty, potentially compromising the reliability and transferability of hydrological model predictions across spatial domains. This limitation is particularly pronounced when employing multi-polynomial descriptors-to-parameters mappings with optimization algorithms that exhibit limited flexibility in escaping local minima.

This chapter presents a Bayesian-weighted estimation approach based on the Bayesian averaging of an ensemble of optimal estimates, initially introduced by Chelil et al. (2022) and Gejadze et al. (2022). The proposed method aims to reduce equifinality in the parameterization of a spatially distributed hydrological model. It is applied within the regionalization framework introduced in Chapter 3 to infer a mapping between physical descriptors and spatially distributed conceptual hydrological parameters using information from multi-gauge river flow observations.

4.2 Method

This study focuses on enhancing background values used for gradient-based optimization algorithms in the calibration process with regionalization multi-polynomial mappings (Figure 4.1).

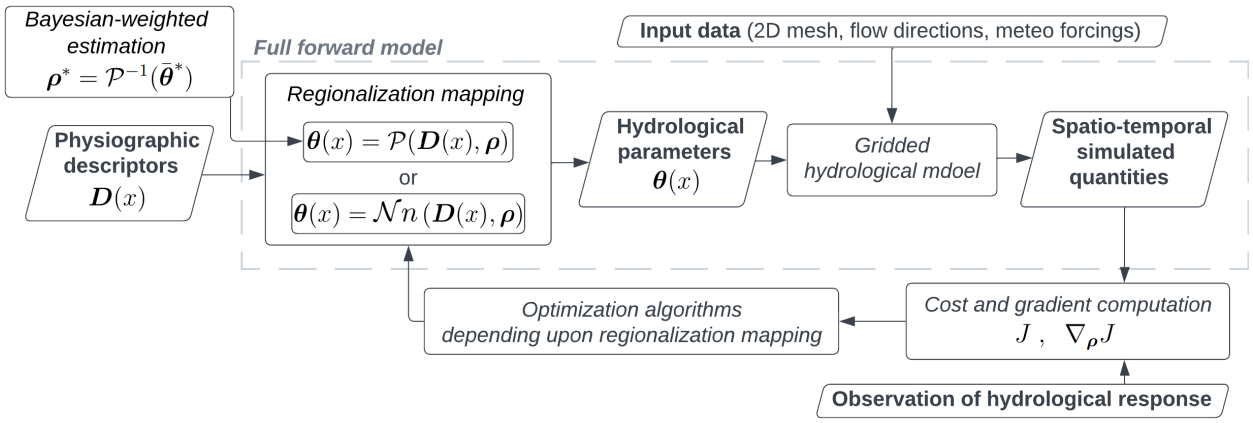


Figure 4.1 – Flowchart of the regionalization framework using Bayesian-weighted estimate of background values.

4.2.1 Revisiting regionalization and calibration framework

First, the forward hydrological model is defined in Section 3.2.1 of Chapter 3. Recall that in this case, the regionalization mapping $\phi \equiv \mathcal{P}$ with tunable parameter $\boldsymbol{\rho}$ consists of a multivariate polynomial regression between input physical descriptors $\boldsymbol{D}(x)$ and hydrological model parameters $\boldsymbol{\theta}(x, \boldsymbol{D}, \boldsymbol{\rho}) \coloneqq \mathcal{P}(\boldsymbol{D}(x), \boldsymbol{\rho})$, such that:

$$\theta_k(x, \boldsymbol{D}, \rho_k) \coloneqq s_k \left(\alpha_{k,0} + \sum_{d=1}^{N_D} \alpha_{k,d} D_d^{\beta_{k,d}}(x) \right), \forall k \in [1..N_\theta], \forall x \in \Omega \tag{4.1}$$

with $s_k(.)$ a Sigmoid-based transformation imposing bound constraints in the forward hydrological model. The lower and upper bounds are assumed to be spatially uniform for each parameter field θ_k of the hydrological model. Then, the inverse problem 3.9 can be formulated as:

$$\arg\min_{\boldsymbol{\rho}} J\left(\boldsymbol{Q}^*, \mathcal{M}_{rr}(., \boldsymbol{\theta} = \mathcal{P}(., \boldsymbol{\rho}))\right) \tag{4.2}$$

where J is the cost function, $\mathcal{M}_{rr}$ represents the rainfall-runoff model, and $\boldsymbol{Q}^*$ denotes the observed discharge time series at multiple gauges.

The control vector $\boldsymbol{\rho} \equiv \left[(\rho_k)_{k=1}^{N_\theta}\right]^T \equiv \left[\left(\alpha_{k,0}, (\alpha_{k,d})_{d=1}^{N_D}\right)_{k=1}^{N_\theta}\right]^T$ of problem 4.2 is optimized using the L-BFGS-B algorithm (Zhu et al., 1997), adapted for high-dimensional controls. In this study, no bound constraints are imposed on $\alpha_{k,\square}$, while the exponents $\beta_{k,d}$ are fixed to 1 (multi-linear regression). This algorithm requires the gradient of the cost function with respect to the sought parameters $\nabla_{\boldsymbol{\rho}} J$. This gradient is computed by solving the adjoint model, which is obtained by automatic differentiation using the Tapenade engine (Hascoet et al., 2013). The convergence criterion involves reaching a maximum iteration limit or meeting conditions related to cost function change or gradient magnitude.

It is worth noting that determining a background value $\boldsymbol{\rho}^*$ is important for the convergence of this algorithm. It is used as a starting point for the optimization, and is defined from a spatially uniform prior $\bar{\boldsymbol{\theta}}^*$ as:

$$\boldsymbol{\rho}^* \equiv \left[\alpha_{k,0} = s_k^{-1}\left(\bar{\theta}_k^*\right), (\alpha_{k,d} = 0, \beta_{k,d} = 1)\right]^T, \forall (k,d) \in [1..N_\theta] \times [1..N_D] \tag{4.3}$$

where $s_k^{-1}(z) = \ln\left(\frac{z - l_k}{u_k - z}\right)$ is the inverse Sigmoid. The spatially uniform low-dimensional (LD) prior $\bar{\boldsymbol{\theta}}^*$ is determined considering the cost function without regionalization, i.e., $\boldsymbol{\rho} \coloneqq \bar{\boldsymbol{\theta}}$, and classically

using a derivative-free global (in terms of "full parameter space exploration") optimization algorithm (SBS in Michel (1989)), which will be explained in the following section.

4.2.2 From low-dimensional to high-dimensional Bayesian-weighted estimates

A Bayesian-weighted estimate (cf. Gejadze et al. (2022) and Chelil et al. (2022)) is used to look at the mean of the posterior distribution $f\left(\boldsymbol{\theta}|\boldsymbol{Q}^*\right)$ that is more stable in context of equifinality than searching its mode (inverse problem 3.9, maximum a posteriori probability (MAP) search is the essence of VDA). A prior probability distribution f is used to generate a sample of spatially uniform parameter sets $\boldsymbol{\theta}^{(i)}, \forall i \in 1..N$ within the hypercube defined by parameters bounds $[l_k, u_k], \forall k \in 1..N_\theta$. The likelihood function is defined as:

$$\mathcal{L}_\gamma^{(i)} = \exp\left(-2^\gamma \left(\frac{J^{(i)}}{J^-} - 1\right)^2\right) \tag{4.4}$$

where γ is a parameter controlling the decay rate of this function that compares the cost value obtained with parameter set i, $J^{(i)} = J\left(\boldsymbol{\theta}^{(i)}\right)$, to the minimum cost value over the sample of N parameter sets, $J^- = \min_{i=1..N} J^{(i)}$. The posterior ensemble mean and variance in the case of LD Bayesian-weighted estimate are computed as follows:

$$\begin{aligned} \bar{\boldsymbol{\theta}}_\gamma^* &= \frac{1}{K}\sum_{i=1}^{N}\left(\mathcal{L}_\gamma^{(i)} \cdot \boldsymbol{\theta}^{(i)} \odot f\left(\boldsymbol{\theta}^{(i)}\right)\right) \\ \mathrm{Var}\left(\bar{\boldsymbol{\theta}}_\gamma^*\right) &= \frac{1}{K}\sum_{i=1}^{N}\left(\mathcal{L}_\gamma^{(i)} \cdot \left(\boldsymbol{\theta}^{(i)} - \bar{\boldsymbol{\theta}}_\gamma^*\right) \odot \left(\boldsymbol{\theta}^{(i)} - \bar{\boldsymbol{\theta}}_\gamma^*\right) \odot f\left(\boldsymbol{\theta}^{(i)}\right)\right) \end{aligned} \tag{4.5}$$

where $K = \sum_{i=1}^{N} \mathcal{L}_\gamma^{(i)} \cdot f\left(\boldsymbol{\theta}^{(i)}\right)$ and "$\odot$" denotes the Hadamard product, a simple scalar product between vectors here but usable with higher dimensional controls. The parameter γ (typically ranging from -20 to 20) is determined using the L-curve approach, considering a parametric curve $\left\{J\left(\bar{\boldsymbol{\theta}}_\gamma^*\right), D_\gamma\right\}$, where the probabilistic (Mahalanobis) distance between the estimate $\bar{\boldsymbol{\theta}}_\gamma^*$ and the average prior $\bar{\boldsymbol{\theta}}^0 = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\theta}^{(i)}$ is defined as:

$$D_\gamma = \mathrm{Var}\left(\bar{\boldsymbol{\theta}}_\gamma^*\right)^{-1} \odot \left(\bar{\boldsymbol{\theta}}_\gamma^* - \bar{\boldsymbol{\theta}}^0\right) \odot \left(\bar{\boldsymbol{\theta}}_\gamma^* - \bar{\boldsymbol{\theta}}^0\right) \tag{4.6}$$

The optimal value of γ (denoted as $\hat{\gamma}$) is determined at the "corner" of the L-curve, where both $J\left(\bar{\boldsymbol{\theta}}_\gamma^*\right)$ and D_γ are minimized:

$$\bar{\boldsymbol{\theta}}^* = \bar{\boldsymbol{\theta}}_{\hat{\gamma}}^* \quad \text{where} \quad \hat{\gamma} = \arg\min_\gamma \left\| J\left(\bar{\boldsymbol{\theta}}_\gamma^*\right) - D_\gamma \right\|_2^2 \tag{4.7}$$

In this way, estimating the spatially distributed high-dimensional (HD) prior $\boldsymbol{\theta}^*(x)$ (typically obtained by pre-calibration using a spatially uniform prior) can be achieved by applying Equation 4.5 to each pixel $x \in \Omega$, with spatially distributed parameter sets $\boldsymbol{\theta}^{(i)}(x), \forall i \in 1..N$ and the corresponding spatialized prior probability distribution f_x. The likelihood function $\mathcal{L}_\gamma^{(i)}$ remains spatially uniform for all pixels. Thus, Equation 4.5 can be rewritten as follows in case of HD Bayesian-weighted estimation, $\forall x \in \Omega$:

$$\begin{aligned}\boldsymbol{\theta}^*_\gamma(x) &= \tfrac{1}{K_x}\sum_{i=1}^{N}\left(\mathcal{L}^{(i)}_\gamma \cdot \boldsymbol{\theta}^{(i)}(x) \odot f_x\left(\boldsymbol{\theta}^{(i)}(x)\right)\right)\\ \mathrm{Var}\left(\boldsymbol{\theta}^*_\gamma(x)\right) &= \tfrac{1}{K_x}\sum_{i=1}^{N}\left(\mathcal{L}^{(i)}_\gamma \cdot \left(\boldsymbol{\theta}^{(i)}(x) - \boldsymbol{\theta}^*_\gamma(x)\right) \odot \left(\boldsymbol{\theta}^{(i)} - \boldsymbol{\theta}^*_\gamma(x)\right) \odot f_x\left(\boldsymbol{\theta}^{(i)}(x)\right)\right)\end{aligned} \tag{4.8}$$

where $K_x = \sum_{i=1}^{N} \mathcal{L}^{(i)}_\gamma \cdot f_x\left(\boldsymbol{\theta}^{(i)}(x)\right)$.

Then, denoting the pixel-averaged prior as $\boldsymbol{\theta}^0(x) = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{\theta}^{(i)}(x)$, the local Mahalanobis distance at each pixel is defined as follows:

$$D_\gamma(x) = \mathrm{Var}\left(\boldsymbol{\theta}^*_\gamma(x)\right)^{-1} \odot \left(\boldsymbol{\theta}^*_\gamma(x) - \boldsymbol{\theta}^0(x)\right) \odot \left(\boldsymbol{\theta}^*_\gamma(x) - \boldsymbol{\theta}^0(x)\right) \tag{4.9}$$

The total Mahalanobis distance is simply defined as the aggregation of local Mahalanobis distances over the 2D spatial domain:

$$D_\gamma = \int_{x\in\Omega} D_\gamma(x)dx \tag{4.10}$$

Finally, the HD Bayesian-weighted estimate, using the L-curve approach to find an optimal value for the decay parameter γ, is:

$$\boldsymbol{\theta}^*(x) = \boldsymbol{\theta}^*_{\hat{\gamma}}(x) \quad \text{where} \quad \hat{\gamma} = \arg\min_\gamma \left\| J\left(\boldsymbol{\theta}^*_\gamma\right) - D_\gamma \right\|_2^2 \tag{4.11}$$

4.3 Results

4.3.1 Numerical experiment

The proposed algorithms are tested on a high-resolution regional modeling of a flash flood prone area, with heterogeneous physical properties including karstic areas, located in the South-East of France. This is a subset of the 126 catchments in the ArcMed region, presented in Section 3.3.1 of Chapter 3. A total of 11 gauged catchments are employed as "donor" catchments for multi-gauge calibration (global cost function), while nine other catchments are treated as pseudo-ungauged for spatial validation to assess regionalization capabilities of the proposed algorithms. Multiple gauges downstream of nested and independent catchments are simultaneously considered, enabling multi-gauge optimization. The time series of observed discharge on donor catchments are of the same length but the method by design enables to work with more sparse and heterogeneous data. In this study, a set of seven physical descriptors (see Table 1.1 in Chapter 1) available throughout Metropolitan France is used to learn the regional transfer functions.

For simplicity and to reduce computational costs in obtaining "good" spatialized prior parameters via a pre-calibration process, this study performs tests only on the LD Bayesian-weighted estimation. However, future work could investigate the efficiency of the HD approach to determine if it is more effective and further reduces the sensitivity to background values. We compare and analyze:

- local uniform $\boldsymbol{\rho} \equiv \bar{\boldsymbol{\theta}}$ and full spatially distributed $\boldsymbol{\rho} \equiv \boldsymbol{\theta}(x)$ calibrations for each gauges, that are respectively under- and over-parameterized hydrological optimization problems, but are served as reference performances ("Uniform (local)" and "Distributed (local)");

and multi-gauge regional calibration methods with:

- lumped model parameters $\boldsymbol{\rho} \equiv \bar{\boldsymbol{\theta}}$ which somehow represents "level 0" regionalization ("UR");
- a multivariate linear mapping (i.e., $\boldsymbol{\rho} \equiv [\alpha_{k,0}, (\alpha_{k,d}, 1)]^T$) using a classical first guess obtained by SBS algorithm ("M2R"), or Bayesian-weighted estimation ("BM2R");

— a multilayer perception (i.e., $\boldsymbol{\rho} \equiv [\boldsymbol{W}, \boldsymbol{b}]^T$) ("ANNR").

Two study periods, namely P1 (August 2011 – July 2015) and P2 (August 2015 – July 2019), are considered for split sample testing. A two-fold cross-temporal calibration approach is employed, where the models are calibrated on one period and validated on the other period. In each case, we consider three types of validation: spatial validation (performance in pseudo-ungauged catchments during the calibration period), temporal validation (performance in gauged catchments during the validation period), and spatio-temporal validation (performance in pseudo-ungauged catchments during the validation period). Note that no warm-up of the forward model is performed for this calibration algorithms comparison.

4.3.2 Model performance and analysis

The performance of all calibration and regionalization methods is presented in Figure 4.2. Un-

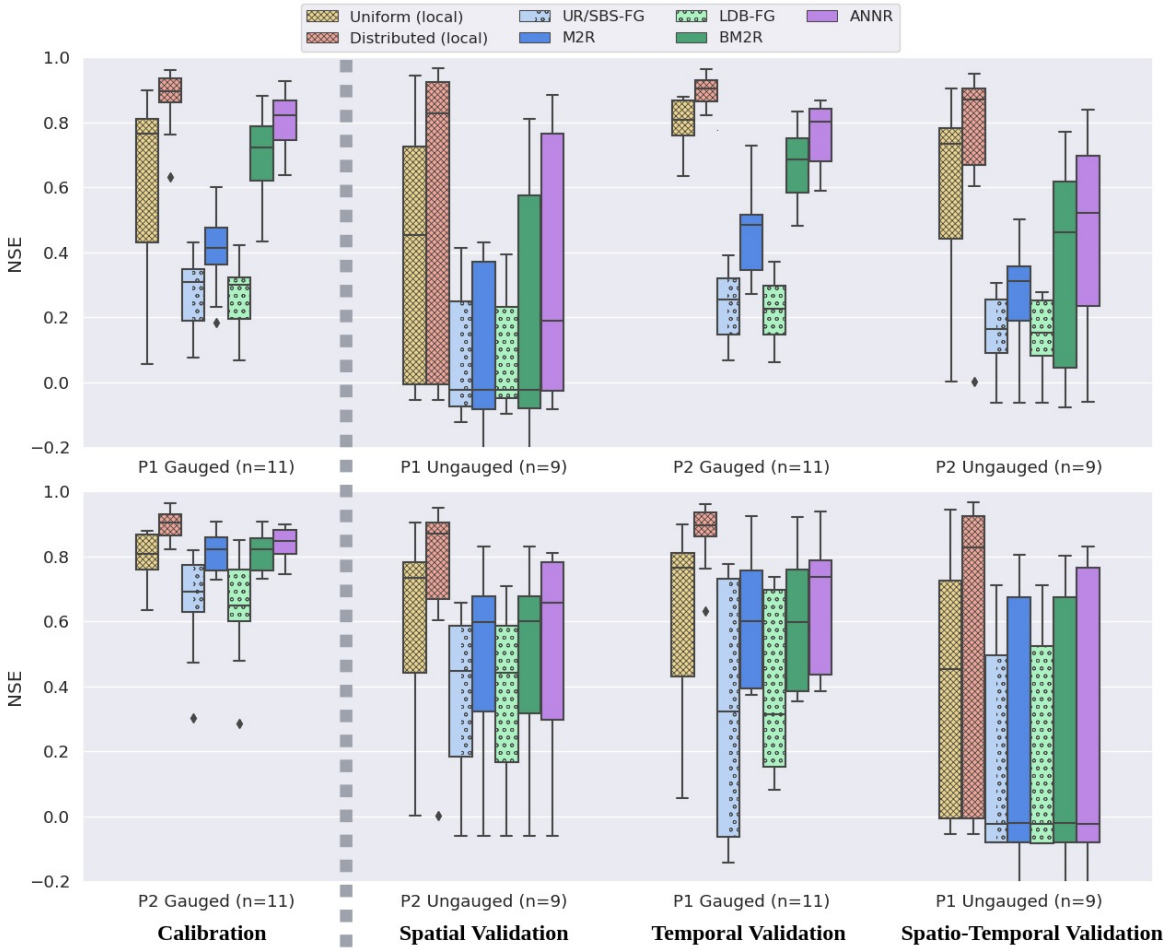


Figure 4.2 – Performance of all calibration and regionalization methods when calibrated on P1 (upper sub-figure) and on P2 (lower sub-figure). BM2R uses the first guess obtained by Low-Dimensional Bayesian estimation (LDB-FG), while the first guess of M2R is obtained by the SBS algorithm (SBS-FG), which also represents the solution of the regionalization method with lumped parameters (UR); n denotes the number of study catchments.

surprisingly, spatially uniform calibration (UR) leads to limited performance in calibration and poor performance in regionalization, especially when compared to the reference local spatially distributed

calibration that is overparameterized. The regionalization methods, which incorporate information from multi-gauge discharge as well as physical descriptor maps, all yield relatively satisfying performances in calibration and temporal validation at pseudo-ungauged sites (median NSE scores higher than 0.4 when calibrated on P1). The regionalization approach based on ANN (ANNR) achieves the best results for both gauged and pseudo-ungauged catchments. In spatio-temporal validation, the most challenging case especially on this zone that is difficult to model, performance are relatively low, with regionalization methods or spatially uniform regionalization (UR/SBS-FG) and further investigations are required.

Regarding the determination of prior parameter sets for the multi-linear regionalization mapping, the Bayesian estimation approach (LDB-FG) demonstrates fairly good performance, comparable to that obtained with the global heuristic algorithm (SBS-FG), in calibration and spatial validation, with only minor differences in temporal validation. We believe that this is reflective of the importance of exploring a Bayesian approach for the definition of the cost function, which would enable intrinsic weighting of model misfits to different gauged hydrological behaviors. Moreover, when considering the performances of M2R and BM2R on P1 (upper sub-figure of Figure 4.2), the Bayesian approach exhibits markedly higher performance in calibration and validations, while relatively similar performances are observed on P2 (lower sub-figure of Figure 4.2). This difference may be attributed to variations in hydrological information on P1 and potentially higher data errors, which have a lesser impact on the Bayesian approach.

Table 4.1 represents several statistical quantities of the distributed parameter maps obtained through different regionalization approaches. All methods result in distinct parameter maps and

Table 4.1 – The optimal parameters obtained by different methods for each calibration period. Spatially distributed parameters are represented as the median (mean, std). The rows in italic and smaller text present the spatially uniform first guess obtained by the SBS algorithm (SBS-FG) used for M2R, and the Low-Dimensional Bayesian estimation (LDB-FG) used for BM2R.

Method (Cal period)	c_p	c_t	k_{exc}	l_{l_r}
UR/*SBS-FG* (P1)	2000	470.88	3.04	55.12
UR/*SBS-FG* (P2)	322.68	239.65	-3.18	45.53
M2R (P1)	2000 (2000, 0)	113.57 (397.78, 436.27)	-0.15 (-3.8, 8.35)	52.79 (83.63, 79.27)
M2R (P2)	398.64 (658.28, 623.15)	338.41 (433.16, 351.08)	-0.53 (-1.56, 5.29)	94.16 (100.66, 75.71)
LDB-FG (P1)	*1917.74*	*527.77*	*2.9*	*72.32*
LDB-FG (P2)	*138.53*	*517.6*	*-13.04*	*53.71*
BM2R (P1)	76.02 (700.93, 869.81)	599.93 (540.72, 404.87)	-13.33 (-10.0, 9.94)	63.47 (92.67, 87.33)
BM2R (P2)	396.71 (651.86, 607.05)	348.38 (439.31, 353.62)	-0.57 (-1.47, 4.98)	95.65 (100.98, 78.13)
ANNR (P1)	369.87 (700.95, 737.21)	393.52 (492.71, 334.31)	-3.63 (-6.92, 7.58)	56.63 (71.04, 62.52)
ANNR (P2)	404.13 (593.4, 477.85)	510.9 (444.66, 220.29)	-2.23 (-2.12, 2.34)	62.72 (67.28, 32.5)

varying levels of temporal stability (see Figure 4.3). The ANNR leads to the most robust inference over P1 and P2, with remarkably stable average parameter values as well as spatial standard deviation over time. The priors inferred with SBS-FG or LDB-FG exhibit slight differences and also lead to a different optimum during regionalization for P1. Interestingly, the opposite trend is observed for P2, where data uncertainty and model adequacy might be better, resulting in similar functioning points after regionalization despite substantially different priors determined with SBS-FG or LDB-FG.

Last but not least, during calibration on P1, M2R and BM2R lead to a negative exchange coefficient ($k_{exc} < 0$), despite starting from priors with positive exchange values (3.04 for SBS-FG (P1) and 2.9 for LDB-FG (P1)); AANR also lead to negative exchange. This intriguing result, of reaching systematically significant negative exchange, is particularly noteworthy because the exchange coefficient directly impacts mass conservation. These findings relate to those on flash

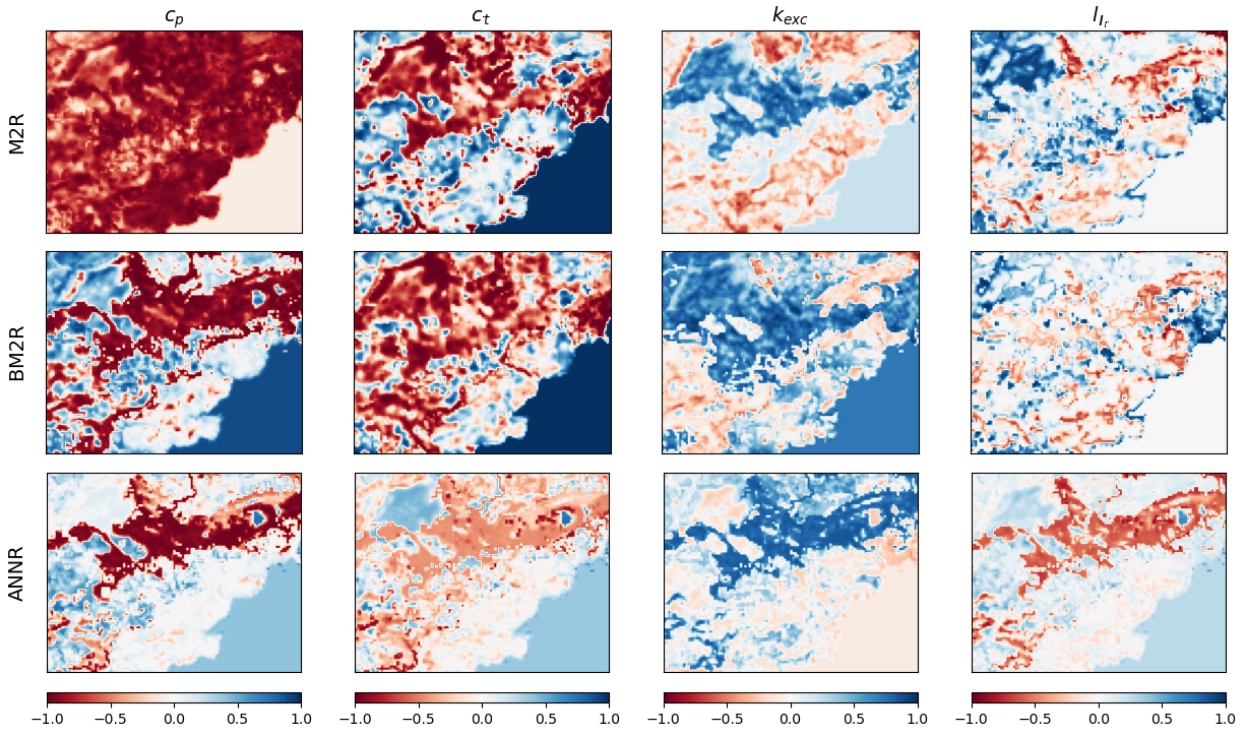


Figure 4.3 – Normalized temporal stability over the periods P1 and P2 of the optimal distributed parameters $\frac{\hat{\theta}_k^{(P2)} - \hat{\theta}_k^{(P1)}}{u_k - l_k}, k \in [1..N_\theta]$, for the three regionalization approaches using physical descriptors.

floods water balance sensitivity and regionalization based on geological descriptors presented by Garambois, Roux, Larnier, Labat, et al. (2015b) for catchments in the same and nearby areas. Their event process-oriented and conservative model required an increase in modeled soil volume, while pedological and geological descriptors provided valuable constraining information, especially in the context of flash floods.

4.4 Discussion

One may question the choice of prior estimation: why not apply the estimation method directly to the control vector $\boldsymbol{\rho}$ instead of to the conceptual parameter $\boldsymbol{\theta}$? In the proposed framework, conceptual parameters $\boldsymbol{\theta}$ are linked to the control vector $\boldsymbol{\rho}$ through a regionalization mapping, which is a multi-polynomial function $\mathcal{P}$, such that:

$$\boldsymbol{\theta} = \mathcal{P}(\cdot, \boldsymbol{\rho}) \tag{4.12}$$

Additionally, for a given spatially uniform parameter value, the corresponding control vector can be obtained using the formulation in Equation 4.3. For LD Bayesian-weighted estimation, the estimation method is applied directly within the sampling space. For HD Bayesian-weighted estimation, where priors for spatially distributed parameters are required, the process begins with sampling a spatially uniform prior control in the same manner as in the LD case. This is followed by a rapid optimization step to derive a distributed prior, and finally, the estimation method is applied at the pixel scale, using an adapted computation of the parametric curve, including the Mahalanobis distance.

Thus, the proposed framework can be interpreted as a sampling process performed on $\boldsymbol{\rho}$, but with a "reduced control space dimensionality", rather than direct sampling in the conceptual parameter space. The key challenge of performing unconstrained sampling over the entire control space lies in its high dimensionality and the difficulty of defining meaningful priors. Unlike conceptual parameters, for which prior knowledge can often be inferred from hydrological understanding, obtaining an equivalent prior in the control space is far more complex.

By formulating priors in the conceptual parameter space and leveraging a Bayesian-weighted estimation approach, the method effectively constrains the search space, improves parameter inference, and enhances model reliability in a spatially distributed hydrological framework.

4.5 Conclusion

A Bayesian-weighted calibration algorithm has been tested in this study, on top of the Hybrid Data Assimilation Parameter Regionalization (HDA-PR) approach, enabling seamless regionalization in spatialized hydrology. The methods were tested in a challenging flash flood-prone area in the South-East of France, characterized by diverse physical properties and hydrological responses. Overall, the methods demonstrated satisfactory performance in several aspects: (i) accurate modeling of discharge at both gauged and pseudo-ungauged sites, and (ii) effective identification of conceptual parameters and extraction of information from physical descriptors.

Notably, the ANN-based regionalization method outperformed other approaches in terms of discharge accuracy and parameter stability. Bayesian-weighted estimation exhibited good performance and relative robustness, even in challenging cases like calibration on the period P1, where data uncertainty and model inadequacy were assumed to be higher. While the Bayesian-weighted method is computationally more demanding than traditional low-dimensional calibration algorithms, it can be efficiently parallelized. Moreover, this approach can be extended to higher-dimensional contexts, such as determining spatially distributed priors and exploring spatial equifinality using the VDA algorithms.

Interestingly, in contrast to the ANN, the regression methods provided insights into more complex modeling situations and potential data-model discrepancies. This highlights the importance of maintaining both "classical" approaches and AI-based solutions in research and applications, particularly in the continuous development of physically and mathematically interpretable methodologies.

Chapter 5

Inferring a Process-Parameterization Neural Network for Refining Internal Fluxes within a Spatialized Rainfall-Runoff Model

Contents

5.1 Introduction . . . 89
5.2 Modeling approach . . . 90
5.2.1 Rainfall-runoff model with regionalizable parameters and process-parameterization neural network . . . 90
5.2.2 Pixel-to-pixel flow routing of runoff with a partial differential equation and regionalizable parameters . . . 93
5.2.3 Learnable mappings for spatialized GR-like model and kinematic wave routing 93
5.2.4 Inverse problem and analysis of the hybrid physics-AI framework . . . 94
5.3 Data and experimental design . . . 94
5.4 Results and discussion . . . 96
5.4.1 Model performance analysis . . . 96
5.4.1.1 Local calibration over 235 French catchments . . . 96
5.4.1.2 Multi-catchment regionalization over a flash-flood prone Mediterranean area . . . 98
5.4.2 Towards learning hydrological behaviors . . . 100
5.4.2.1 Analysis of internal flux corrections . . . 101
5.4.2.2 Hybridization effect on main mass fluxes involved in basin's water balance . . . 103
5.4.3 Research perspectives and further discussion . . . 106
5.5 Conclusion . . . 107

This chapter is based on Huynh, Garambois, Renard, Colleoni, et al. (2025) published in *Hydrology and Earth System Sciences (HESS)*, with modifications to align with the overall objectives and structure of this thesis. The methodological development builds upon the regionalization approach presented in Chapter 3 and explores the use of an additional neural network within the hydrological model to correct internal water fluxes and enhance learnability. The key idea is to leverage physics as a strong constraint in a spatially distributed manner, taking advantage of the numerical differentiability of the `smash` model.

5.1 Introduction

In Chapter 3, we proposed a spatially distributed hybrid approach, HDA-PR, which incorporates a regionalization neural network into the forward model. This approach has enhanced regional flash flood modeling at a relatively high-resolution ($dx = 1$ km, $dt = 1$ h). Meanwhile, C. Wang et al. (2024) developed a large-scale spatialized hybrid hydrological model that improves evapotranspiration modeling for the Amazon basin by incorporating a replacement neural network and a regionalization neural network. In their study, C. Wang et al. (2024) combine a conceptual hydrological model with a simple bucket-based routing structure, using a hybrid approach that predicts conceptual parameters and corrects evapotranspiration (adjusting a Penman-Monteith estimate via a 3D convolutional neural network (CNN)) without correcting other internal fluxes, and operates at a coarser resolution ($dx = 0.5°$, $dt = 1$ day). In addition, enhancing the physical modeling of runoff and flood propagation necessitates the use of hydraulic models that must be differentiable to facilitate gradient-based optimization, a requirement for effective hybrid modeling. This approach has been demonstrated for the optimization of large parameter sets from heterogeneous data at the river network scale using various approaches based on differentiable hydraulic models, including: a 2D or multi-dimensional complete shallow water model coupled with a differentiable semi-lumped GR model (Pujol, Garambois, and Monnier, 2022); a 1D Saint-Venant river network model (e.g., Larnier, Garambois, et al. (2025), though without a differentiable hydrological model); and a kinematic wave model integrated into a grid differentiable spatialized hydrological model (Huynh, Garambois, Colleoni, Renard, Monnier, et al., 2024). These hydraulic models consist in partial differential equations (PDE). Therefore, hybrid hydrological models embedding neural networks should advance from lumped, semi-lumped or spatialized ODE-based models, to spatially distributed differentiable hybrid PDE-based hydrological-hydraulic models in view to improve modeling capabilities at basin scale and realism, at least for hydraulic processes which modeling is less uncertain than hydrological processes occurring in the earth critical zone (the near-surface environment where complex interactions between water, soil, rock, and living organisms regulate the Earth's surface dynamics).

To address the aforementioned challenges, this research proposes a spatially distributed hybrid modeling framework combining process-based modeling and neural networks, which is amenable to hydrological-hydraulic (H&H) models and other geophysical models. It builds on a parsimonious, GR-like hydrological model well-suited for flood modeling and regionalization learning, coupled with a differentiable kinematic wave model for spatially distributed flow routing. Additionally, a simple neural network is introduced to correct internal water fluxes within the rainfall-runoff model. A relatively "parsimonious hybridization" with a dense flux correction neural network is proposed, as it effectively captures non-linear flux corrections while leveraging the memory effect already embedded in the original hydrological model. This reduces the need for recurrent architectures such as LSTMs, which are typically designed for sequential data but introduce additional complexity that is unnecessary for this application. The approach is implemented into the open-source `smash` software that enables multiscale modeling, numerical adjoint model derivation via automatic differ-

entiation and variational data assimilation (VDA). The performance, robustness and interpretability of the proposed approach are studied at a relatively high spatio-temporal resolution of 1 km and 1 h, over a large sample of French catchments and also over a challenging flash flood-prone area. This study aims to demonstrate the feasibility and advantages of distributed hybrid modeling for spatio-temporal learning of hydrological processes at basin and regional scales. It provides a general framework for PDE-based spatially distributed modeling, taking advantage of AI and big data.

5.2 Modeling approach

The forward differentiable hybrid model $\mathcal{M}$ is obtained by partially composing (i) a dynamic process-based, differentiable, and spatially distributed rainfall-runoff model ($\mathcal{M}_{rr}$) with simplified hydraulic routing ($\mathcal{M}_{hy}$)—referred to as $\mathcal{M}_{rr-hy}$ for simplicity—with (ii) a learnable (neural network-based), differentiable process-parameterization and regionalization operator ϕ, resulting in Equation 5.1:

$$\mathcal{M} = \mathcal{M}_{rr-hy}\left(\,.\,,\ \phi\left(.\right)\right) \tag{5.1}$$

The spatially distributed rainfall-runoff model with hydraulic routing, $\mathcal{M}_{rr-hy}$, now incorporates $\boldsymbol{f}_q$, the vector of spatially distributed corrections applied to $\boldsymbol{q}$, the internal fluxes, as expressed in Equation 5.2:

$$\boldsymbol{U}(x,t) = [\boldsymbol{Q},\boldsymbol{h},\boldsymbol{q}]\,(x,t) = \mathcal{M}_{rr-hy}\left(\mathcal{D}_\Omega, \boldsymbol{\mathcal{I}}(x,t); \boldsymbol{f}_q(x,t), [\boldsymbol{\theta},\boldsymbol{h}_0]\,(x)\right) \tag{5.2}$$

A neural network-based estimator ϕ, with trainable parameters $\boldsymbol{\rho}$, is embedded into the hydrological model $\mathcal{M}_{rr-hy}$ and forms part of the complete model $\mathcal{M}$. Its purpose is to predict corrections of internal fluxes $\boldsymbol{f}_q$ as well as to estimate parameters $\boldsymbol{\theta}$ regionally, based on various input data, including atmospheric forcings $\boldsymbol{\mathcal{I}}$ and spatialized physical descriptors $\boldsymbol{\mathcal{D}}$, as described in Equation 5.3:

$$\phi : (\boldsymbol{\mathcal{I}}, \boldsymbol{h}, \boldsymbol{\mathcal{D}}; \boldsymbol{\rho}) \rightarrow (\boldsymbol{f}_q, \boldsymbol{\theta}, \boldsymbol{h}_0) \tag{5.3}$$

By construction, the complete forward model $\mathcal{M}$ is learnable, through the neural network-based mapping ϕ embedded into $\mathcal{M}_{rr-hy}$. Moreover, if $\mathcal{M}_{rr-hy}$ and ϕ are differentiable, then $\mathcal{M}$ is differentiable which is required to obtain its output gradient derivatives with respect to the neural network parameters as needed for their optimization.

The differentiable spatially distributed hydrological model studied thereafter is based on ODEs for local runoff production coupled with neural networks. This runoff is then conveyed on the spatial grid with a PDE-based hydraulic model.

5.2.1 Rainfall-runoff model with regionalizable parameters and process-parameterization neural network

The spatially distributed differentiable, learnable, regionalizable model proposed and studied in this study is detailed in this section. Its hydrological component is based on GR, that is a parsimonious, widely used and efficient lumped hydrological model (Perrin et al., 2003). Interception, production and fast and slow transfer branches structures are used at pixel scale, without unit hydrographs that are not needed for small scale pixels (refer to Chapter 3 and references therein) and could be, if needed for example for a semi distributed model with larger sub-catchments, replaced by a nash cascade that is differentiable and quasi equivalent (cf. Santos et al. (2018)). Moreover, the selected hydrological operators simply consist in algebraic relations obtained from analytical integration in time of first order ODEs that describe state evolution in interception, production and transfer reservoirs plus closure laws. This simple hydrological model produces "runoff" at pixel scale

that is then routed on the spatial grid with a kinematic wave model (Te Chow et al., 1988). These model operators and their numerical implementation are fully differentiable, a property further detailed in the following sections. Additionally, this dynamical hydrological model, implemented as recurrence relations, maintains a spatio-temporal memory via the reservoir states $\boldsymbol{h}(x,t)$. This property is used for defining the hybridization for internal flux corrections via a simple ANN based on the principle of parsimony. The ANN uses previous reservoir states and atmospheric forcings as inputs, which requires no additional information beyond the original model. The proposed model, which is spatially distributed and differentiable, is schematized in Figure 5.1. The following sections describe the complete forward model, including details on the neural networks used for flux corrections and parameter regionalization.

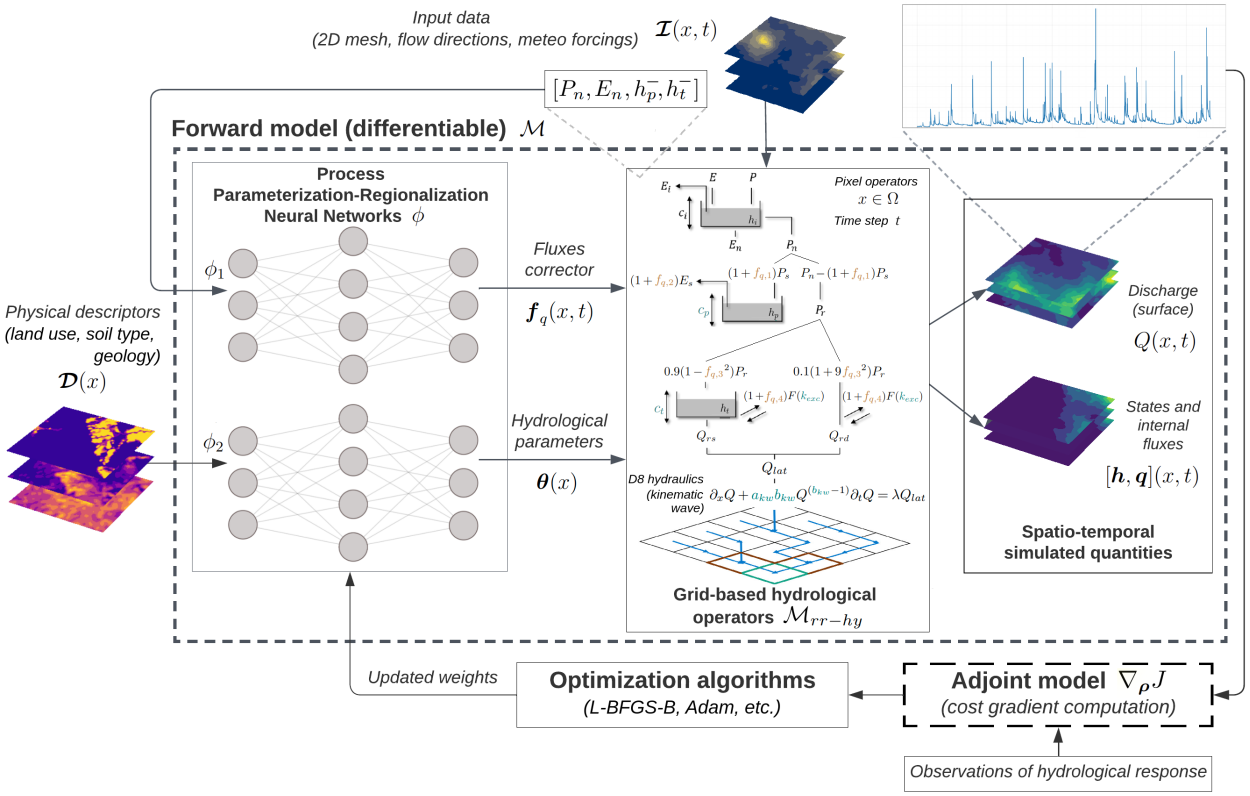


Figure 5.1 – Hybrid physics-AI framework, applied to the spatially distributed GR-like and kinematic wave model, involving a pair of process-parameterization and regionalization neural networks. The pair of neural networks is used to (i) correct internal fluxes (using neutralized atmospheric data) and (ii) estimate the model parameters (using physical descriptors), with their weights optimized through high-dimensional optimization algorithms using an adjoint model to obtain accurate gradients of the cost function.

For a given cell $x \in \Omega$ and time step $t > 0$, $P(x,t)$ and $E(x,t)$ represent the local precipitation and potential evapotranspiration. For simplicity, spatio-temporal dependencies are omitted, with flux corrections highlighted in brown and regionalized parameters in turquoise.

Interception. First, an interception reservoir of capacity c_i, automatically computed with flux matching technique (cf. Ficchì et al. (2019)), enables computing the neutralized rainfall P_n and the neutralized evapotranspiration E_n.

Production. Then, the infiltration flux $\tilde{P}_s$ into the production reservoir is obtained by applying a correction term $f_{q,1}$, predicted by the neural network ϕ_1, to the classical GR infiltration flux P_s

as follows:

$$\tilde{P}_s = \min(P_n, (1 + f_{q,1})P_s) \text{ with } P_s = c_p \left(1 - \left(\frac{h_p^-}{c_p}\right)^2\right) \frac{\tanh\left(\frac{P_n}{c_p}\right)}{1 + \left(\frac{h_p^-}{c_p}\right)\tanh\left(\frac{P_n}{c_p}\right)} \tag{5.4}$$

where c_p represents the capacity of the production reservoir predicted by the neural network ϕ_2, and h_p^- is the production state at previous time step. The actual evapotranspiration flux $\tilde{E}_s$ substracted to the production reservoir is obtained by applying a correction term $f_{q,2}$, predicted by ϕ_1, to the classical GR evapotranspiration flux E_s as follows:

$$\tilde{E}_s = \min(E_n, (1 + f_{q,2})E_s) \text{ with } E_s = h_p^- \left(2 - \frac{h_p^-}{c_p}\right) \frac{\tanh\left(\frac{E_n}{c_p}\right)}{1 + \left(1 - \frac{h_p^-}{c_p}\right)\tanh\left(\frac{E_n}{c_p}\right)} \tag{5.5}$$

Transfer (subgrid). A subgrid transfer with slow and fast "lateral" flow components is fed by a learnable partition of net rainfall using a correction term $f_{q,3}$, predicted by ϕ_1, where the net rainfall $P_r = P_n - \tilde{P}_s$ is split into $\tilde{Q}_{dl}$ and $\tilde{Q}_{dr}$, representing the fluxes that feed the delayed and direct transfer branches:

$$\tilde{Q}_{dl} = 0.9(1 - {f_{q,3}}^2)P_r \text{ and } \tilde{Q}_{dr} = 0.1(1 + 9{f_{q,3}}^2)P_r \tag{5.6}$$

A non-conservative exchange flux $\tilde{F}$, applied to both the transfer reservoir and the fast transfer branch (direct runoff), is obtained by applying a correction term $f_{q,4}$, predicted by ϕ_1, to the classical GR exchange flux F:

$$\tilde{F} = (1 + f_{q,4})F \text{ with } F = k_{exc} \times \left(\frac{h_t^-}{c_t}\right)^{7/2} \tag{5.7}$$

where k_{exc} and c_t represent respectively the exchange coefficient and the capacity of the transfer reservoir predicted by ϕ_2, and h_t^- the transfer reservoir state at previous time step. The outflow subtracted to the transfer reservoir is:

$$Q_{rs}(t) = h_t^- - \left((h_t^-)^{-4} + {c_t}^{-4}\right)^{-1/4} \tag{5.8}$$

The remaining net rainfall flux $\tilde{Q}_{dr}$ feeds the direct transfer branch where the exchange flux $\tilde{F}$ is also applied, and its outflow is $Q_{rd} = \tilde{Q}_{dr} + \tilde{F}$. The hydrological runoff flux produced at pixel scale is $Q_{lat} = Q_{rs} + Q_{rd}$ and is routed over a 2D mesh with a simple hydraulic routing module.

Note that, the values of $f_{q,i=1..4}$, which are the outputs of ϕ_1, are bounded between -1 and 1 (section 5.2.3). Thus, by definition, the transformation functions applied to these internal flux corrections (i.e., $1 + f_{q,1}$, $1 - f_{q,3}^2$, etc.) enable the preservation of the original conceptual model structure when the neural network output is zero, as all transformations equal 1 in this case. These terms were defined based on the specific fluxes being corrected and mathematical constraints. For example, the correction $f_{q,3}$ is squared to ensure non-negativity, and its transformations ($1-f_{q,3}^2$ and $1+9f_{q,3}^2$) are specifically designed to preserve mass conservation in the transfer branch partitioning, as $0.9(1 - f_{q,3}^2) + 0.1(1 + 9f_{q,3}^2) = 1$ for any value of $f_{q,3}$, while allowing the model to learn and adjust the partition between delayed and direct transfer branches from their default values of 0.9 and 0.1 respectively.

5.2.2 Pixel-to-pixel flow routing of runoff with a partial differential equation and regionalizable parameters

The routing module used here is based on a conceptual 1D kinematic wave model that is numerically solved with a linearized implicit numerical scheme (Te Chow et al., 1988). The discharge routing problem is classically reduced to a 1D problem by considering a "D8" drainage plan $\mathcal{D}_\Omega(x)$, obtained by terrain digital elevation model processing with the condition that a unique pixel has the highest drained area.

The kinematic wave model is a PDE obtained by simplifying the 1D Saint-Venant equations assuming that the momentum reduces to flow friction slope equal bottom slope. Using a conceptual parameterization of the momentum $A = a_{kw}Q^{b_{kw}}$, with A the flow cross sectional area, Q the discharge, a_{kw} and b_{kw} two parameters to be estimated, and injecting it into the mass equation $\partial_x Q + \partial_t A = \lambda Q_{lat}$, with Q_{lat} the lateral discharge (total runoff produced at a pixel from GR operators) and λ the conversion factor, a single-equation model is obtained. The model is discretized with a classical finite differences approach (cf. Te Chow et al. (1988)) resulting in the following expression for the discharge propagation model:

$$\partial_x Q + a_{kw} b_{kw} Q^{(b_{kw}-1)} \partial_t Q = \lambda Q_{lat} \tag{5.9}$$

5.2.3 Learnable mappings for spatialized GR-like model and kinematic wave routing

In this study, we use two multilayer perceptrons, the first one ϕ_1 for spatio-temporal corrections of the model internal fluxes $\boldsymbol{f}_q(x,t)$ and the second ϕ_2 for spatialized parameters $\boldsymbol{\theta}(x)$ regionalization as used in Chapter 3. Namely, ϕ consists of a pair of neural networks designed to ingest (i) neutralized atmospheric inputs $\boldsymbol{\mathcal{I}}_n = (P_n, E_n)$ (using the wording of GR conceptual model (Perrin et al., 2003; Santos et al., 2018)), along with the model states at previous time step $\boldsymbol{h}(x, t-1)$, for correcting spatio-temporal internal fluxes $\boldsymbol{q}$ (process-parameterization pipeline) and (ii) physical descriptors $\boldsymbol{\mathcal{D}}$ for estimating spatialized hydrological parameters $\boldsymbol{\theta}$ (regionalization pipeline), as shown in Equation 5.10.

$$\phi : \begin{cases} \boldsymbol{f}_q(x,t) & = \phi_1\left(\boldsymbol{\mathcal{I}}_n(x,t), \boldsymbol{h}(x,t-1); \boldsymbol{\rho}_1\right) \\ \boldsymbol{\theta}(x) & = \phi_2\left(\boldsymbol{\mathcal{D}}(x); \boldsymbol{\rho}_2\right) \end{cases} \tag{5.10}$$

with $\boldsymbol{\rho} = (\boldsymbol{\rho}_1, \boldsymbol{\rho}_2)$ the vector of trainable parameters, invariant to the spatial coordinate x over Ω, of the (pair of) neural network(s). Note that more advanced neural networks, such as CNN, RNN, or LSTM, can be explored in future studies. For instance, applying a CNN to the regionalization neural network ϕ_2 is possible and has been implemented into `smash`, but not investigated since it is out of scope of this study.

Here, the first neural network ϕ_1 has a single hidden layer with 16 neurons, followed by a Leaky ReLU activation function. The output layer uses a TanH activation function, which is bounded from -1 to 1. Then, the flux corrections $\boldsymbol{f}_q = (f_{q,i=1..4})^T$, predicted by ϕ_1, are applied for each pixel x and time t to correct simultaneously the internal fluxes of the GR hydrological operators as described in section 5.2.1. The second network ϕ_2 consists in 3 hidden layers with 96, 48 and 16 neurons. ReLU activation functions are used between hidden layers, while the Sigmoid function is applied in the output layer and followed by a scaling function to constrain the model parameters in accordance with their feasible bounds. The vector of conceptual spatialized parameters, mapped by ϕ_2, is $\boldsymbol{\theta} = (c_p, c_t, k_{exc}, a_{kw}, b_{kw})^T$ composed of production and transfer reservoir capacities c_p and c_t, exchange coefficient k_{exc}, kinematic wave parameters a_{kw} and b_{kw}. Finally, the parameter control vector to optimize is $\boldsymbol{\rho} = (\boldsymbol{\rho}_1, \boldsymbol{\rho}_2)$, i.e., the weight and biais of the process-parameterization and regionalization mappings.

5.2.4 Inverse problem and analysis of the hybrid physics-AI framework

The inverse problem is addressed using a multi-site calibration cost function and the VDA algorithm, incorporating the coupled gradients of the regionalization neural network Jacobian. This approach follows the methodology described in Section 3.2.2 and Section 3.2.3 in Chapter 3 (the hydrological model $\mathcal{M}_{rr}$ is replaced by the rainfall-runoff model with hydraulic routing $\mathcal{M}_{rr-hy}$).

After optimization with the proposed approach, enabling to jointly learn physical processes parameterization and regionalization, a hybrid process-based spatially distributed calibrated hydrological model $\mathcal{M}_{\hat{\rho}}$ is obtained and is therefore reusable for space-time extrapolation. Contrarily to PINNs where the physical model residual serves as a weak constrain in optimization, in our proposed conceptualization, the physics is used as a strong constrain. In this sense, the approach can be seen as a learnable spatialized physical model. Moreover, contrarily to PINNs and LSTM, which are composed of neural networks only, our hybrid model is physically interpretable through its conceptual parameters $\boldsymbol{\theta}(x)$, internal states $\boldsymbol{h}(x,t)$ and fluxes $\boldsymbol{q}(x,t)$. Moreover, the ANNs ϕ_1 and ϕ_2 coupled with the conceptual model $\mathcal{M}_{rr-hy}$ at the pixel scale for each time step, are capable of capturing non-linear and multi-resolution effects. The conceptualization, where the physics is used as a strong constrain in the forward model, enables using other differentiable hydrological and hydraulic models for example, on structured or unstructured meshes. Such an approach enables integrating data that are not directly usable nor explicitly represented in the model such as the physical descriptors for regionalization of conceptual parameters here.

5.3 Data and experimental design

We evaluated our method on two datasets (see Figure 5.2). The first dataset includes 235 non-

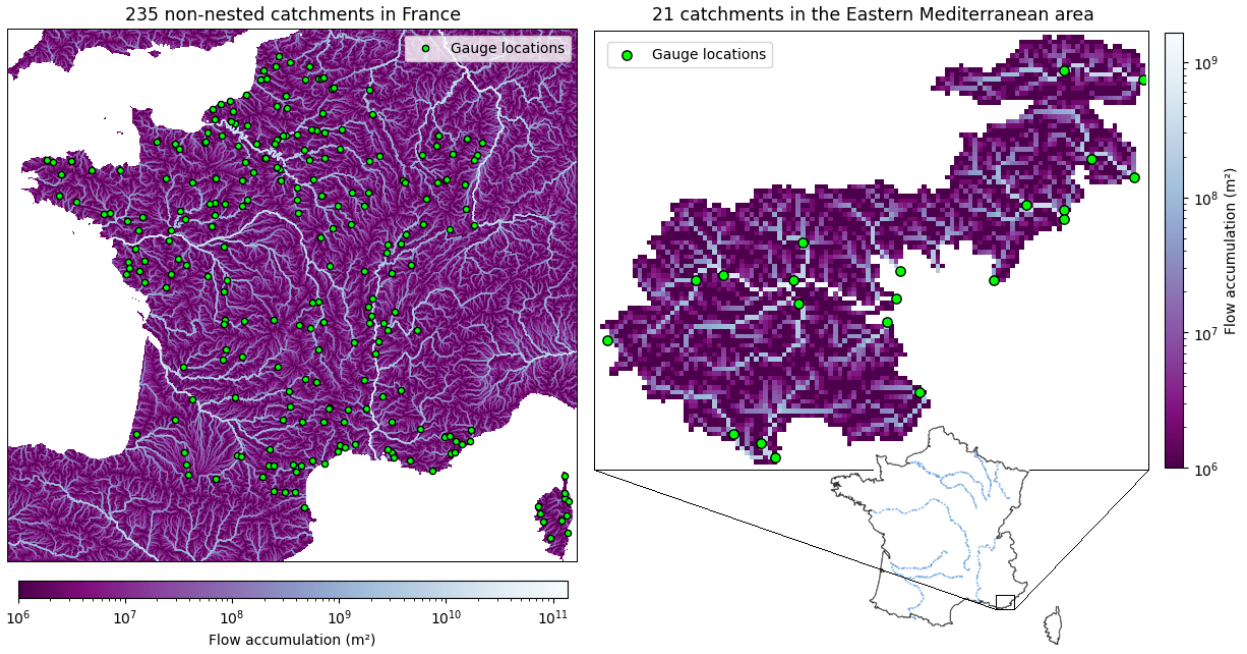


Figure 5.2 – Study areas used for evaluation. The first area consists of 235 non-nested catchments in France, while the second area includes 21 catchments in a multiple-catchment setup in the Eastern Mediterranean region, representing contrasting hydrological conditions.

nested catchments selected from Hashemi et al. (2022), which is part of a larger dataset containing

4,190 French catchments provided by the INRAE-HYCAR research unit (Delaigue et al., 2020; Brigode et al., 2020). The second dataset consists of 21 catchments, a subset of the ArcMed region, taken from Chapter 3. The `smash` model is run on a spatial grid with a resolution of $dx = 1$ km and a temporal step of $dt = 1$ h (see Section 1.4.2 in Chapter 1).

The first dataset contains hourly time series over a 13-year period (August 2006 to July 2019) for downstream gauges only. It is used to evaluate single-gauge optimization (local calibration) without regionalization, solely focusing on the process-parameterization neural network ϕ_1, which is the key novelty of this study. The 13-year period is divided into two segments: the calibration period covers the first 7 years (including a one-year warmup), and the remaining 6 years are used for temporal validation. Four methods are compared to evaluate the learning capacity of the neural network ϕ_1:

— Two classic GR models with spatially uniform parameters (GR.U) and spatially distributed parameters (GR.D), which, in some cases, exhibit under- or over-parameterization issues in the spatially distributed hydrological model;

— Two hybrid GR models that integrate the neural network ϕ_1 (called ϕ_1-hybrid) with spatially uniform parameters (GRNN.U) and spatially distributed parameters (GRNN.D).

The second dataset includes hourly time series over 7 years (August 2009 to July 2016) for both nested and independent catchments in the Eastern Mediterranean region (known as "MedEst"). This dataset is used to assess the relevance of the learnable structure for simultaneous multi-gauge regionalization with physical descriptors. A set of seven descriptors (see Table 1.1 in Chapter 1), with a spatial resolution of 0.01° in the WGS 84 projection, encompassing various types such as topography, morphology, land use, and hydrogeology, is used as inputs for the regionalization mapping ϕ_2 (Figure 5.3).

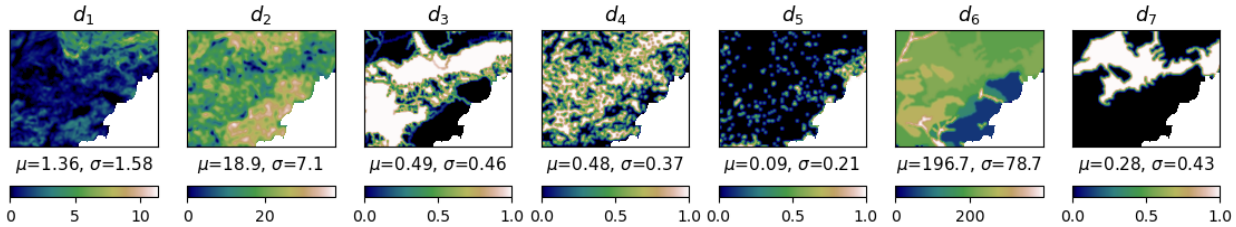


Figure 5.3 – Maps of seven physical descriptors in the MedEst area at a resolution of 0.01° in the WGS 84 projection, where μ and σ represent the spatial average and standard deviation for each descriptor.

The MedEst region presents a challenging case due to its contrasting hydrological properties, including steep topography and highly heterogeneous soils and bedrock (e.g., Garambois, Roux, Larnier, Labat, et al., 2015b). This region is prone to intense rainfall events that trigger non-linear flash flood responses and contains a significant proportion of karstic zones. The first 4-year time series, including a one-year warmup period, is used for calibration, while the remaining 3 years are used for validation. Four methods are compared to evaluate the learning capacity of both the process-parameterization neural network ϕ_1 and the regionalization neural network ϕ_2:

— The classic GR model with regional, spatially uniform parameters (GR.U);

— The hybrid GR model integrating the neural network ϕ_1 (ϕ_1-hybrid) to correct internal fluxes in hydrological processes, with regional, spatially uniform parameters (GRNN.U);

— The classic GR model integrating the neural network ϕ_2 (ϕ_2-hybrid) to learn the mapping between physical descriptors and spatially distributed hydrological parameters (GR.NN);

— The hybrid GR model integrating both neural networks ϕ_1 and ϕ_2 (GRNN.NN), representing the fully integrated hybrid approach (ϕ-hybrid) among the studied methods.

We recall that the design of the architecture and the hyper-parameters of both neural networks were determined as described in section 5.2.3.

5.4 Results and discussion

In this section, we first present the performance of the hybrid spatially distributed models tested on both datasets. Then, we will provide further interpretation and discussion of the learning process to analyze the proposed framework and enhance the understanding of hydrological behaviors in the process-based model through internal fluxes.

5.4.1 Model performance analysis

5.4.1.1 Local calibration over 235 French catchments

Figure 5.4 illustrates typical simulated streamflows from different methods for small, medium, and large catchments. Overall, we observe significant improvements in the hybrid methods compared to the classic models in simulating both peak flows and low flows. For example, in the case of the medium catchment, GRNN.U more accurately predicts the peak flows in January 2014 compared to GR.U, while also reliably reproducing the low flows, particularly during the period between 2018 and 2019.

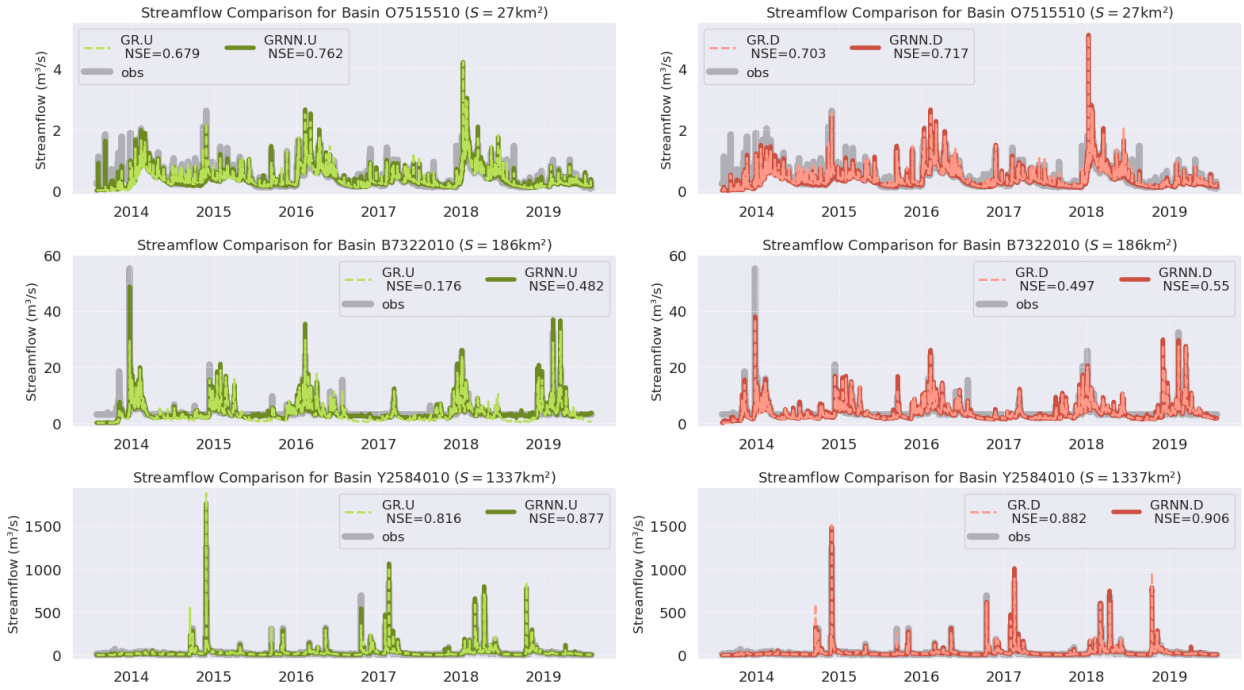


Figure 5.4 – Comparison of streamflow simulation across representative small, medium, and large catchments randomly selected among the 235 catchments in France during the validation period.

Figure 5.5 shows a global comparison of performances in terms of Nash–Sutcliffe efficiency (NSE), Kling–Gupta efficiency (KGE), and root mean squared error (RMSE) across both calibration and validation periods for different methods. The results suggest that ϕ_1-hybrid methods (GRNN.U and GRNN.D) consistently achieve superior efficiency scores and lower errors compared to the classic models (GR.U and GR.D) in various scenarios. In calibration, both hybrid models outperform the

classic ones, with significantly higher median NSE scores (0.85 and 0.86 compared to 0.79 and 0.83), a narrower and higher interquartile range, and a shorter lower whisker.

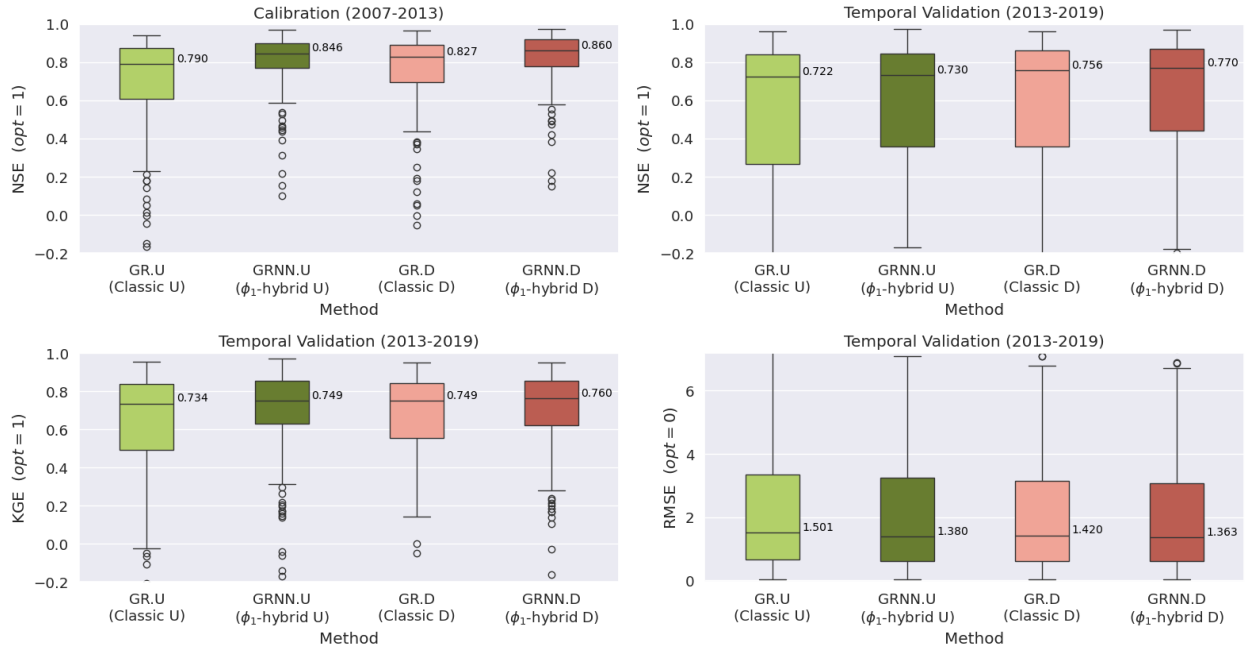


Figure 5.5 – Model performance comparison of local calibration methods across 235 catchments in France. The two boxes in lighter colors represent the classical GR models (GR.U and GR.D), while the two boxes in darker colors represent the ϕ_1-hybrid models (GRNN.U and GRNN.D). The evaluation is based on simulated discharges over the calibration and validation period, using NSE, KGE, and RMSE metrics.

In temporal validation, GRNN.U achieves a median NSE of 0.73 compared to 0.76 for GR.D, and both models reach a median KGE of 0.75, while GRNN.U shows a lower median RMSE of 1.38 compared to 1.42 for GR.D. Although median improvements may appear small, it is important to consider the entire distribution. In addition to the median values, we observe notable enhancements in other statistical measures, such as the interquartile range (0.25 and 0.75 quantiles) and whiskers in the boxplots. For catchments that already exhibit satisfactory performance, the effect of hybridization is relatively small, leading to similar median, 0.75 and 0.95 quantile values. However, for poorly performing basins, the hybrid models provide substantial improvements, as evidenced by enhanced performance in the lower quartiles. Notably, the hybrid model with spatially uniform hydrological parameters (GRNN.U) performs comparably to, and in some cases surpasses, the classic GR model with spatially distributed parameters (GR.D). This result is promising, as it demonstrates that while the original model with spatially uniform conceptual parameters (GR.U) inherently leads to under-parameterization compared to GR.D, this limitation can be compensated for by the spatially distributed flux correction in GRNN.U.

To evaluate model performance in terms of flash floods, an automatic segmentation algorithm has been introduced in Chapter 2 to compute several flood event signatures. These signatures help depict the model behavior during flood events. Relative error is used as the evaluation metric to quantify the difference between simulated and observed flood event signatures, including peak flow, runoff coefficient, flood flow, and baseflow. Figure 5.6 shows the cumulative distribution function (CDF) of the relative error for these signatures, based on over 2,700 flood events that occurred during the 6-year validation period. The hybrid models achieve the best performance, outperforming the

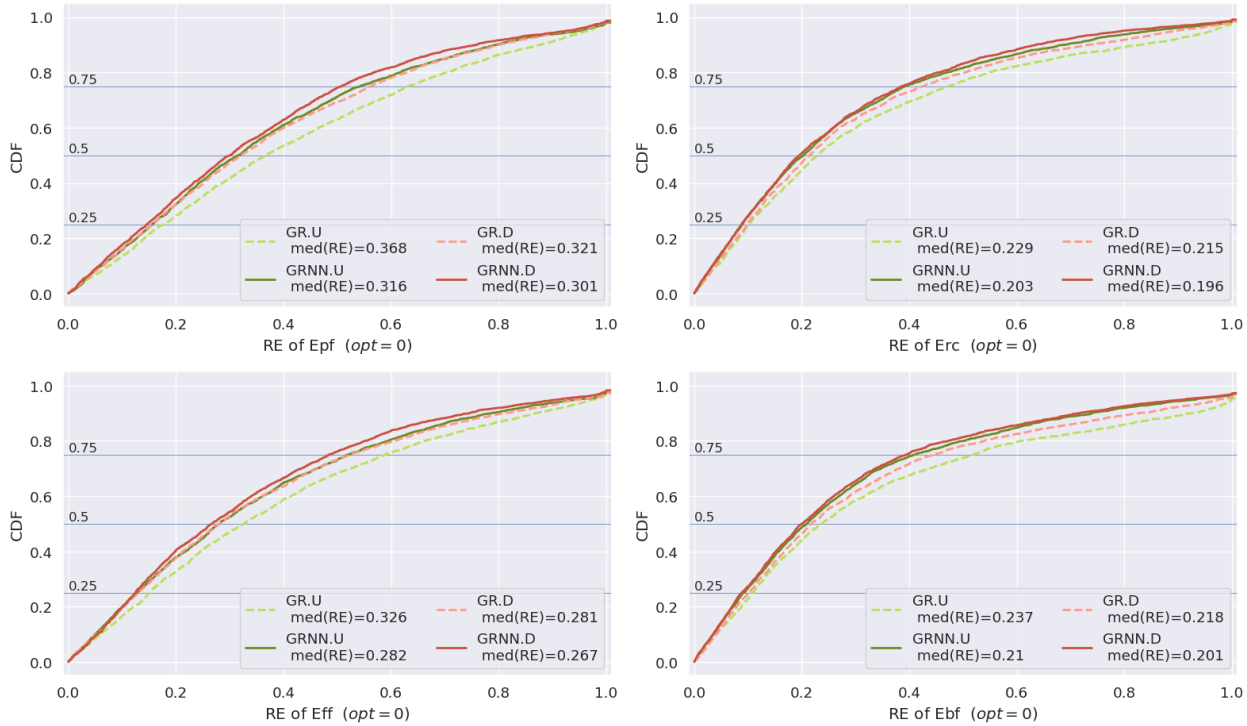


Figure 5.6 – Comparison of model performance in simulating flood event signatures, presented as the cumulative distribution function (CDF) of the relative error (RE) between observed and simulated values for peak flow (Epf), runoff coefficient (Erc), flood flow (Eff), and baseflow (Ebf). The evaluation is based on 2,718 flood events across 235 catchments during the validation period (08/2013–07/2019).

classic GR.U model, with their CDF lines consistently above. Notably, the hybrid model GRNN.U, using only spatially uniform parameters, attains performance comparable to or even better than the classic model with spatially distributed parameters (GR.D). GRNN.U shows similar performance to GR.D in reproducing peak flows (with the same median error of 0.32) and flood flow (both with a median error of 0.28), while performing slightly better in reproducing the runoff coefficient (0.20 compared to 0.22) and baseflow (0.21 compared to 0.22). This highlights the strength of the hybrid process-parameterization framework, particularly its relevance in improving flood modeling systems.

5.4.1.2 Multi-catchment regionalization over a flash-flood prone Mediterranean area

Here, we investigate how the hybrid models perform in the study of a multi-catchment regionalization setup. In calibration, it is evident from Figure 5.7 that the ϕ_2-hybrid model (GR.NN) and the ϕ-hybrid (GRNN.NN) model, both using the regionalization neural network ϕ_2, outperform the models with lumped parameters (GR.U and GRNN.U). Notably, the fully integrated hybrid model GRNN.NN dominates the radar plot, with a large shape extending toward the outer edges, fully enveloping the other methods. Although GRNN.U clearly falls short of the two regionalization-based models, it still shows a significant improvement over the classic GR.U model. Similar results can be observed in both temporal and spatio-temporal validation, as seen in the boxplots. This proves that using physical descriptors with a learnable mapping is an effective approach in this regionalization setup (multi-catchment in a large, flash flood-prone area with high spatio-temporal resolution data), compared to lumped models (without physical descriptors) or simpler regionalization methods (e.g.,

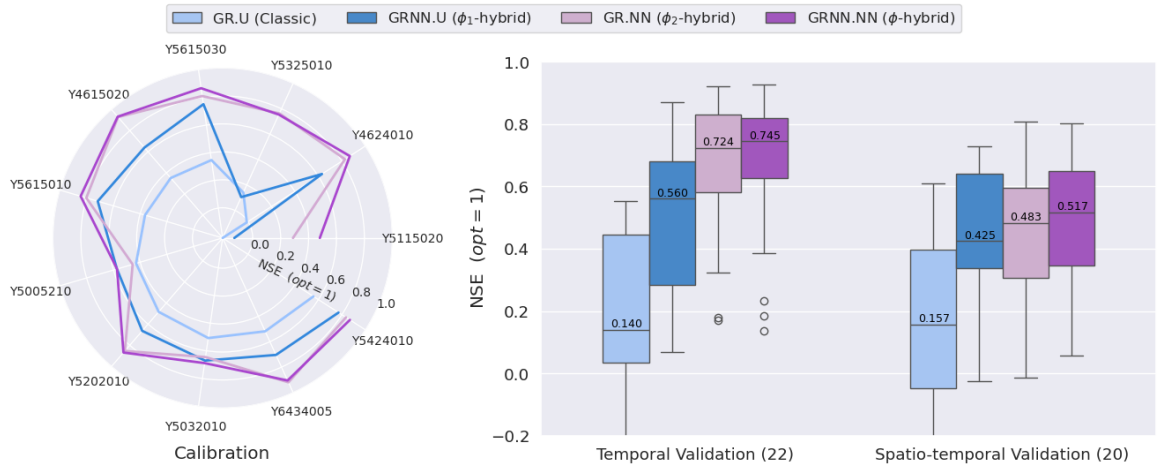


Figure 5.7 – Comparison of multi-catchment regionalization performance for different methods. The evaluation is based on NSE scores computed over three periods: the 3-year calibration period (excluding a 1-year warmup), the first 18 months, and the last 18 months of the 3-year validation period for 11 calibration catchments and 10 validation catchments. The numbers in parentheses on the boxplots indicate the total number of samples evaluated, with the validation period split into two 18-month periods (sample counts are doubled for all catchments).

multi-linear or multi-polynomial mappings) as demonstrated in Chapter 3. Interestingly, while the ϕ-hybrid model GRNN.NN, which delivers the best overall performance, shows a moderate gain over GR.NN—with median NSE scores of 0.75 (compared to 0.72) and 0.51 (compared to 0.48) in temporal and spatio-temporal validation—the ϕ_1-hybrid model GRNN.U, that uses lumped parameters without regionalization using physical descriptors, makes a dramatic improvement over the classic GR.U model (median NSE of 0.56 compared to 0.14, and 0.43 compared to 0.16). In this way, learning internal flux corrections has made it possible to improve the regionalizability of a distributed conceptual hydrological model even with spatially uniform conceptual parameters (i.e., without using physical descriptors). This may represent a compelling research direction for reducing structural uncertainty in modeling, using a minimum of data and enabling more efficient extraction of multi-scale information through hybrid flux correction using GRNN.U, with potential for flexible semi-spatializations of conceptual parameters and even proximity-based regionalizations for spatially dense gauging networks (cf. Oudin, Andréassian, et al. (2008)).

In the context of flood prediction, the hybrid models (GRNN.U, GR.NN, and GRNN.NN) consistently yield superior performance compared to the classic GR.U model. Figure 5.8a presents the RMSE and NSE metrics computed using short time series from 143 flood events that occurred across the entire MedEst area during the validation period from August 2013 to July 2016. Figure 5.8b shows typical streamflow simulations, demonstrating a significant enhancement for both hybrid process-parameterization and regionalization-based approaches, compared to classical methods, in simulating high flow characteristics and behaviors during flood events.

Although the NSE values computed for the 143 flood events are relatively low across all models, it is important to note that NSE for flood events—which are short time series with high values—is highly sensitive to small timing errors. Even slight discrepancies in peak timing can lead to substantial decreases in NSE. Additionally, data and modeling uncertainties may vary between events, making the accurate prediction of highly contrasted events particularly challenging. The models are calibrated on the entire time series, and we evaluate the validation results specifically for flood events, where classical approaches often struggle to accurately estimate water dynamics. This dis-

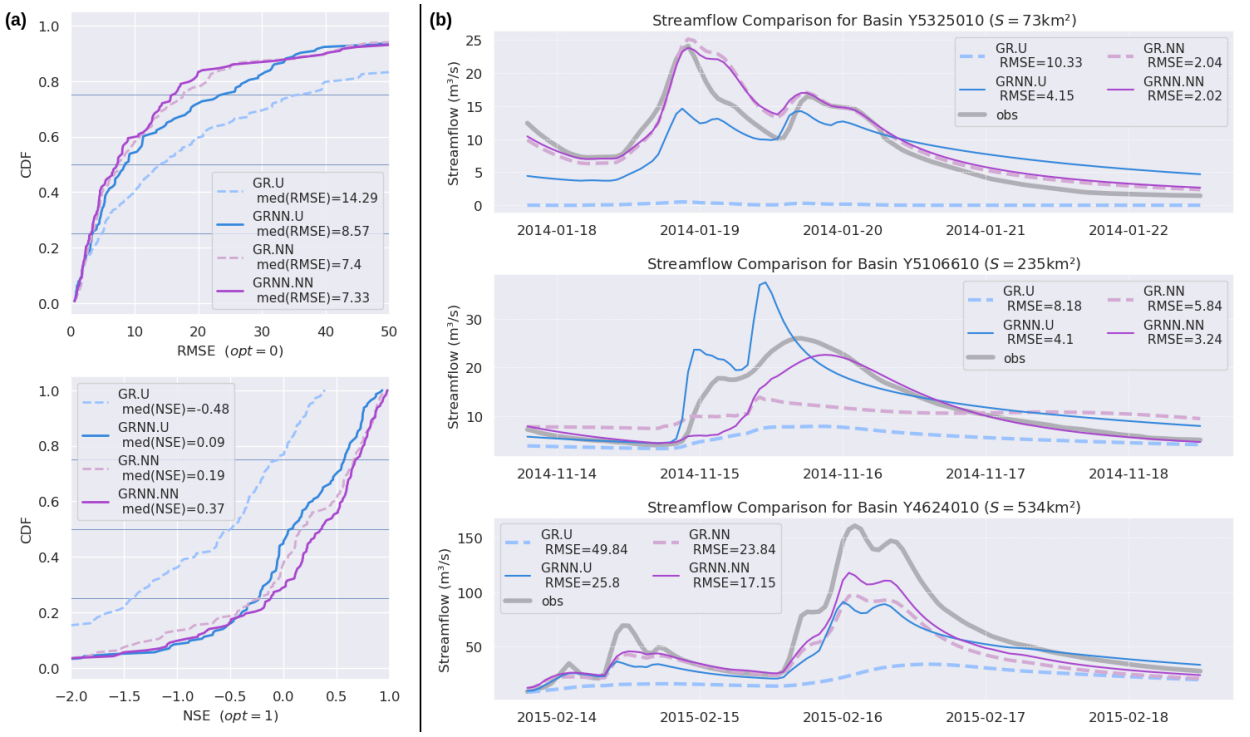


Figure 5.8 – (a) Comparison of NSE and RMSE metrics computed for 143 flood events detected in the validation period across 21 catchments in the MedEst region. (b) Comparison of streamflow simulations for representative small, medium, and large catchments during several flood events selected in the validation period.

crepancy highlights the difficulty in capturing the rapid and intense nature of flood events, even with advanced hybrid models. This underscores the need to investigate potential sources of error, including input data quality and model structural limitations, as well as the impact of using a calibration metric based solely on flood events. These factors could explain the overall challenges in flood event simulation.

5.4.2 Towards learning hydrological behaviors

Here, we focus on uncovering the hydrological behaviors inferred with the hybrid approach consisting in neural networks embedded into a physical model for learnable correction of internal fluxes. In the studied hybrid structure GRNN, the learned correction of GR-like model consists in 4 flux corrections $\boldsymbol{f}_q(x,t) = (f_{q,i}(x,t))_{i=1..4} = \phi_1(P_n, E_n, h_p, h_t)(x,t)$ for each pixel and time step of the simulation domain from atmospheric forcings and previous model states. A positive (or negative) correction of $f_{q,i}$ (where $i = 1, 2, 4$), with values bounded in $]-1;1[$ due to the TanH activation function used in the output layer, results in an increase (or decrease) in the original fluxes P_s, E_s, and $|F|$—the absolute value of F (cf. Equations 5.4, 5.5, 5.7), thereby influencing the simulated mass balance. Meanwhile, $f_{q,3}^2$, with values in $[0;1[$, produces a conservative re-repartition of net rainfall P_r between direct and delayed transfer branches (cf. Equation 5.6), thereby affecting the subgrid transfer dynamics. The following quantitative analysis begins with the spatio-temporal averages of these flux corrections and proceeds to explore their variability across the 235 independently calibrated catchments, as well as in the regionalization test case.

5.4.2.1 Analysis of internal flux corrections

Figure 5.9 shows the maps of spatio-temporal average flux corrections (the corrections are first averaged spatially within each catchment, then temporally across the calibration period), obtained through local calibrations using GRNN.U. Red and blue indicate positive and negative flux cor-

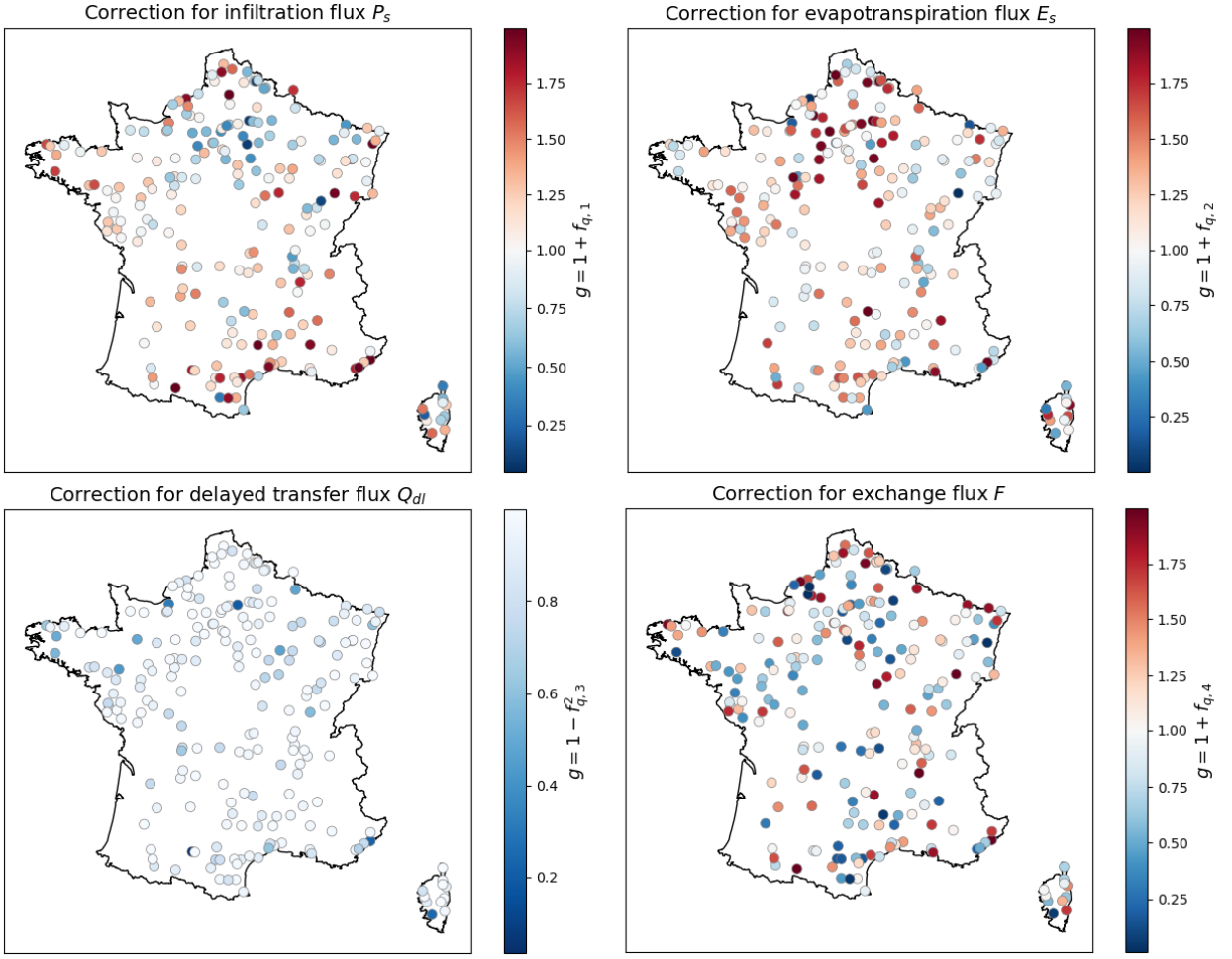


Figure 5.9 – Maps of spatio-temporal average flux corrections $\overline{\overline{g\left(\boldsymbol{f}_q\right)}^{x\in\Omega_j}}^{t}$, where $j = 1..N_g$, for the $N_g = 235$ catchments, obtained through local calibrations of spatially uniform parameters with the hybrid model structure (GRNN.U). The function $g(.)$ represents the transformation applied to the neural network output $\boldsymbol{f}_q$, which may differ depending on the specific flux being corrected. Red indicates corrections that tend to increase the current flux, while blue indicates corrections that reduce it, with white representing minimal or no effective correction.

rections in the spatio-temporal average, respectively. For fluxes affecting the production reservoir, namely infiltrating rainfall P_s and evapotranspiration E_s, the average corrections show opposite signs for the majority of basins and the same sign for a minority. These maps reveal different trends of flux corrections across France. Several regions exhibit strong corrections (either negative or positive) for P_s and E_s, while others show near-zero corrections (white points with values close to 1). However, the exchange flux is generally the most influenced by the corrections, as indicated by the dark colors across the maps, playing a crucial role in refining the model's state dynamic. Note that the transformation function $1 - f_{q,3}^2$ applied to correct the delayed transfer flux Q_{dl} result in reduction of this flux and conservative augmentation of Q_{dr} flux feeding the direct branch.

Now, we turn to the analysis of time series of spatially averaged flux corrections presented in Figure 5.10. Figure 5.10a illustrates these corrections over time for three randomly selected catchments that are representative of the 235 French catchments. We observe opposite signs in

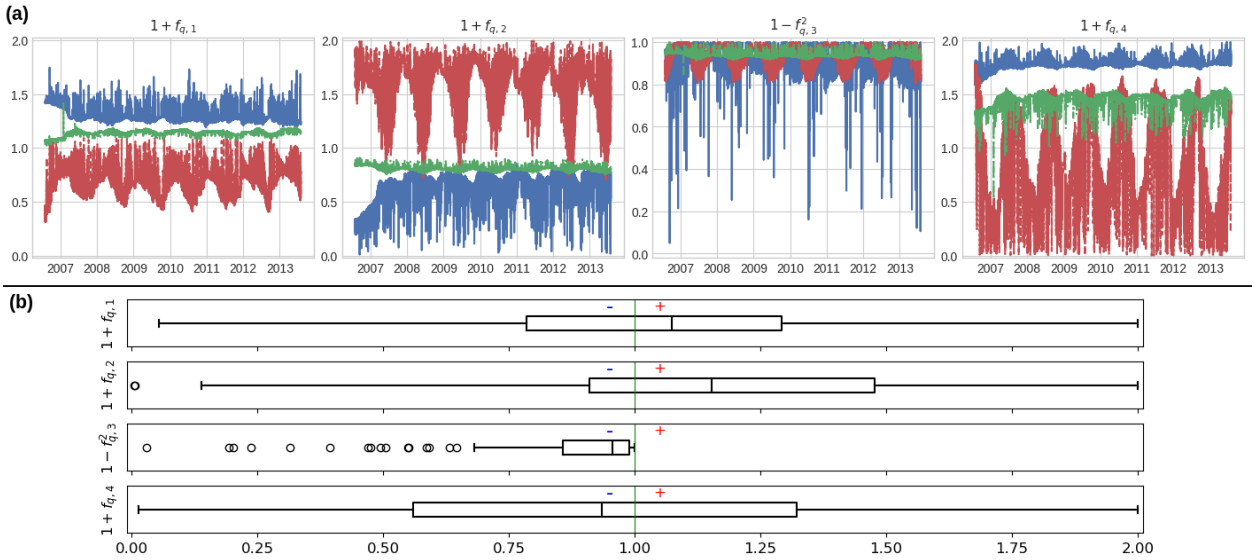


Figure 5.10 – (a) Spatially averaged flux correction time series $\overline{g\left(\boldsymbol{f}_q(t)\right)}^{x\in\Omega_j}$ for three randomly selected catchments from the set of 235. These are obtained through local calibrations of spatially uniform parameters with the hybrid model structure (GRNN.U). (b) Boxplot of spatio-temporal average flux corrections $\overline{\overline{g\left(\boldsymbol{f}_q\right)}^{x\in\Omega_j}}^{t}$, where each boxplot represents 235 catchment-specific spatio-temporal averages.

the corrections $f_{q,1}$ and $f_{q,2}$ for infiltration P_s and evapotranspiration E_s from the production reservoir. Catchments exhibiting positive corrections to P_s and negative corrections to E_s suggest that more water is directed towards the production reservoir and less is lost by evapotranspiration, leading to increased moisture state. Conversely, in catchments with negative corrections for P_s and positive corrections for E_s, reduced infiltration and increased evapotranspiration imply lower moisture states. Furthermore, periodic behaviors are observed over time for corrections of the four water fluxes, highlighting the temporal patterns of flux corrections. This pattern likely reflects the footprint of the annual periodicity of the production state h_p, which is an input to the neural network ϕ_1. Overall, the corrected infiltrating rainfall $\tilde{P}_s$ is generally 10% higher than the original, as indicated by the median of $1 + f_{q,1}$ being approximately 1.10 in Figure 5.10b. This implies an increased water level in the production reservoir, and hence more water being directed there rather than feeding the transfer branch. This observation somehow explains why the production capacity c_p, calibrated for the hybrid models, is generally slightly higher than that of the classical models (see Table 5.1). Additionally, fewer corrections are obtained for re-repartition of net rainfall flux into

Table 5.1 – Median (mean; standard deviation) of the calibrated hydrological parameters across the study area of 235 catchments.

Method	c_p	c_t	k_{exc}	a_{kw}	b_{kw}
GR.U	281.2 (384.5; 283.8)	155.4 (388.1; 758.6)	-0.58 (-2.15; 4.73)	3.41 (6.5; 9.92)	0.82 (0.77; 0.25)
GRNN.U	344.8 (469.2; 398.9)	150.7 (450.2; 1135)	-0.64 (-1.75; 5.37)	4.8 (9.21; 12)	0.82 (0.73; 0.29)
GR.D	271.4 (379; 299.6)	151.5 (461.5; 984.5)	-0.62 (-1.76; 3.67)	4.6 (6.11; 6.46)	0.74 (0.7; 0.19)
GRNN.D	301.3 (411.1; 309)	184.2 (524.8; 991.7)	-0.63 (-1.37; 3.4)	4.62 (6.42; 6.9)	0.74 (0.71; 0.19)

the direct and delayed transfer branches (i.e., the corrections show less variation and are closer to 1). Negative corrections that tend to reduce the flux magnitude are applied to the delayed transfer branch $\tilde{Q}_{dl}$, which implies positive corrections for the direct transfer branch $\tilde{Q}_{dr}$. In this case, the hybrid model suggests that more water reaches the outflow Q_{rd} via the direct transfer branch. Both transfer branches are affected by the exchange flux $\tilde{F}$ for which, across most catchments, a reduction is obtained by flux correction (the median of $1 + f_{q,4}$ is lower than 1).

In multi-gauge regionalization setup, distinct spatio-temporal patterns emerge over the MedEst area, as shown in Figure 5.11 for GRNN.U. Figure 5.11a illustrates that the spatially averaged corrections for infiltrating rainfall P_s show relatively high temporal variability; moreover, they still exhibit stable periodic patterns after the first-year warmup. The spatial average of the corrected

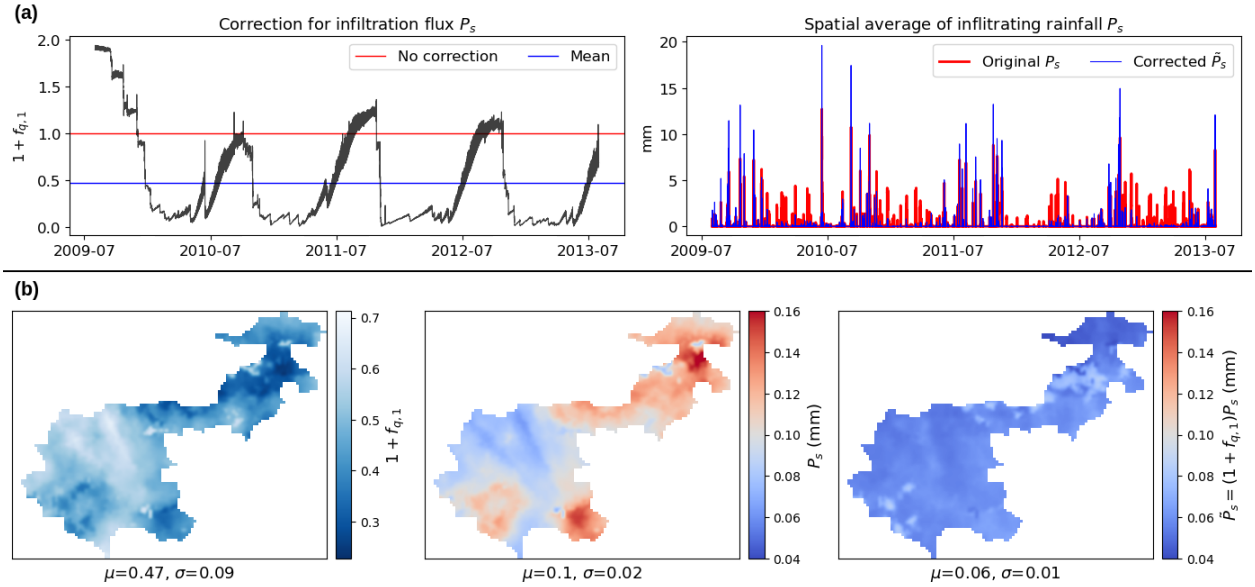


Figure 5.11 – Visualization of flux corrections in the MedEst region obtained through regional calibration of spatially uniform parameters with the hybrid model (GRNN.U): (a) Spatial average of infiltrating flux correction $\overline{1 + f_{q,1}(t)}^{x}$, original and corrected infiltrating rainfall $\overline{P_s(t)}^{x}$, $\overline{\tilde{P}_s(t)}^{x}$; (b) Maps of time-averaged infiltrating flux correction $\overline{1 + f_{q,1}(x)}^{t}$, original and corrected infiltrating rainfall $\overline{P_s(x)}^{t}$, $\overline{\tilde{P}_s(x)}^{t}$, where μ and σ represent the spatial average and standard deviation.

flux $\tilde{P}_s$ tends to be lower during moderate-rain events, while it is higher during high-rain events compared to the original flux P_s. This suggests that the hybrid model directs more rainfall into the transfer branch during moderate-rain events (which may have longer duration), while the opposite behavior is observed for high-rain events (which can be shorter in duration). The spatial maps of time-averaged flux corrections in Figure 5.11b further indicate that the hybrid model generally applies negative corrections, reducing the spatio-temporal mean of infiltrating rainfall from 0.1 mm to 0.06 mm. Interestingly, these maps also reveal spatial variability in internal flux corrections, which may explain the improved regionalizability of the hybrid GRNN models, as demonstrated by its performance in spatio-temporal validation, even with spatially uniform conceptual parameters (without regionalization using physical descriptors).

5.4.2.2 Hybridization effect on main mass fluxes involved in basin's water balance

This section examines the effect of ϕ_1-hybridization on the primary mass fluxes involved in the hydrological mass balance, as simulated using the original GR-like spatially distributed model

structure. For a given catchment domain Ω, the annual catchment-scale flux $\Psi_{f,\mathcal{A}}$ of a state-flux $f(x,t)$—such as actual evapotranspiration, exchange flux, or runoff flux—simulated using either the classical model or the hybrid model (with flux corrections omitted for brevity) is computed as follows:

$$\Psi_{f,\mathcal{A}} = \frac{1}{|\Omega|} \int_{t\in\mathcal{A}} \int_{x\in\Omega} f(x,t) dx dt \tag{5.11}$$

where $\mathcal{A}$ denotes the annual period, $|\Omega|$ represents the drainage area.

A basin scale analysis is performed for each of the 235 French basins simulated, focusing on the flux of rainfall P inflowing the model and three key fluxes affected by the ϕ_1-hybridization: evapotranspiration E_s from production store, exchange F, and pixel-scale discharge Q_{lat} prior to routing. The annual average of each flux is calculated using Equation 5.11, and the interannual averages of these water gain or loss fluxes over the 6-year calibration period (2007–2013) are shown in Figure 5.12. This figure quantitatively illustrates the impact of ϕ_1-hybridization on the classical

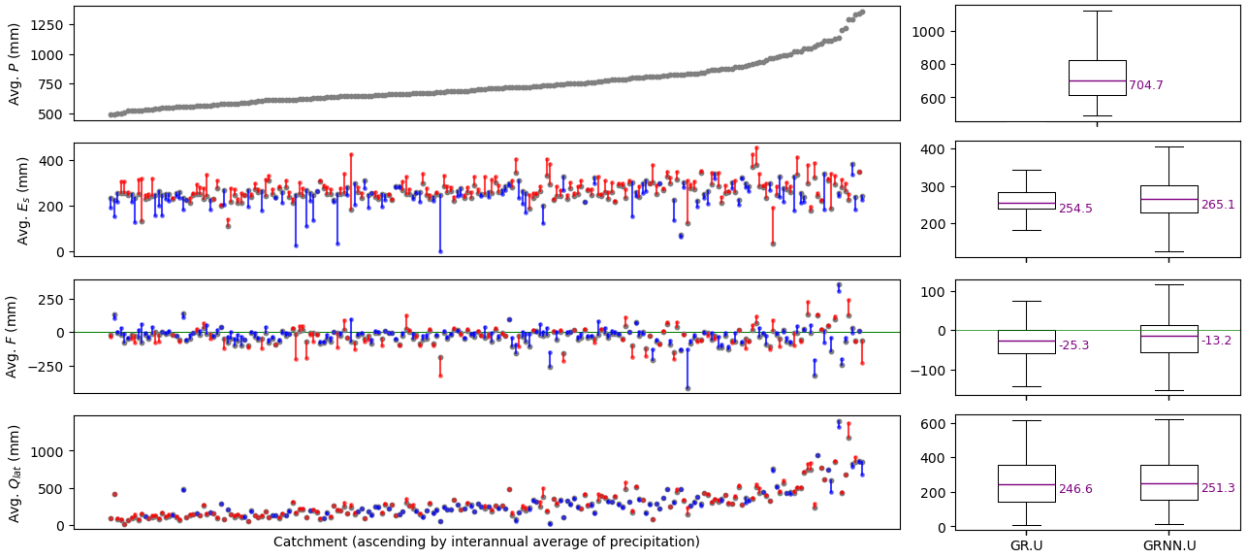


Figure 5.12 – Comparison of mass fluxes affecting water balance in interannual mean at basin scale over the 235 catchment set, for the classical GR model (GR.U) and the ϕ_1-hybrid model (GRNN.U) using uniform mapping "U" for conceptual parameters $\boldsymbol{\theta}$, in local calibration. The x-axis shows 235 catchments, sorted by their average precipitation. Grey dots represent the interannual averages of precipitation P, actual evapotranspiration E_s, exchange flux F, and lateral discharge Q_{lat} over the calibration period for the classical model. Red dots and lines represent increases according to the hybrid model, while blue ones indicate decreases. For cases where $F < 0$, red indicates a larger magnitude in F for the hybrid model (more negative), while blue indicates a lesser magnitude (closer to zero).

GR-like model, with uniform conceptual parameters for each basin and each model structure. Over the variety of hydrological behaviors and annual rainfall regimes of this large catchment set, it is noteworthy that hybridization results show almost no change for nearly all basins in terms of interannual discharge runoff volume, with a median of 246.6 mm for GR.U and 251.3 mm for GRNN.U and similar quantiles, while dynamic changes have been obtained as suggested by improved NSE, flood signatures, and hydrographs (cf. performance analysis in section 5.4.1), as well as internal flux corrections (such as infiltration and repartition between direct and delayed lateral transfer branches).

The simulated water balance is influenced by the correction of k_{exc}, which represents the exchange flux and can result in either gains or losses relative to the original model (which is already non-conservative). In this case, the exchange flux is moderately affected by hybridization, with a median trend of reduced exchange from -23.5 mm to -13.2 mm. This reduction is compensated, in terms of water balance, by an increase in evaporation from the production reservoir (from 254.5 mm to 265.1 mm in median). Notably, both fluxes exhibit a larger interquartile range across basins compared to the classic model structure. Therefore, the proposed ϕ_1-hybridization enables the learning of spatio-temporal corrections of internal model dynamics, resulting in physically interpretable fluxes that remain within imposed ranges and lead to overall model improvement.

Figure 5.13 depicts the versatile nature of the learnable hybrid model in comparison to classical conceptual models for correcting internal fluxes and vividly illustrates the learned non-linear relationship between the corrected net rainfall and neutralized data, as well as internal states. The

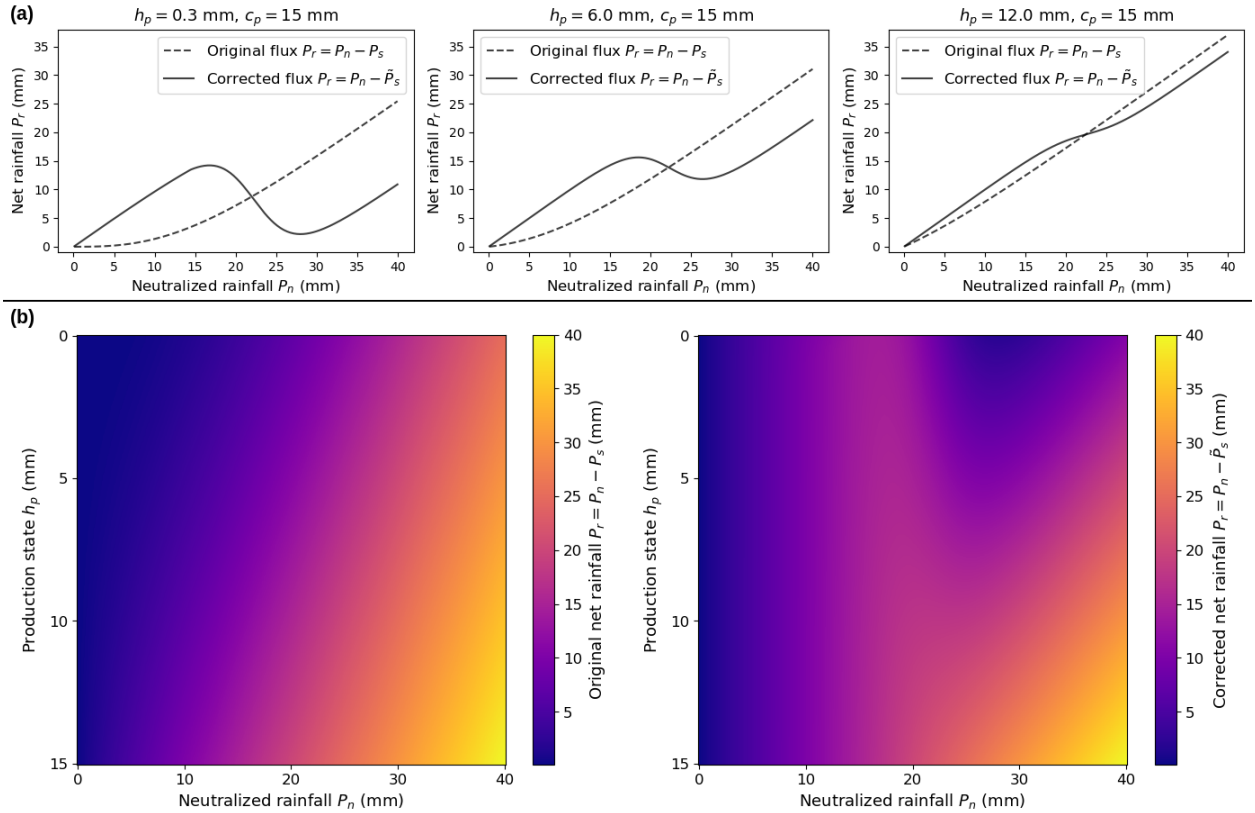


Figure 5.13 – Hydrological interpretation of the non-linear response surface obtained using the learned flux correction neural network for infiltrating rainfall $\tilde{P}_s = (1 + f_{q,1})P_s$, plotted with a production capacity of $c_p = 15$ mm and neutralized evapotranspiration of $E_n = 0$ mm: (a) Original and corrected net rainfall P_r for different levels of the production state h_p; (b) Response surface of original and corrected net rainfall P_r as a function of both the production state h_p and neutralized rainfall P_n.

model response surface of the net rainfall $P_r = P_n - \tilde{P}_s$, obtained with the corrected infiltrating rainfall $\tilde{P}_s = (1 + f_{q,1})P_s$, is shown for different levels of the production state h_p and neutralized rainfall P_n. Interestingly, this corrected net rainfall $P_n - \tilde{P}_s$, regardless of the level of production state (i.e., $h_p = 0.3, 6$, and 15 mm), exhibits a non-monotonic behavior with respect to the intensity of neutralized rainfall P_n (Figure 5.13a). However, this non-monotonic behavior becomes less pronounced as the production state h_p approaches the production capacity c_p. Figure 5.13b

further clarifies the non-linear response surface, showing that the corrected net rainfall undergoes two changes in monotonicity as the neutralized rainfall when the reservoir is less than half utilized ($h_p < c_p/2$). In contrast, when the reservoir is fully or nearly fully utilized ($h_p \approx c_p$), the corrected flux $P_n - \tilde{P}_s$ behaves similarly to the original flux $P_n - P_s$. Interestingly, a non-linear infiltration behavior is obtained after learning with the hybrid GRNN model structure, especially for drier conditions of the production reservoir where classical GR models are known to fail in flood generation (cf. Astagneau, Bourgin, Andréassian, and Perrin (2021)). Further research could focus on deeper analysis of learned physical behaviors, for example by investigating the approximation of learned behaviors with known mathematical functions. One could also investigate how to impose physical a priori using other mathematical expressions directly into the forward model structure, for example to impose a explicit monotonicity or even a shape of a dependency, such as the rainfall intensity related modifications of original lumped GR model in Astagneau, Bourgin, Andréassian, and Perrin (2022).

5.4.3 Research perspectives and further discussion

This study proposed a spatially distributed hybrid GR-like model and a comprehensive analysis of a large catchment sample. Future research should concentrate on refining the model's hybridization strategy to enhance its applicability across even larger datasets (e.g., the CARAVAN database in Kratzert, Nearing, et al. (2023)) and to improve extrapolation capabilities for extreme hydrometeorological events. This quest for generalized structures of spatially distributed hydrological models requires scalable hybrid solvers applicable over very large domains. Immediate work will focus on developing a `smash` version for parallel GPU-based forward-inverse computation, and adapting the ϕ-hybrid model to a state-space GR model (cf. Santos et al. (2018)), thereby enabling the investigation of additional non-linearities in hydrological model differential equations. In addition, improving the routing model may deliver a more realistic flood wave propagation. Such improvement could be based on the use of known hydraulic models (e.g., kinematic wave in Roux et al. (2011) and Vergara et al. (2016), non-inertial 1D or 2D in Fleischmann et al. (2020), or full 1D Saint-Venant at network scale in Larnier, Garambois, et al. (2025)), and/or the use of fine topography data such as LiDAR made during low flows (capturing a significant part of river bathymetry), and/or the use of more observations of flow depth, extent, and velocity.

Although hybrid models have more degrees of freedom compared to classical GR models, it is important to note that the inputs and outputs of the flux correction model are physically consistent and of the same dimension as the original model. This design allows the hybrid model to learn nonlinearities in the internal flux laws, which we analyze thoroughly in the flux correction analysis in both time and space throughout the paper. The hybrid models do not necessarily have more conceptual parameters (maintaining the same number of reservoirs and connections here), they do introduce more nonlinearity in the internal flux laws corrections with neural network ϕ_1. This added complexity effectively increases the model's degrees of freedom while maintaining robustness in both spatial and temporal validations, as demonstrated by numerical results.

It is worth noting that the two neural networks will not be extrapolated in the same way when the model is used in prediction. The regionalization neural network ϕ_2 will not be extrapolated as long as the model is used in the region it has been calibrated. At the opposite, the flux correction neural network ϕ_1 is bound to be extrapolated since its inputs (P_n, E_n, h_p, h_t) are varying in time, so that the range observed in calibration will be exceeded sooner or later. This is particularly the case when the model is calibrated locally, as done in the first case study involving 235 French catchments. By contrast, a multi-catchment regionalization setup (Mediterranean case study) is advantageous since it offers more opportunity to expose ϕ_1 to extreme values of its inputs. Quantifying the uncertainty affecting the estimated parameters of the neural networks would be useful

to raise awareness of a likely loss of precision when ϕ_1 is extrapolated, but this comes with many difficulties (e.g., Papamarkou et al., 2022). An alternative approach would be to look for parsimonious regressions that are able to adequately reproduce the behavior revealed by ϕ_1, while being amenable to uncertainty quantification.

Finally, the proposed hybrid hydrological framework should be extended to other models structures, as other GR or VIC available in `smash` platform, but also using more complex physics based modeling approaches and hypothesis testing such as in Douinot et al. (2018) with various subsurface flow modelings. Note that the proposed physics-AI framework for spatially distributed modeling could help unifying top-down approaches such as GR or other data based conceptual models with bottom-up physics based hydrological models that suffer from (up)scaling problems of physical laws and parameterization. In the context of relatively sparse discharge data compared to model dimensionality, such a model discovery process could greatly benefit from the wealth of surface information provided by remote sensing. This includes data on terrain and vegetation properties, surface moisture, snow cover, surface temperature, and total water storage (Meyer Oliveira et al., 2021), along with river network data (e.g., river flow surface topography variability through altimetry and imagery), which necessitates a differentiable river network hydraulic model to achieve coherence with hydraulic observables while enabling the inference of complex and large spatio-temporal parameters from heterogeneous data (Larnier, Garambois, et al., 2025). Such a model would also support information feedback from these data to the hydrological model within a differentiable H&H coupling framework (Pujol, Garambois, and Monnier, 2022).

5.5 Conclusion

This study introduces a hybrid physics-AI framework that integrates neural networks to infer spatio-temporal internal fluxes and spatially distributed conceptual parameters within a differentiable, gridded hydrological model, all encapsulated in a VDA algorithm. Numerical results from local calibration-validation across 235 French catchments and regionalization in a complex, flash flood-prone area demonstrate the superiority of the hybrid models. These models excel not only in performance scores during both calibration and validation but also in producing physically interpretable results, with improved representations of simulated hydrological behavior.

The proposed approach, relying on process-based equations hybridized with ANNs, allows obtaining interpretable spatially distributed hydrological models, contrarily to pure machine learning approaches, while taking advantage of non-linear and multi-resolution effects of neural networks. Accordingly, it is applicable to any other differentiable hydrological, hydraulic or geophysical model, on structured or unstructured meshes.

Chapter 6

Integrating Neural Networks into an Ordinary Differential Equation System Solved Using an Implicit Numerical Scheme

Contents

6.1 Introduction . . . 109
6.2 Modeling approach . . . 109
6.2.1 Hybrid continuous state-space structure for a regionalizable and gridded hydrological model . . . 110
6.2.2 Resolution of neural ODEs within an implicit Euler scheme . . . 112
6.2.3 Model training . . . 113
6.3 Case study and results . . . 114
6.3.1 Study area and experimental design . . . 114
6.3.2 Validation of the ODE solver . . . 116
6.3.3 Model performance and interpretation . . . 117
6.4 Conclusion . . . 123

This chapter is based on a manuscript that has been submitted to a peer-reviewed journal. The work focuses on hybridizing an ODE solver with neural networks, following a similar approach to Chapter 5. However, in this case, the neural network is embedded within a continuous state-space model, which is solved using an implicit numerical scheme rather than being obtained through analytical resolution based on an algebraic formulation.

6.1 Introduction

Throughout the preceding chapters, we have demonstrated the efficiency and promise of hybrid approaches that integrate ML/DL components into physically based models. Despite their significant advantages over traditional models, hybrid approaches also face their own set of challenges. A primary limitation is maintaining physical consistency throughout the hybrid framework while leveraging the flexibility of data-driven approaches (Reichstein et al., 2019). Moreover, the integration of ML components with physical models may introduce complexities in model coupling and error propagation across different components (Frame et al., 2021). Recent research suggests that ensuring numerical stability and physical process consistency, particularly when embedding neural networks into differential equation-based hydrological models, may help address some of these key challenges (Rackauckas et al., 2021; Höge et al., 2022; Y. Song et al., 2024). For instance, Höge et al., 2022 proposed an approach based on neural ordinary differential equations (neural ODEs), where neural networks are implemented within an implicit numerical scheme. Neural ODEs provide a mathematically grounded approach to hybrid modeling that not only reduces numerical errors but also enhances the stability of hydrological simulations under varying conditions. Such hybrid approaches would be particularly promising if implemented within a variational data assimilation (VDA) framework for spatially distributed hydrological modeling (Jay-Allemand, Javelle, et al., 2020; Colleoni, Garambois, et al., 2022). However, their implementation remains challenging due to high-dimensional parameter spaces, especially when coupling the adjoint model to obtain spatial cost gradients as seen in Chapter 3 and Chapter 5, and integrating satellite-based moisture maps into the cost function (Ettalbi, Garambois, Huynh, Arnaud, et al., 2025).

This chapter proposes a hybrid physics-AI approach that integrates neural ODEs, solved using an implicit numerical scheme, into a differentiable, regionalizable, and spatially distributed hydrological model within a VDA framework. The hydrological model is based on a continuous state-space GR structure (Perrin et al., 2003; Santos et al., 2018), and an integrated process-parameterization neural network. Additionally, the regionalization neural network from Chapter 3 is extended to explore alternative neural network architectures to improve the adaptability and scalability of parameter estimation. The development aims to bridge the gap between ML flexibility and physical model interpretability, in view to uncovering hydrological behaviors and scale-relevant theories inferred with AI techniques. It enables addressing several open research questions by providing a robust, versatile and powerful tool for enhancing flood forecasting systems, mitigating structural uncertainty in modeling, optimizing data efficiency, and enabling more effective multi-scale information extraction through hybrid flux correction.

6.2 Modeling approach

This study employs three key components. First, we implement the continuous state-space GR structure presented in Santos et al., 2018 into `smash`, in addition to the classical GR model without an explicitly formulated water balance in Perrin et al., 2003. This state-space model solves the water balance differential equations continuously using numerical schemes, instead of splitting the equations to compute the solutions analytically. Second, the seamless regionalization method using

neural networks, HDA-PR, proposed in Chapter 3, enabling the estimation of conceptual hydrological parameters from physical descriptors, is extended to incorporate convolutional neural networks (CNNs) in addition to multilayer perceptrons (MLPs). Finally, the process-parameterization neural network previously used for the analytical resolution of a discrete state-space model in Chapter 5—which relied on an algebraic structure—is now directly embedded into the ODEs. These ODEs are solved using the Newton-Raphson method within an implicit Euler scheme, which mitigates numerical errors that arise when using simple explicit schemes with sequential computations and split operators.

6.2.1 Hybrid continuous state-space structure for a regionalizable and gridded hydrological model

The hydrological model $\mathcal{M}$ is constructed by partially composing two operators: (i) a learnable, neural network-based operator ϕ, predicting parameters and fluxes corrections of (ii) a dynamic and spatially distributed rainfall-runoff model with simplified hydraulic routing, consisting in a set of chained differential equation and denoted as $\mathcal{M}_{rr-hy}$, resulting in Equation 6.1:

$$\mathcal{M} = \mathcal{M}_{rr-hy}\left(\,.\,,\ \phi\left(\,.\,\right)\right) \tag{6.1}$$

The rainfall-runoff model considered in this study follows a GR-like structure, as described in Chapter 3. Now incorporating $\boldsymbol{f}_q$, the vector of spatially distributed corrections applied to $\boldsymbol{q}$ (the internal fluxes), it is expressed as in Equation 6.2:

$$\boldsymbol{U}(x,t) = [\boldsymbol{Q},\boldsymbol{h},\boldsymbol{q}]\,(x,t) = \mathcal{M}_{rr-hy}\left(\mathcal{D}_\Omega, \boldsymbol{\mathcal{I}}(x,t); \boldsymbol{f}_q(x,t), [\boldsymbol{\theta},\boldsymbol{h}_0]\,(x)\right) \tag{6.2}$$

A neural network-based estimator ϕ, consisting of a pair of neural networks, takes as input (i) neutralized (by interception reservoir) atmospheric inputs $\boldsymbol{\mathcal{I}}_n = (P_n, E_n)(x,t)$, along with the model states at previous time step $\boldsymbol{h}(x,t-1)$, to correct spatio-temporal internal fluxes $\boldsymbol{q}(x,t)$ (process-parameterization pipeline) and (ii) physical descriptors $\boldsymbol{\mathcal{D}}(x)$ to estimate spatialized hydrological parameters $\boldsymbol{\theta}(x)$ (regionalization pipeline), as shown in Equation 6.3.

$$\phi : \begin{cases} \boldsymbol{f}_q(x,t) & = \phi_1\left(\boldsymbol{\mathcal{I}}_n(x,t), \boldsymbol{h}(x,t-1); \boldsymbol{\rho}_1\right) \\ \boldsymbol{\theta}(x) & = \phi_2\left(\boldsymbol{\mathcal{D}}(x); \boldsymbol{\rho}_2\right) \end{cases} \tag{6.3}$$

with $\boldsymbol{\rho} = (\boldsymbol{\rho}_1, \boldsymbol{\rho}_2)$ the vector of trainable parameters, invariant to the spatial coordinate x over Ω, of the neural network(s) ϕ. In this study, the neural networks considered are two MLPs, or an MLP for water flux corrections and a CNN for regional parameter estimation.

Now, the mathematical details of the hybrid hydrological model operators can be summarized as follows. For a given cell $x \in \Omega$ and time step $t > 0$, $P(x,t)$ and $E(x,t)$ represent the local precipitation and potential evapotranspiration. For simplicity and readability, spatio-temporal dependencies are omitted, with flux corrections $\boldsymbol{f}_q$ highlighted in brown and regionalizable conceptual parameters $\boldsymbol{\theta}$ in turquoise in the equations.

First, an interception reservoir with a capacity of c_i, automatically computed using the flux matching technique (Ficchì et al., 2019), allows for the computation of the neutralized rainfall P_n and neutralized evapotranspiration E_n.

Next, the model states $\boldsymbol{h}$ are computed by solving a neural ODE system, generally expressed as:

$$\frac{d\boldsymbol{h}}{dt} = f_{phys}\left(\,.\,, \boldsymbol{h}, \phi(\,.\,, \boldsymbol{h}; \boldsymbol{\rho})\right) \tag{6.4}$$

where f_{phys} is the set of physically-based equations. This neural ODE system employs the following two key components:

— The neural network ϕ takes the model states as part of its inputs, thus affecting the model dynamics and state gradient information. It is expected to learn the model behavior by leveraging memory effects through state updates.
— The set of physical equations f_{phys} integrates the network ϕ_1 as a complementary component, which refines the internal water fluxes that describe the state dynamics. This differs from partly replacing internal fluxes as $\frac{d\boldsymbol{h}}{dt} = f_{phys} \circ \phi(., \boldsymbol{h}; \boldsymbol{\rho})$, or fully replacing them as $\frac{d\boldsymbol{h}}{dt} = \phi(., \boldsymbol{h}; \boldsymbol{\rho})$ (e.g., Höge et al. (2022)). The approach allows the neural ODEs to preserve an original structure driven purely by physical equations, rather than directly relying on neural network outputs, and enables learning under stronger physical constraints.

In this study, we consider f_{phys} in Equation 6.4 to be a set of GR production and transfer operators. The GR production state h_p and transfer state h_t are computed by solving Equation 6.5. The numerical resolution of this neural ODE system, ensuring both accurate resolution and numerical differentiability with respect to model parameters for optimization, will be explained in Section 6.2.2.

$$\frac{d\boldsymbol{h}}{dt} = \begin{pmatrix} \frac{dh_p}{dt} \\ \frac{dh_t}{dt} \end{pmatrix} = \begin{pmatrix} \left(1 - \left(\frac{h_p}{c_p}\right)^{\alpha_1}\right) P_n(1 + f_{q,1}) - \frac{h_p}{c_p}\left(2 - \frac{h_p}{c_p}\right) E_n(1 + f_{q,2}) \\ 0.9\left(\frac{h_p}{c_p}\right)^{\alpha_1} P_n(1 + f_{q,1}) + k_{exc}\left(\frac{h_t}{c_t}\right)^{\alpha_3}(1 + f_{q,3}) - \frac{c_t}{\alpha_2 - 1}\left(\frac{h_t}{c_t}\right)^{\alpha_2}(1 + f_{q,4}) \end{pmatrix} \tag{6.5}$$

where $\alpha_1 = 2$, $\alpha_2 = 5, \alpha_3 = 3.5$ are classical GR constants (cf. Perrin et al. (2003) and Santos et al. (2018)); $f_{q,i=1..4}$ are the corrections applied to internal fluxes and predicted by the process-parametrerization neural network ϕ_1; c_p, c_t, and k_{exc} represent the capacities of the production and transfer reservoirs, and the exchange coefficient, which are the conceptual production parameters predicted by the regionalization neural network ϕ_2. Then, hydrological runoff flux (lateral discharge feeding the routing module) produced at the pixel scale is computed by the closure equation of the ODE system as follows:

$$Q_{lat} = 0.1\left(\frac{h_p}{c_p}\right)^{\alpha_1} P_n(1 + f_{q,1}) + k_{exc}\left(\frac{h_t}{c_t}\right)^{\alpha_3}(1 + f_{q,3}) + \frac{c_t}{\alpha_2 - 1}\left(\frac{h_t}{c_t}\right)^{\alpha_2}(1 + f_{q,4}) \tag{6.6}$$

It is important to note that the values of $f_{q,i=1..4}$ are constrained within the range of -1 to 1 due to the TanH activation function in the output layer of ϕ_1. Consequently, the transformation functions applied to these internal flux corrections (e.g., $1 + f_{q,1}$, $1 + f_{q,2}$, etc.) preserve the structure of the original conceptual model when $\boldsymbol{f}_q \equiv \boldsymbol{0}$, as all transformations result in a value of 1 in that case. These terms were defined according to the specific fluxes being corrected and relevant mathematical constraints.

Finally, the routing module, previously explained in Section 5.2.2 of Chapter 5, employs a 1D kinematic wave model (considering a "D8" drainage plan $\mathcal{D}_\Omega(x)$), leading to a single-equation discharge propagation model:

$$\partial_x Q + a_{kw} b_{kw} Q^{(b_{kw}-1)} \partial_t Q = \lambda Q_{lat} \tag{6.7}$$

where a_{kw} and b_{kw} are the routing parameters to be estimated by the regionalization neural network ϕ_2, λ is the conversion factor. This equation is solved numerically using a classical finite differences approach (cf. Te Chow et al. (1988)).

The following section will detail the numerical method used to solve the ordinary differential equation 6.5, which includes the neural network ϕ_1 that depends on the ODE state $\boldsymbol{h}$, hence called a neural ODE.

6.2.2 Resolution of neural ODEs within an implicit Euler scheme

To solve Equation 6.5, we employ an implicit Euler scheme. For a small time step dt, by defining $\dot{\boldsymbol{h}} = \frac{d\boldsymbol{h}}{dt}$, we have:

$$\boldsymbol{h}(t+dt) = \boldsymbol{h}(t) + dt \cdot \dot{\boldsymbol{h}}\left(\boldsymbol{h}(t+dt)\right) \tag{6.8}$$

Now, define:

$$\boldsymbol{g}\left(\boldsymbol{h}(t+dt)\right) = \boldsymbol{h}(t+dt) - \boldsymbol{h}(t) - dt \cdot \dot{\boldsymbol{h}}\left(\boldsymbol{h}(t+dt)\right) \tag{6.9}$$

Approximating the sought state $\boldsymbol{h}(t+dt)$ thus reduces to numerically solving the equation:

$$\boldsymbol{g}(y) = y - c - dt \cdot \dot{\boldsymbol{h}}(y) = \boldsymbol{0} \tag{6.10}$$

where $y = \boldsymbol{h}(t+dt)$ and $c = \boldsymbol{h}(t)$. Then, the solution of Equation 6.10 is approximated using the Newton-Raphson method as follows:

$$\begin{cases} y_0 = c, \\ y_{n+1} = y_n + \Delta y, \quad \text{where } \Delta y \text{ is the solution of } \nabla \boldsymbol{g}(y_n) \cdot \Delta y + \boldsymbol{g}(y_n) = \boldsymbol{0} \end{cases} \tag{6.11}$$

The Jacobian matrix $\nabla \boldsymbol{g}$ is given by:

$$\nabla \boldsymbol{g} = \begin{pmatrix} 1 - dt\frac{\partial \dot{h_p}}{\partial h_p} & \frac{\partial h_p}{\partial h_t} - dt\frac{\partial \dot{h_p}}{\partial h_t} \\ \frac{\partial h_t}{\partial h_p} - dt\frac{\partial \dot{h_t}}{\partial h_p} & 1 - dt\frac{\partial \dot{h_t}}{\partial h_t} \end{pmatrix} \approx \begin{pmatrix} 1 - dt\frac{\partial \dot{h_p}}{\partial h_p} & -dt\frac{\partial \dot{h_p}}{\partial h_t} \\ -dt\frac{\partial \dot{h_t}}{\partial h_p} & 1 - dt\frac{\partial \dot{h_t}}{\partial h_t} \end{pmatrix} \tag{6.12}$$

This simplification holds because $\frac{\partial h_p}{\partial h_t} = 0$ since production does not depend on transfer. Additionally, we assume that h_t depends on h_p only through its time derivative rather than instantaneously. Thus, h_t evolves as an accumulated effect of h_p, meaning its dependence on h_p is indirect and primarily through its time derivative, justifying the approximation $\frac{\partial h_t}{\partial h_p} \approx 0$. The remaining terms, which are $\frac{\partial \dot{h_p}}{\partial h_p}$, $\frac{\partial \dot{h_p}}{\partial h_t}$, $\frac{\partial \dot{h_t}}{\partial h_p}$ and $\frac{\partial \dot{h_t}}{\partial h_t}$, can be derived analytically with distinct formulations depending on whether the process-parameterization neural network ϕ_1 is included, as follows.

First, we introduce a variable change: $\tilde{\boldsymbol{h}} = \left(\tilde{h_p}; \tilde{h_t}\right)^T = \left(\frac{h_p}{c_p}; \frac{h_t}{c_t}\right)^T$. Since the conceptual parameters c_p and c_t remain constant over time, their derivatives vanish, leading to: $\frac{d\tilde{h_p}}{dt} = \frac{1}{c_p}\frac{dh_p}{dt}$ and $\frac{d\tilde{h_t}}{dt} = \frac{1}{c_t}\frac{dh_t}{dt}$. Then, the Jacobian matrix $\nabla \boldsymbol{g}$ in Equation 6.12 can be computed as follows in two different cases:

For classical continuous state-space structure (ODE). The process-parameterization neural network is absent, we set $\boldsymbol{f}_q \equiv \boldsymbol{0}$ in Equation 6.5. The resulting Jacobian components are:

$$\begin{aligned} \frac{\partial \dot{h_p}}{\partial \tilde{h_p}} &= -\alpha_1 P_n \tilde{h_p}^{\alpha_1 - 1} - 2E_n(1 - \tilde{h_p}) \\ \frac{\partial \dot{h_p}}{\partial \tilde{h_t}} &= 0 \\ \frac{\partial \dot{h_t}}{\partial \tilde{h_p}} &= 0.9\alpha_1 P_n \tilde{h_p}^{\alpha_1 - 1} \\ \frac{\partial \dot{h_t}}{\partial \tilde{h_t}} &= \alpha_3 k_{exc} \tilde{h_t}^{\alpha_3 - 1} - \frac{\alpha_2}{\alpha_2 - 1} c_t \tilde{h_t}^{\alpha_2 - 1} \end{aligned} \tag{6.13}$$

For hybrid continuous state-space structure (neural ODE). When incorporating the process-parameterization neural network $\phi_1(., \boldsymbol{h})$, which depends on the ODE state and predicts

the model flux corrections $\boldsymbol{f}_q$, one obtains a set of so-called neural ODEs (Equation 6.5). The resulting Jacobian components become:

$$
\begin{aligned}
\frac{\partial \dot{\tilde{h}_p}}{\partial \tilde{h}_p} &= P_n \left(\left(1 - \tilde{h_p}^{\alpha_1}\right) \frac{\partial f_{q,1}}{\partial \tilde{h}_p} - \alpha_1 \tilde{h_p}^{\alpha_1 - 1} \left(1 + f_{q,1}\right) \right) - E_n \left(\tilde{h_p} \left(2 - \tilde{h_p}\right) \frac{\partial f_{q,2}}{\partial \tilde{h}_p} + 2 \left(1 - \tilde{h_p}\right) \left(1 + f_{q,2}\right) \right) \\
\frac{\partial \dot{\tilde{h}_p}}{\partial \tilde{h}_t} &= P_n \left(1 - \tilde{h_p}^{\alpha_1}\right) \frac{\partial f_{q,1}}{\partial \tilde{h}_t} - E_n \left(2 - \tilde{h_p}\right) \tilde{h_p} \frac{\partial f_{q,2}}{\partial \tilde{h}_t} \\
\frac{\partial \dot{\tilde{h}_t}}{\partial \tilde{h}_p} &= 0.9 P_n \tilde{h_p}^{\alpha_1 - 1} \left(\alpha_1 \left(1 + f_{q,1}\right) + \tilde{h_p} \frac{\partial f_{q,1}}{\partial \tilde{h}_p} \right) + k_{exc} \tilde{h_t}^{\alpha_3} \frac{\partial f_{q,3}}{\partial \tilde{h}_p} - \frac{c_t \tilde{h_t}^{\alpha_2}}{\alpha_2 - 1} \frac{\partial f_{q,4}}{\partial \tilde{h}_p} \\
\frac{\partial \dot{\tilde{h}_t}}{\partial \tilde{h}_t} &= 0.9 P_n \tilde{h_p}^{\alpha_1} \frac{\partial f_{q,1}}{\partial \tilde{h}_t} + k_{exc} \tilde{h_t}^{\alpha_3 - 1} \left(\alpha_3 (1 + f_{q,3}) + \tilde{h_t} \frac{\partial f_{q,3}}{\partial \tilde{h}_t} \right) - \frac{c_t \tilde{h_t}^{\alpha_2 - 1}}{\alpha_2 - 1} \left(\alpha_2 (1 + f_{q,4}) + \tilde{h_t} \frac{\partial f_{q,4}}{\partial \tilde{h}_t} \right)
\end{aligned}
\tag{6.14}
$$

where the neural network Jacobian $\frac{\partial \boldsymbol{f}_q}{\partial \boldsymbol{h}}$ is computed explicitly using the backpropagation method, which determines the gradient of its outputs with respect to its inputs.

We note that all components of the Jacobian matrix, including those from the neural network, are computed directly from the analytical mathematical formulations, which significantly reduces the computational cost compared to calling a numerical adjoint during the resolution of the ODEs with an implicit scheme. Alternative methods, which do not directly rely on mathematical formulations but could be more computationally intensive, include using linear tangents or, more recently, the "discretize-then-optimize" adjoint method proposed in Y. Song et al., 2024, which aims to reduce the time consumption associated with adjoint calls in a differentiable implicit numerical scheme.

The implementation of the ODE solvers is validated against the reference algebraic model, using a test case in a small catchment in France. This test shows moderate differences, mostly for high flows, in model states and responses between the model structure with algebraic resolution and the models that use numerical resolution to solve the ODEs when running a forward pass with the same model parameters. These differences, observed under similar testing conditions, may stem from the fact that the algebraic model computes fluxes sequentially with exact analytical expressions, whereas the ODE resolution, though approximate, solves fluxes simultaneously. However, upon calibration, these different models yield distinct parameter maps, representing different effective functioning points for each approach, as will be detailed and discussed in the results section.

6.2.3 Model training

First, the optimization problem, using a multi-gauge calibration cost function on discharge and the chained gradients of the regionalization neural network Jacobian, is described in Section 3.2.2 and Section 3.2.3 in Chapter 3 (the hydrological model $\mathcal{M}_{rr}$ is replaced by the rainfall-runoff model with a simplified hydraulic routing $\mathcal{M}_{rr-hy}$). The neural network(s) are trained as follows:

— **Pre-calibration of conceptual parameters:** In the first step, we pre-train only the regionalization neural network ϕ_2 to find a spatially distributed first guess for the conceptual parameters. The weights of the process-parameterization neural network ϕ_1, which is embedded in the ODE solver, are initialized using a normal distribution centered at zero with a small variance. This ensures that, while the non-zero initialization allows for proper flow of gradients in the network, the outputs of ϕ_1 being close to zero result in minimal corrections, thus preserving the original hydrological model structure. In other words, ϕ_1 has very limited impact during this phase. The pre-training uses a relatively high learning rate of 0.004 for faster gradient descent over 40 iterations. In the case of lumped parameters (i.e., $\boldsymbol{\rho} = \bar{\boldsymbol{\theta}}$ and $\phi_2 \equiv \mathrm{id}_{\phi_2}$), we simply use a heuristic search algorithm to provide a uniform first guess.

— **Main training:** In this phase, both neural networks are trained simultaneously (or ϕ_1 and the conceptual parameters in the case of lumped parameters). The training uses a smaller learning rate of 0.003 to ensure stability over 240 iterations. The gradients of ϕ_2 are computed using the coupling approach described in Section 3.2.3, while the gradients of ϕ_1, which is embedded in the differentiable Fortran code, are computed using the adjoint model.

It is worth noting that the embedded network ϕ_1 must be twice differentiable mathematically (once for solving the ODEs and once for the calibration process). While many activation functions in neural networks are not differentiable at zero (e.g., ReLU), stochastic optimization can often bypass this problem since the network generally produces non-zero (or non-close-to-zero) values. However, as we aim to preserve the original structure by producing outputs close to zero during pre-calibration, numerical errors can arise. To mitigate this issue, we use the SiLU (Sigmoid Linear Unit) activation function, which is twice differentiable everywhere and provides smooth gradients.

6.3 Case study and results

6.3.1 Study area and experimental design

The methods are tested on the Aude River basin, located in southern France. This area comprises 25 catchments, including 12 upstream gauges and 13 downstream gauges (Figure 6.1). The models

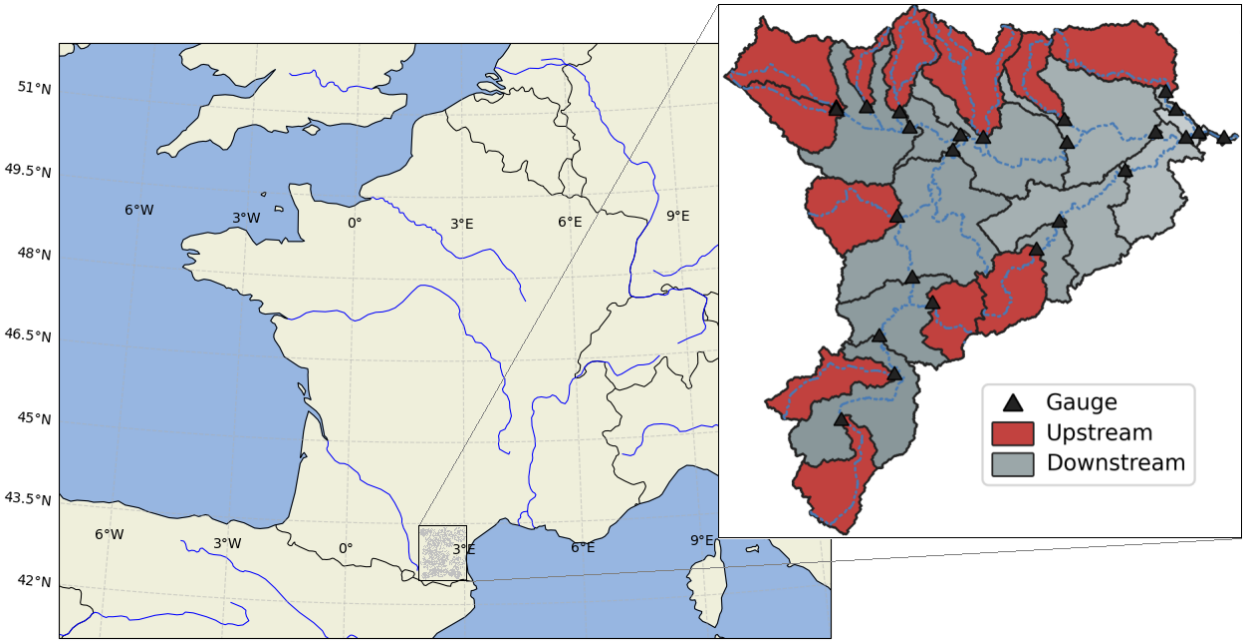


Figure 6.1 – The Aude River basin, located in southern France, consists of 25 sub-catchments, including 12 upstream catchments (red) and 13 downstream catchments (gray).

are run at a spatial resolution of 1 km and an hourly time step, using national databases as described in Section 1.4.2 of Chapter 1. The study period spans 9 years, from August 2014 to July 2023, divided into two sub-periods: P1 (2015-2019) and P2 (2019-2023). We use additional one-year warm-up periods (2014-2015 for P1 and 2018-2019 for P2) to calibrate or validate the models. Period P1 is used for calibration, while P2 is used for validation.

We perform calibration across multiple catchments using upstream gauges and evaluate regionalization performance by transferring parameters to downstream gauges, which represents a more

challenging regionalization case compared to transferring parameters from downstream to upstream gauges, as seen in Chapter 3. A set of seven descriptors is used as inputs for the regionalization mapping ϕ_2 (Figure 6.2, see Table 1.1 in Chapter 1 for a detailed description). We evaluate the

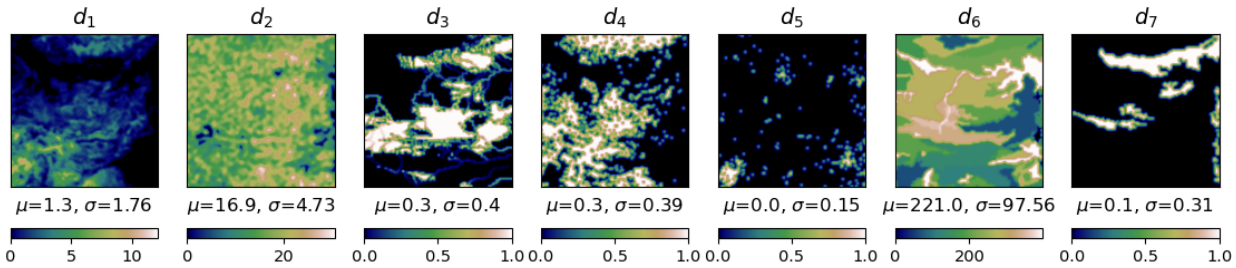


Figure 6.2 – Maps of seven physical descriptors in the Aude River basin at a resolution of 0.01° in the WGS 84 projection, where μ and σ represent the spatial average and standard deviation for each descriptor.

models based on model structure flexibility and regionalization ability. Model structure flexibility is assessed by the model's capacity to produce interpretable water flux dynamics, both with and without the process-parameterization network ϕ_1. Regionalization ability is evaluated based on the capability of the neural networks ϕ_2, which can be either a multilayer perceptron (MLP) or a convolutional neural network (CNN), to regionalize the conceptual parameters using physical descriptors. The descriptions of the evaluated models are provided in Table 6.1.

Table 6.1 – Summary of evaluated models. The notation consists of two parts, separated by a dot. The first part describes the model structure: GR denotes the GR4 structure (Perrin et al., 2003), which uses an algebraic formulation for state updates; ODE denotes the continuous GR4-like state-space model (Santos et al., 2018); and NODE denotes the neural ODE structure that incorporates a neural network into the ODEs. The second part indicates the mapping used to constrain the conceptual parameters: U represents lumped parameters without constraints; while MLP or CNN indicate regionalization mappings based on MLP or CNN. All models include a kinematic wave routing over a flow direction grid.

Notation	Model	Description
GR.U	Algebraic (lumped parameters)	GR4 model with lumped parameters
GR.MLP	Algebraic + MLP	Grid-based GR4 model using an MLP for regionalization
GR.CNN	Algebraic + CNN	Grid-based GR4 model using a CNN for regionalization
ODE.U	ODE (lumped parameters)	Continuous state space GR4-like model with lumped parameters
ODE.MLP	ODE + MLP	Continuous state-space GR4-like model with MLP-based regionalization
ODE.CNN	ODE + CNN	Continuous state space GR4-like model with CNN-based regionalization
NODE.U	Neural ODE (lumped parameters)	Neural ODE-integrated GR4-like model with lumped parameters
NODE.MLP	Neural ODE + MLP	Neural ODE-integrated GR4-like model with MLP-based regionalization
NODE.CNN	Neural ODE + CNN	Neural ODE-integrated GR4-like model with CNN-based regionalization

6.3.2 Validation of the ODE solver

A major advantage of numerically solving ODEs, as opposed to algebraic resolution methods, lies in their generalizability. While an analytical solution derived through an algebraic approach is exact, it can only be obtained under specific assumptions about the ODE system (e.g., certain fixed values of the coefficients $(\alpha_i)_{i=1..3}$ in Equation 6.5). In contrast, numerical methods can solve ODEs without requiring such assumptions, albeit with approximate solutions. It is thus necessary to validate the ODE solver (ODE solutions obtained by numerical scheme) against the algebraic approach before performing any numerical experiments based on this solver. This is particularly important as this work presents the first implementation of both classical and hybrid ODE solvers into a fully distributed hydrological modeling and VDA framework.

To carry out this validation, we compare the GR algebraic model with lumped parameters (GR.U), which solves an analytical solution of the time-integrated ODEs, and the continuous state-space model also with lumped parameters (ODE.U), which solves the ODEs using an implicit numerical scheme. For both models, we set identical conceptual parameters and initial states, which are non-calibrated. Figure 6.3 shows similar simulated hydrological responses for GR.U and ODE.U obtained at the most downstream gauge. Although slight differences are found in the production

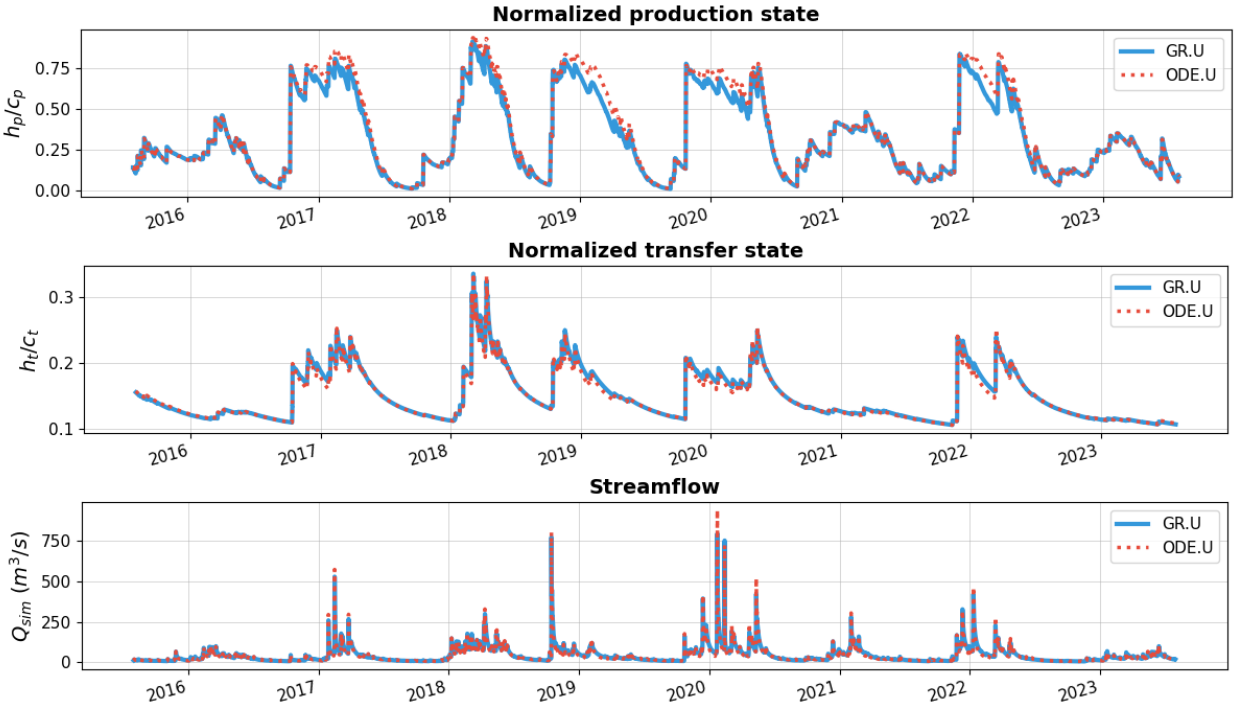


Figure 6.3 – Comparison of normalized production state (top), normalized transfer state (middle), and simulated streamflow (bottom) at the most downstream gauge, obtained by algebraic resolution (GR.U) and numerical resolution (ODE.U) using the same lumped conceptual parameters and initial states.

state (often higher state for ODE.U) at certain periods, the transfer state for both models remains nearly identical, while streamflow simulation shows minor differences at several peak flows.

In addition to similar temporal hydrological responses, Figure 6.4 demonstrates nearly identical spatial patterns and values of the final model states obtained for both methods. The bias map between ODE.U and GR.U for the production state shows small deviations centered around zero, while the bias for the transfer state is nearly negligible, aligning with the earlier temporal

observations. These minor discrepancies between the two approaches can be attributed primarily to approximation errors, and the fundamental difference in their solution methodology—namely, the simultaneous numerical resolution of the ODEs for both state variables (h_p and h_t) versus the exact analytical solution obtained through sequential algebraic resolution.

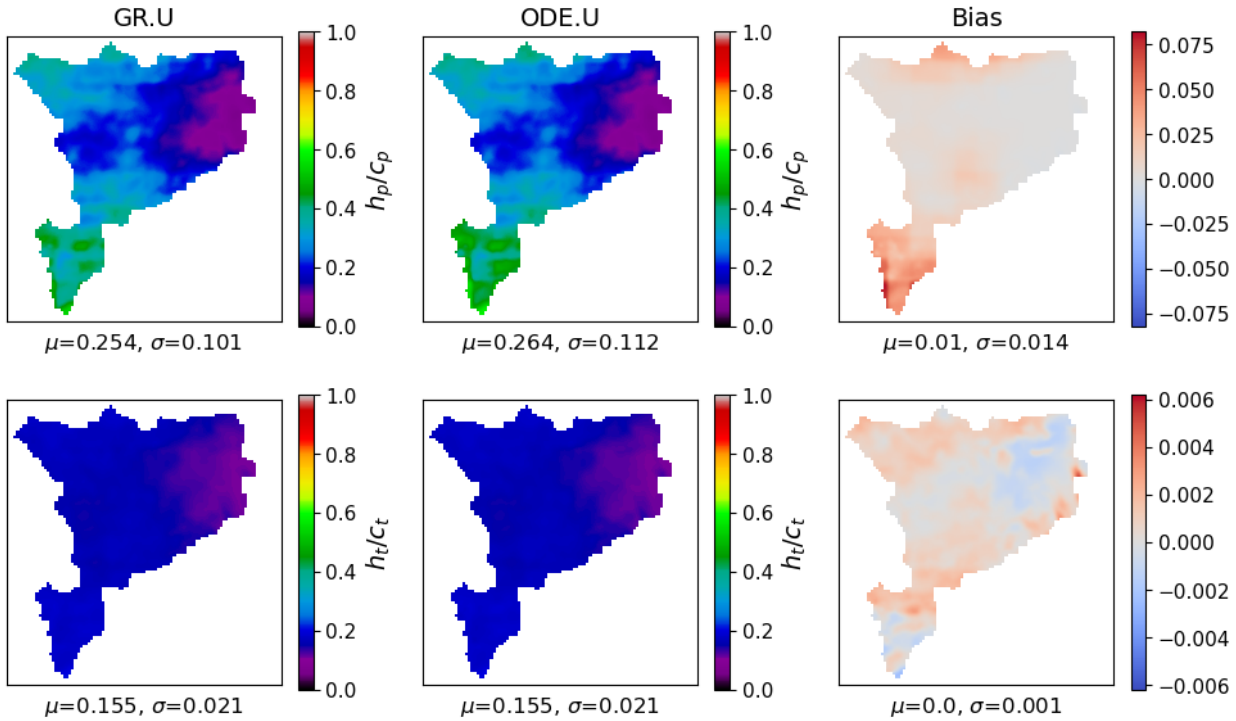


Figure 6.4 – Maps of the normalized production state (h_p/c_p, top) and transfer state (h_t/c_t, bottom) at the final time step, as simulated by GR.U (left), ODE.U (middle), and the corresponding bias (right) computed as the difference between the ODE.U and GR.U states. For each map, μ and σ denote the spatial average and standard deviation.

While both methods initially produce similar hydrological responses, it will be particularly interesting to observe in the next section how the model dynamics evolve during calibration, and whether differences in the numerical formulation influence model behavior and performance.

6.3.3 Model performance and interpretation

Figure 6.5 provides a global view of model performance across calibration and various validation scenarios. Results demonstrate that model scores in calibration increase with the complexity of regionalization mappings (from uniform to MLP and CNN). For each regionalization approach, we observe improved scores progressing from classical GR structure to ODE and NODE structures, exemplified by a median NSE over 0.85 for NODE.CNN compared to 0.82 for ODE.CNN and 0.8 for GR.CNN. Moreover, models with ODE and NODE structures show improved interquartile ranges and whiskers when using CNN or MLP regionalization compared to GR, while NODE.U exhibits larger variance despite maintaining higher median scores than GR.U. For temporal validation, ODE.MLP and NODE.MLP yield similar high median scores of 0.71 compared to 0.59 for GR.MLP. Additionally, NODE.MLP demonstrates a superior interquartile range, with a 0.25-quantile of 0.65 compared to 0.57 for ODE.MLP and 0.42 for GR.MLP. Models with the same regionalization approaches (uniform or CNN) show similar performance across different structures

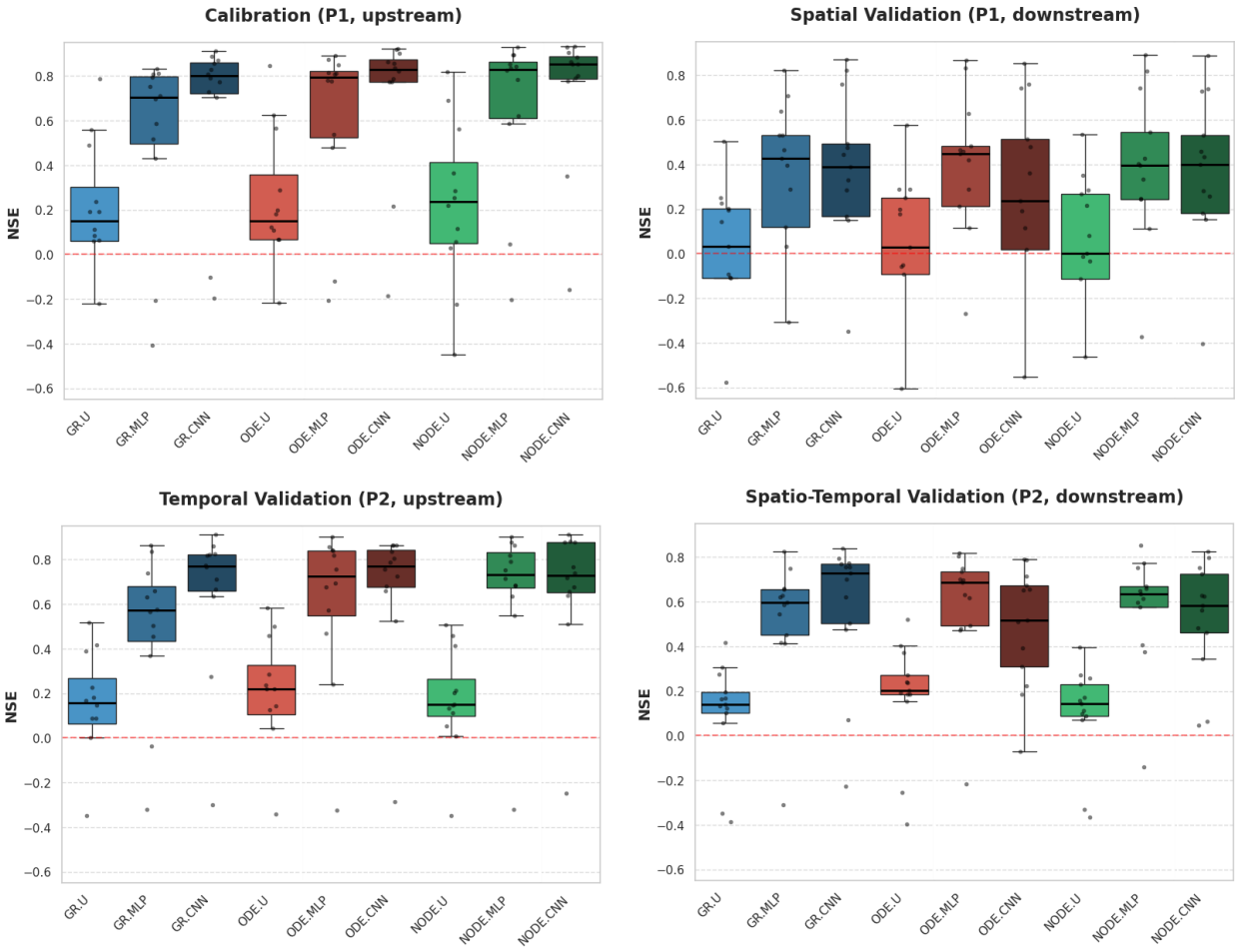


Figure 6.5 – Comparison of model performance for different methods. The NSE scores are computed to evaluate: (i) calibration performance (scores computed over P1 for the 12 upstream gauges), (ii) spatial validation (scores computed over P1 for 13 downstream gauges), (iii) temporal validation (scores computed over P2 for 12 upstream gauges), and (iv) spatio-temporal validation (scores computed over P2 for downstream gauges).

(GR, ODE, NODE), with only slight differences in median and distribution. Spatial validation shows comparable performance across structures for each mapping, except for ODE.CNN which performs relatively worse than GR.CNN and NODE.CNN. While all models struggle with spatial validation, spatio-temporal validation yields generally good performance overall. Although ODE.CNN maintains lower performance compared to GR.CNN and NODE.CNN, the MLP-based models exhibit a more stable performance across validation scenarios. For instance, ODE.MLP achieves the best median scores among the three MLP-based models for spatio-temporal validation, while NODE.MLP yields the best interquartile range. Interestingly, GR.CNN exhibits promising performance in spatio-temporal validation with a median score of 0.71. Overall, GR.CNN and NODE.MLP demonstrate consistently strong and stable performance across all calibration and validation scenarios. While GR.CNN generally outperforms its MLP counterpart in validation scenarios, NODE.MLP shows stable performance with good scores and interquartile ranges across all validation scenarios.

To illustrate the models' flood simulation capabilities, Figure 6.6 presents two representative flood events during the validation period, as observed at an upstream and a downstream gauge. At both locations, all models using uniform mapping significantly underestimate the flood magni-

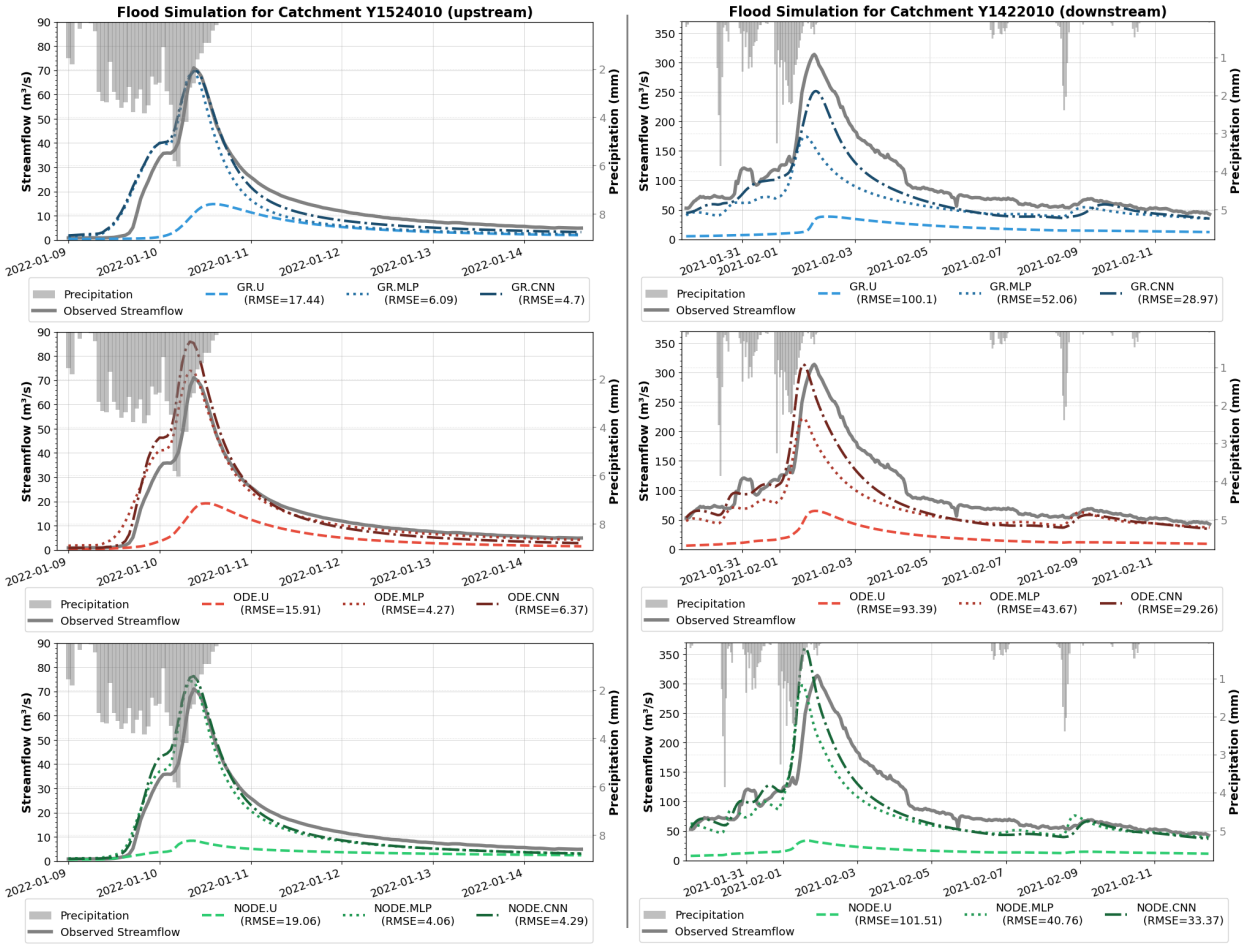


Figure 6.6 – Comparison of flood simulation performance across different methods for an upstream gauge (left) and a downstream gauge (right).

tudes, resulting in the poorest simulation performance. This result highlights the importance of regionalization approaches in accurately simulating floods, particularly in ungauged basins, where traditional methods that use lumped parameters often fail to capture real hydrological dynamics.

For the upstream gauge example, regionalization approaches using MLP and CNN produce relatively similar discharge simulations for each model structure (GR, ODE, NODE). The ODE and NODE structures with MLP show slightly better performance, with RMSE values of 4.27 and 4.06, compared to 6.09 for GR.MLP. Both CNN and MLP mappings applied to the GR structure accurately simulate the peak flow, whereas the regionalization methods for the ODE and NODE structures tend to slightly overestimate peak flow, particularly in the case of ODE.CNN. Despite this, the timing of flood events remains accurate across all regionalization methods, with the rising limbs of the simulated hydrographs closely aligning with observations. Note that in this upstream catchment characterized by a quick hydrological response, the start of the flood event occurs nearly at the same time as the moment of heavy rainfall.

In contrast, the downstream gauge event shows an approximately 8-hour delay between heavy rainfall and flood response. Most models struggle to predict the lag time correctly. This emphasizes the need to improve model realism by accounting for rainfall intensity and studying its impact in triggering non-linear flash flood responses, improving river network hydraulics, which presents a promising avenue for future research. Returning to flood simulation at the downstream, while

the majority of models yield poor performance in predicting the event's beginning, generating flood responses earlier than observed, GR.CNN demonstrates impressive timing accuracy with simulations very close to observations. However, GR structure models, including GR.CNN, still underestimate flood magnitude, whereas ODE.CNN, NODE.MLP, and NODE.CNN perform more accurately.

Figure 6.7 shows the maps of spatially distributed conceptual parameters calibrated using CNN and MLP regionalization approaches. Overall, CNN-based models produce smoother parameter

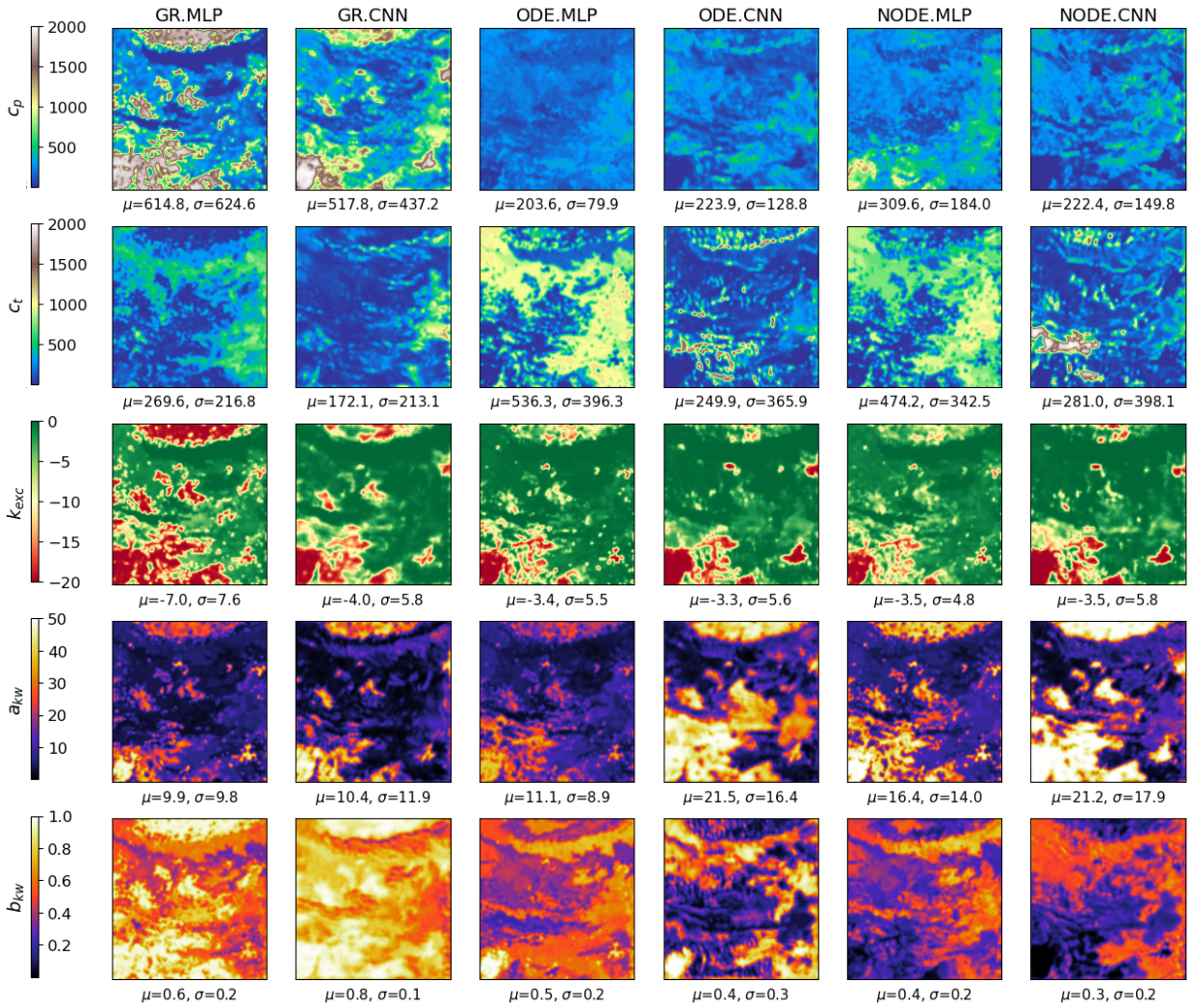


Figure 6.7 – Estimated conceptual parameters across different regionalization methods. The calibrated parameters are $\boldsymbol{\theta}(x) = (c_p(x), c_t(x), k_{exc}(x), a_{kw}(x), b_{kw}(x))^T$ with μ and σ denoting their spatial average and standard deviation.

maps compared to MLP-based models, as evidenced in the maps of c_p and c_t for GR.CNN (compared to GR.MLP), k_{exc} for all CNN-based structures, a_{kw} for all CNN-based structures, and b_{kw} for GR.CNN. This smoothing effect results from the convolution operations applied to physical descriptor maps. Furthermore, we observe spatial patterns from physical descriptors (Figure 6.2) reflected in the parameter maps, such as the pattern at the top of the forest cover map (d_4) or the pattern at the bottom left of the slope (d_1). For each state-space structure, different regionalization mappings produce distinct parameter distributions (e.g., different patterns of GR.MLP compared to GR.CNN, ODE.MLP compared to ODE.CNN, and NODE.MLP compared to NODE.CNN). Ad-

ditionally, notable differences in parameter patterns emerge across model structures, with major differences for GR-based models, while ODE and NODE structures yield relatively similar parameter maps (ODE.MLP compared to NODE.MLP, and ODE.CNN compared to NODE.CNN). This pattern divergence is expected since classical GR models have different state dynamics compared to continuous state-space models, even when employing the same regionalization mapping.

Figure 6.8 examines model dynamics through the spatial average of normalized states (production state h_p and transfer state h_t). The two continuous state-space models (ODE.MLP and

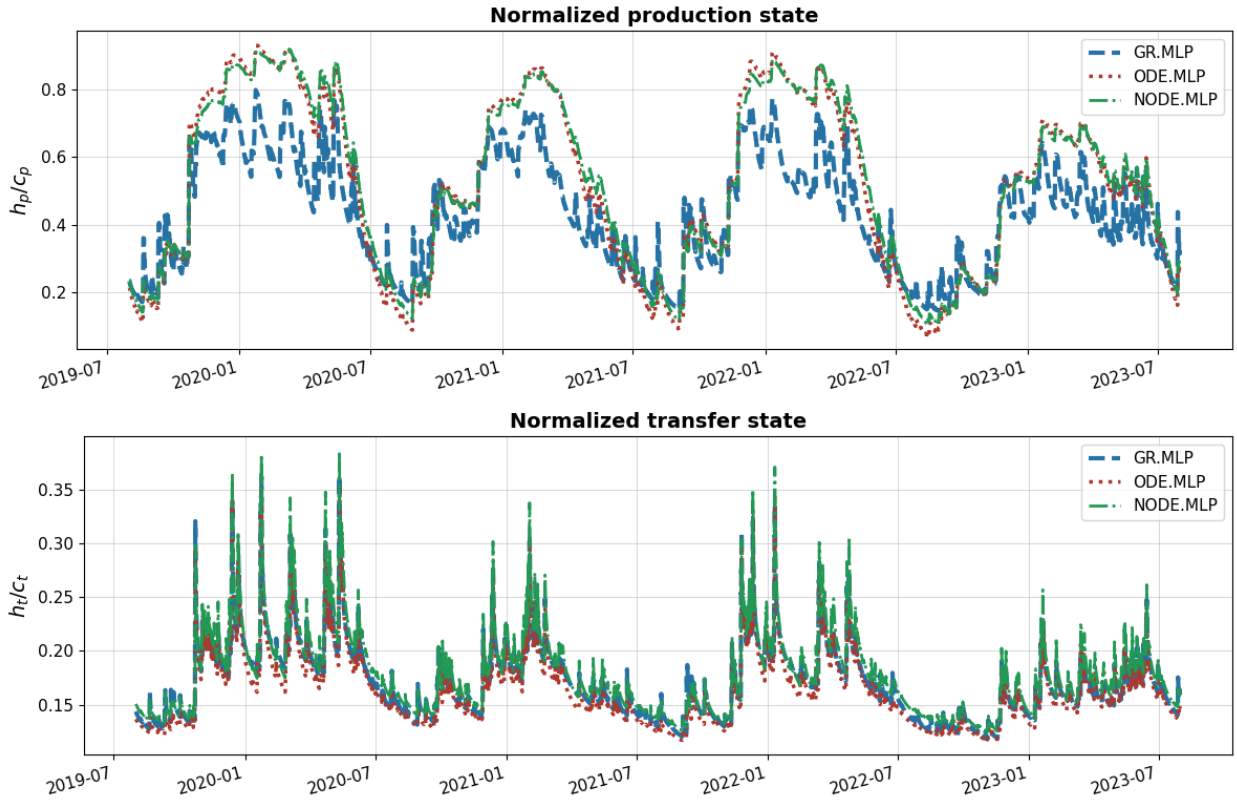


Figure 6.8 – Comparison of the spatial average of the normalized production state $\overline{h_p(t)/c_p(t)}^x$ and transfer state $\overline{h_t(t)/c_t(t)}^x$ during the validation period P2, across the three model structures (GR, ODE, NODE), each employing an MLP for regionalization.

NODE.MLP) demonstrate similar behavior in production state, showing higher values and lower variability compared to GR.MLP. However, for the transfer state, the hybrid NODE.MLP model produces higher values and greater variability compared to the ODE.MLP and GR.MLP models. This likely results from the hybridization effect of the process-parameterization neural network ϕ_1, which incorporates neutralized rainfall to refine the physical equations in the ODE system. This creates a rainfall sensitivity in the transfer state, which is not accounted for in the classical GR and ODE structures. Nevertheless, Figure 6.9 reveals that all three model structures exhibit relatively similar patterns in time-averaged transfer state maps, though with higher mean values (0.18 compared to 0.16 for ODE.MLP and 0.17 for GR.MLP) in the case of NODE.MLP. The maps of time-averaged production state for NODE.MLP and ODE.MLP display different patterns from those of GR.MLP. However, no evident differences emerge between ODE.MLP and NODE.MLP for both h_p and h_t. This is unsurprising as spatial hybridization effects are not expected since the process-parameterization neural network employs a simple MLP (MLP of ϕ_1 and not MLP of the regionalization mapping ϕ_2) that does not account for or accounts less for spatial information.

Future work could employ other types of ϕ_1 to explore this aspect.

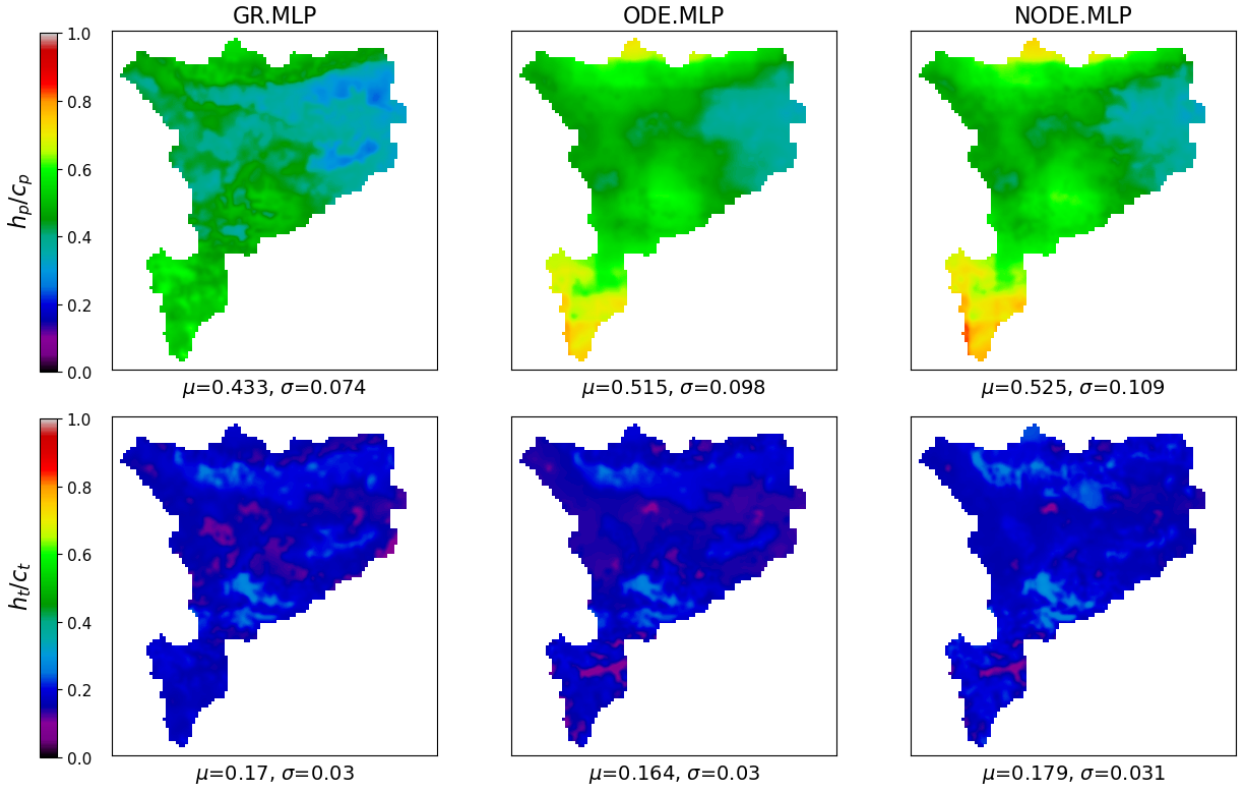


Figure 6.9 – Maps of the time-averaged normalized production state $\overline{h_p(x)/c_p(x)}^t$ and transfer state $\overline{h_t(x)/c_t(x)}^t$ over the validation period P2, for the three model structures (GR, ODE, NODE), each employing an MLP for regionalization. For each map, μ and σ denote the spatial average and standard deviation.

Finally, Figure 6.10 illustrates the hybridization effect on runoff flow (lateral discharge for routing operator) by examining high-pass filtered lateral discharge from the most downstream gauge during the validation period. A 3-day cutoff frequency high-pass filter removes seasonal and long-term pat-

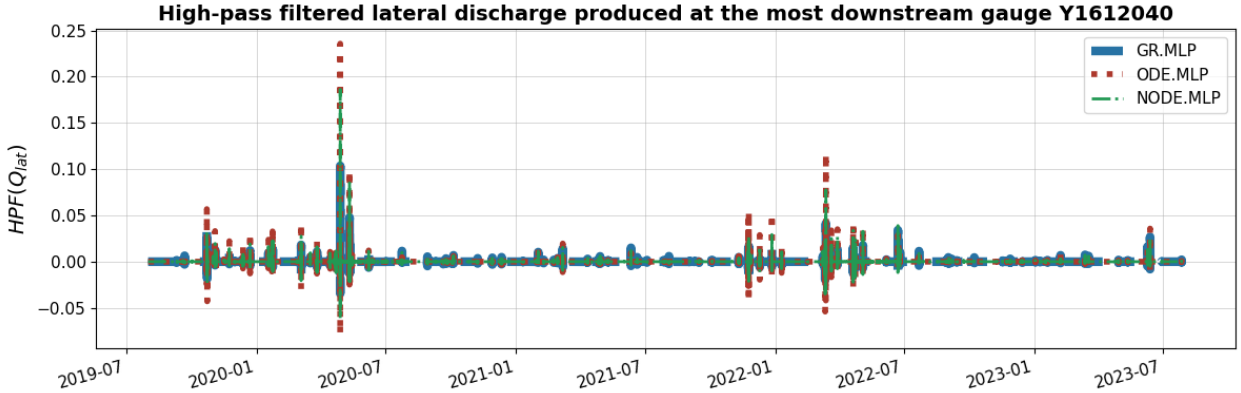


Figure 6.10 – High-pass filtered lateral discharge from the most downstream gauge, using a 3-day cutoff frequency, to emphasize flood events during the validation period P2. The results are shown for three model structures (GR, ODE, NODE), each employing an MLP for regionalization.

terns to focus on flood event behavior. During major events, GR.MLP consistently produces lower lateral discharge compared to ODE.MLP. In certain situations, this results in an overestimation of flood magnitude with the ODE structure and an underestimation with the GR structure. The hybrid NODE structure effectively addresses this issue by refining internal water fluxes in the ODE system, resulting in moderate lateral discharge values that fall between those of ODE.MLP and GR.MLP. This effect is consistent with previously observed flood simulation results, demonstrating the improved hydrological response representation achieved through neural ODE integration.

6.4 Conclusion

This study proposed a framework for integrating neural networks into ordinary differential equations (ODEs) of a process-based hydrological model. The hybrid ODE system, referred to as neural ODEs, incorporates a neural network that refines internal water fluxes and is governed by physical equations to describe the model's state dynamics. Moreover, the hydrological model includes a kinematic wave routing and a regionalization mapping. This mapping enables learning the transfer function between physical descriptors and spatially distributed conceptual parameters, as introduced in Chapter 3, which is now further enhanced by employing a convolution neural network (CNN) in addition to a multilayer perceptron (MLP).

The proposed methods were tested on multiple catchments in the Aude River basin, demonstrating that increased complexity in model structures (from classical GR to ODE and neural ODE structures) and regionalization mapping (from uniform to MLP and CNN) leads to improved model calibration scores. Furthermore, models using CNN-based regionalization mapping generally exhibited smoother estimated parameter maps. This smoothness is a result of the convolution operation, which applies a relatively fine filter to the input descriptors. Evaluation scores from various validation scenarios suggest that the classical GR model using CNN (GR.CNN) and the neural ODE structure using MLP (NODE.MLP) demonstrated consistently strong and stable performance compared to other models. Analysis of state dynamics and runoff flow revealed the hybridization effect of the neural ODE structure, which modifies internal water fluxes to achieve more reliable streamflow simulations during flood events. This preliminary analysis examines recently proposed hybrid solvers, which will be further explored under a broader range of hydrological conditions, experimental hypotheses, and state-parameter analyses.

Chapter 7

Synthesis and Future Pathways in Physics-AI Integration

Since the resurgence of AI in the late 2000s, marked notably by the review paper on the potential and future of deep learning by LeCun, Bengio, et al. (2015), AI has experienced exponential growth, achieving remarkable performance in terms of both precision and computational efficiency. However, many foundational AI models (e.g., MLP, CNN, RNN, LSTM) and key concepts (e.g., backpropagation, gradient-based optimization) that form the core of modern AI were developed much earlier in the 20th century but did not achieve such widespread success. The primary reason for this was the lack of high-quality, extensive data and sufficient computational power to train these models effectively at that time (LeCun, Bengio, et al., 2015). The intelligence we commonly reference in modern AI systems is, indeed, primarily the ability to learn patterns provided by humans through data. In other words, current AI models do not possess the capability to explore patterns beyond the given data, or even when they do, such explorations are not considered significant discoveries if researchers cannot interpret them (LeCun, 2018). Physics, on the other hand, focuses on decoding phenomena using systems of mathematical equations, discovered through human intelligence and continually validated in pursuit of a unified theory of the universe. As we live in a "physical world" governed by physical laws, understanding physics is crucial for any modeling approach, including neural networks, and as pointed out by one of the pioneers of AI, Yann Le Cun, in a recent interview:

"*Artificial intelligence needs physics to evolve.*"

On the other side, to make hydrological theories more feasible, physically-based models can also benefit from AI's capability to extract and process large datasets (Shen, 2018; Reichstein et al., 2019; Sit et al., 2020), especially in the context of estimating spatially distributed conceptual parameters in data-scarce regions. The question then arises: should we focus on employing complex AI models to partially or fully replace process-based models, or should we prioritize understanding the physical interpretability of hybrid approaches, starting with simpler AI architectures? There is no straightforward answer to this question. However, based on our knowledge of both disciplines and as discussed in Chapter 1, current AI models are not capable of fully replacing process-based models, given the inherent complexity of hydrological processes and uncertainties in modeling from limited observations. This might become feasible if at least the following two conditions are met:

— AI models become more intelligent, producing more interpretable results and demonstrating a comprehensive understanding of physics. However, this is not currently the case, as highlighted by LeCun (2022);

— We obtain nearly perfect and complete hydrological data, accounting for all uncertainties. This requires significant effort and remains a distant goal (Beven, 2019).

I began my journey in hydrology with a background in applied mathematics and machine learning. Over the past few years, my focus has been on leveraging AI's predictive capabilities within a hybrid framework that seamlessly integrates neural networks into process-based numerical models, aiming at enhancing distributed hydrological modeling at high spatio-temporal resolution. The following synthesis reflects my path from discovering hydrology to taking the first steps toward developing a robust hybrid AI approach that emphasizes physical understanding and interpretability:

— The first step was to understand the challenges of calibrating high-resolution hydrological models with spatially distributed conceptual parameters within a VDA framework. The work in Chapter 2 aimed to enhance model performance for flood modeling by integrating multiscale hydrological signatures into a multi-criteria cost function. Additionally, the multi-objective function can serve as a weak constraint to help reduce the non-uniqueness issue. The results showed that, while maintaining acceptable global scores, the model made significant improvements in reproducing flood-scale signatures in both calibration and validation. However, the model still needs stronger constraints on the spatialization of the conceptual parameters, which could be addressed using physical descriptor maps.

— The work in Chapter 3 is motivated by the recognition that conceptual parameters can be explained by several physical descriptors via a regionalization mapping, thereby addressing the challenge of predicting at ungauged sites. The use of neural networks for such a mapping is due to our limited physical knowledge of the conceptual parameters. We expect neural networks to provide a more generic approximation than simple linear regressions to extract and employ spatial patterns in input descriptors. To calibrate the model using gradient-based optimization algorithms, we coupled spatialized gradients of the cost function with respect to the conceptual parameters and combined them with the backpropagation of the neural network. This represents the first study combining a numerical adjoint model and neural network Jacobian, enabling effective high-dimensional optimization using accurate gradients. We evaluated the proposed hybrid method on a large study area containing 125 catchments and covering approximately a quarter of France. The results highlight a strong regionalization performance of the proposed method, especially in the most challenging upstream-to-downstream extrapolation scenario with ANN. We also demonstrated that ANNs can learn a non-linear descriptors-to-parameters mapping, providing better model controllability than a linear mapping for complex calibration scenarios.

— Even with conceptual parameters constrained by physical maps, model calibration still suffers from the non-uniqueness problem and faces the challenge of equifinality (Beven, 2006). The calibrated model parameters are particularly sensitive to the first guess, especially in the case of linear regression. Chapter 4 studied a Bayesian-weighted estimation approach based on Chelil et al. (2022) for estimating a spatially uniform first guess (LD Bayesian-weighted estimate) and then generalized it for a spatially distributed one (HD Bayesian-weighted estimate). Tests performed using cross-temporal calibration showed improved stability of the conceptual parameters in the case of linear regression, while ANNs exhibited greater stability despite starting from random parameter initialization.

— Based on the findings in the previous works, Chapter 5 focused on hybridizing a differentiable hydrological model with neural networks, leveraging the temporal memory effect of the original model on top of a differentiable kinematic wave routing over a flow direction grid. In addition to the regionalization neural network, we introduced a second ANN into the forward hydrological model to refine internal water fluxes. We evaluated flood modeling performance and analyzed the interpretability of learned conceptual parameters and corrections of internal fluxes using two high-resolution datasets. The first dataset involved 235 catchments in France, used for local calibration-validation and model structure comparisons between the classical GR-like model and the hybrid approach. The second dataset presented

a challenging multi-catchment modeling setup in flash flood-prone areas to demonstrate the framework's regionalization learning capabilities. The results showed that the hybrid models achieved superior accuracy and robustness compared to classical approaches in both spatial and temporal validation. Analysis of the spatially distributed parameters and internal fluxes revealed the hybrid models' nuanced behavior and enhanced model versatility, demonstrating the potential of hybrid physics-AI approaches for uncovering physical processes.

— Building on the promising results in Chapter 5, Chapter 6 further enhanced the model's versatility by incorporating neural networks into a dynamic state-space system, solved using an implicit Euler scheme. This state-space model solves the ODEs governing reservoir dynamics, while embedding an ANN to refine internal fluxes, all without relying on an analytical solution, instead computing the model states simultaneously. Results suggested that hybrid approaches demonstrated consistently strong and stable performance in both calibration and various validation scenarios. Additionally, the neural ODE structure exhibited a hybridization effect that modifies state dynamics and runoff flow, achieving more reliable streamflow simulations in flood contexts.

Another aspect of my thesis involves contributing to the maintenance and development of `smash` (Colleoni, Huynh, et al., 2025), which is now an open-source platform. It has been implemented at Service Central VigiCrues (SCHAPI) as the core solver for the current French flash flood forecasting system, VigiCrues Flash, and for the upcoming higher-resolution flood warning platform. Our work has focused on producing efficient and reproducible code, along with comprehensive documentation that includes an API reference, a user guide, and detailed explanations of the mathematical models and algorithms used. This documentation helps users understand and utilize the platform effectively, from basic to advanced features, demonstrating its suitability and efficiency for operational applications. Current efforts aim to upgrade the code from version 1.0 to 1.1, with an emphasis on developing and testing various hybrid models on large datasets. Additionally, recent work by Ettalbi, Garambois, Huynh, Ferreira, et al. (2025) has developed a GPU-enabled cell-to-cell routing algorithm, which enables fast and efficient computational processing of `smash` across large spatial domains. This represents an ongoing direction for software development.

For future research paths, we will continue to focus on the expansion of the hybrid physics-AI framework and the refinement of data assimilation techniques (Huynh, Garambois, Renard, and Monnier, 2025). Short-term goals include:

— studying the generalizability of structural corrections with neural networks across larger datasets and diverse model structures;

— integrating probabilistic frameworks such as Bayesian inference to quantify the uncertainties surrounding the application of the hybrid models (e.g., Renard, Garreta, et al. (2006));

— exploring mathematical properties, such as equifinality issues between neural networks and conceptual parameters, and analyzing the response surfaces of universal differential equation sets for flexible hydrological modeling in time and space.

The long-term direction is to achieve a broader coverage of neural networks in the modeling process. This integration can be done within a coupled differentiable river network hydraulic model to improve 1D-2D hydrodynamic realism (Garambois, 2025). The coupling will enable feedback by assimilating hydraulic observations into a differentiable H&H chain (hydraulic-hydrological assimilation chain) (Pujol, Garambois, and Monnier, 2022), such as the unprecedented hydraulic visibility (Garambois, Calmant, et al., 2017) brought by SWOT (Biancamaria et al., 2016) and multi-satellite data (e.g., with VDA in Pujol, Garambois, Finaud-Guyot, et al. (2020), Malou et al. (2021), and Larnier, Garambois, et al. (2025)). Such differentiable and learnable H&H modeling frameworks are expected to enhance the representation of basin internal state fluxes and enable the efficient fusion of machine learning with process-based modeling, advancing the discovery of scale-relevant hydrological laws through the maximal extraction of information from multi-source data.

References

Abbott, M., J. Bathurst, J. Cunge, P. O'Connell, and J. Rasmussen (1986). "An introduction to the European Hydrological System — Systeme Hydrologique Europeen, "SHE", 1: History and philosophy of a physically-based, distributed modelling system". In: *Journal of Hydrology* 87.1, pp. 45–59. DOI: 10.1016/0022-1694(86)90114-9.

Abdulla, F. A. and D. P. Lettenmaier (1997). "Development of regional parameter estimation equations for a macroscale hydrologic model". In: *Journal of hydrology* 197.1-4, pp. 230–257. DOI: 10.1016/S0022-1694(96)03262-3.

Agency, E. E. (2016). *European Digital Elevation Model (EU-DEM)*. Version 1.0. URL: https://sdi.eea.europa.eu/catalogue/srv/api/records/19cff95e-61ac-45ed-8ee3-c43220d709cf.

— (2019). *CORINE Land Cover 2012 (raster 100 m)*. Version 20.01. DOI: 10.2909/a84ae124-c5c5-4577-8e10-511bfe55cc0d.

Alipour, A., A. Ahmadalipour, and H. Moradkhani (2020). "Assessing flash flood hazard and damages in the southeast United States". In: *Journal of Flood Risk Management* 13.2, e12605. DOI: 10.1111/jfr3.12605.

Althoff, D., L. N. Rodrigues, and D. D. da Silva (2021). "Addressing hydrological modeling in watersheds under land cover change with deep learning". In: *Advances in Water Resources* 154, p. 103965. DOI: 10.1016/j.advwatres.2021.103965.

Arsenault, R., A. Poulin, P. Côté, and F. Brissette (2014). "Comparison of Stochastic Optimization Algorithms in Hydrological Model Calibration". In: *Journal of Hydrologic Engineering* 19.7, pp. 1374–1384. DOI: 10.1061/(ASCE)HE.1943-5584.0000938.

Assran, M., Q. Duval, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, et al. (2023). "Self-Supervised Learning From Images With a Joint-Embedding Predictive Architecture". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15619–15629.

Astagneau, P. C., F. Bourgin, V. Andréassian, and C. Perrin (2022). "Catchment response to intense rainfall: Evaluating modelling hypotheses". In: *Hydrological Processes* 36.8, e14676. DOI: 10.1002/hyp.14676.

Astagneau, P. C., F. Bourgin, V. Andréassian, and C. P. and (2024). "Improving the structure of a hydrological model to forecast catchment response to intense rainfall". In: *LHB* 110.1, p. 2341027. DOI: 10.1080/27678490.2024.2341027.

Astagneau, P. C., F. Bourgin, V. Andréassian, and C. Perrin (2021). "When does a parsimonious model fail to simulate floods? Learning from the seasonality of model bias". In: *Hydrological Sciences Journal* 66.8, pp. 1288–1305. DOI: 10.1080/02626667.2021.1923720.

Azzini, I., T. A. Mara, and R. Rosati (2021). "Comparison of two sets of Monte Carlo estimators of Sobol'indices". In: *Environmental Modelling & Software* 144, p. 105167. DOI: 10.1016/j.envsoft.2021.105167.

Bastola, S., H. Ishidaira, and K. Takeuchi (2008). "Regionalisation of hydrological model parameters under parameter uncertainty: A case study involving TOPMODEL and basins across the globe". In: *Journal of Hydrology* 357.3, pp. 188–206. DOI: 10.1016/j.jhydrol.2008.05.007.

Beard, L. R. (1962). *Statistical methods in hydrology*. Army Engineer District, Sacramento.

Beck, H. E., A. I. J. M. van Dijk, A. de Roo, D. G. Miralles, T. R. McVicar, J. Schellekens, et al. (2016). "Global-scale regionalization of hydrologic model parameters". In: *Water Resources Research* 52.5, pp. 3599–3622. DOI: 10.1002/2015WR018247.

Beck, H. E., M. Pan, P. Lin, J. Seibert, A. I. J. M. van Dijk, and E. F. Wood (2020). "Global fully distributed parameter regionalization based on observed streamflow from 4,229 headwater catchments". In: *Journal of Geophysical Research: Atmospheres* 125.17, e2019JD031485. DOI: 10.1029/2019JD031485.

Beck, M. B. (1987). "Water quality modeling: a review of the analysis of uncertainty". In: *Water resources research* 23.8, pp. 1393–1442. DOI: 10.1029/WR023i008p01393.

Bergström, S. (1976). *Development and application of a conceptual runoff model for Scandinavian catchments*.

Berne, A. and W. Krajewski (2013). "Radar for hydrology: Unfulfilled promise or unrecognized potential?" In: *Advances in Water Resources* 51. 35th Year Anniversary Issue, pp. 357–366. DOI: 10.1016/j.advwatres.2012.05.005.

Beven, K. (1987). "Towards a new paradigm in hydrology". In: *IN: Water for the Future: Hydrology in Perspective. IAHS Publication* 164.

— (1989). "Changing ideas in hydrology — The case of physically-based models". In: *Journal of Hydrology* 105.1, pp. 157–172. DOI: 10.1016/0022-1694(89)90101-7.

— (1993). "Prophecy, reality and uncertainty in distributed hydrological modelling". In: *Advances in water resources* 16.1, pp. 41–51. DOI: 10.1016/0309-1708(93)90028-E.

— (2001). "How far can we go in distributed hydrological modelling?" In: *Hydrology and Earth System Sciences* 5.1, pp. 1–12. DOI: 10.5194/hess-5-1-2001.

— (2002). "Towards an alternative blueprint for a physically based digitally simulated hydrologic response modelling system". In: *Hydrological Processes* 16.2, pp. 189–206. DOI: 10.1002/hyp.343.

— (2006). "A manifesto for the equifinality thesis". In: *Journal of Hydrology* 320.1. The model parameter estimation experiment, pp. 18–36. DOI: 10.1016/j.jhydrol.2005.07.007.

— (2012). "Data for Rainfall–Runoff Modelling". In: *Rainfall-Runoff Modelling*. John Wiley & Sons, Ltd. Chap. 3, pp. 51–82. DOI: 10.1002/9781119951001.ch3.

— (2019). "How to make advances in hydrological modelling". In: *Hydrology Research* 50.6, pp. 1481–1494. DOI: 10.2166/nh.2019.134.

— (2020). "Deep learning, hydrological processes and the uniqueness of place". In: *Hydrological Processes* 34.16, pp. 3608–3613. DOI: 10.1002/hyp.13805.

Beven, K. and M. 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.

Biancamaria, S., D. P. Lettenmaier, and T. M. Pavelsky (2016). "The SWOT Mission and Its Capabilities for Land Hydrology". In: *Surveys in Geophysics* 37.2, pp. 307–337. DOI: 10.1007/s10712-015-9346-y.

Blank, J. and K. Deb (2020). "pymoo: Multi-Objective Optimization in Python". In: *IEEE Access* 8, pp. 89497–89509. DOI: 10.1109/ACCESS.2020.2990567.

Blöschl, G., M. Sivapalan, T. Wagener, H. Savenije, and A. Viglione (2013). *Runoff prediction in ungauged basins: synthesis across processes, places and scales*. Cambridge University Press.

Bout, V. and V. Jetten (2018). "The validity of flow approximations when simulating catchment-integrated flash floods". In: *Journal of hydrology* 556, pp. 674–688. DOI: 10.1016/j.jhydrol.2017.11.033.

Bouttier, F. and P. Courtier (2002). "Data assimilation concepts and methods March 1999". In: *Meteorological training course lecture series. ECMWF* 718, p. 59.

Brigode, P., B. Génot, F. Lobligeois, and O. Delaigue (2020). *Summary sheets of watershed-scale hydroclimatic observed data for France*. Version V1. Portail Data INRAE. DOI: 10.15454/UV01P1.

Brown, T. B., B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, et al. (2020). "Language models are few-shot learners". In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. NIPS '20. Vancouver, BC, Canada: Curran Associates Inc.

Burn, D. H. (1997). "Catchment similarity for regional flood frequency analysis using seasonality measures". In: *Journal of Hydrology* 202.1, pp. 212–230. DOI: 10.1016/S0022-1694(97)00068-1.

Campolongo, F., A. Saltelli, and J. Cariboni (2011). "From screening to quantitative sensitivity analysis. A unified approach". In: *Computer physics communications* 182.4, pp. 978–988. DOI: 10.1016/j.cpc.2010.12.039.

Caruso, A., A. Guillot, and P. Arnaud (2013). "Notice sur les indices de confiance de la méthode Shyreg-Débit–Définitions et Calculs". In: *Aix en Provence: IRSTEA, Convention DGPR/SNRH*.

Castaings, W., D. Dartus, F.-X. Le Dimet, and G.-M. Saulnier (2009). "Sensitivity analysis and parameter estimation for distributed hydrological modeling: potential of variational methods". In: *Hydrology and Earth System Sciences* 13.4, pp. 503–517. DOI: 10.5194/hess-13-503-2009.

Champeaux, J.-L., P. Dupuy, O. Laurantin, I. Soulan, P. Tabary, and J.-M. Soubeyroux (2009). "Les mesures de précipitations et l'estimation des lames d'eau à Météo-France: état de l'art et perspectives". In: *La Houille Blanche* 5, pp. 28–34. DOI: 10.1051/lhb/2009052.

Chelil, S., H. Oubanas, H. Henine, I. Gejadze, P. O. Malaterre, and J. Tournebize (2022). "Variational Data Assimilation to improve subsurface drainage model parameters". In: *Journal of Hydrology*, p. 128006. DOI: 10.1016/j.jhydrol.2022.128006.

Chen, R. T. Q., Y. Rubanova, J. Bettencourt, and D. Duvenaud (2019). *Neural Ordinary Differential Equations*. DOI: 10.48550/arXiv.1806.07366.

Cho, K. and Y. Kim (2022). "Improving streamflow prediction in the WRF-Hydro model with LSTM networks". In: *Journal of Hydrology* 605, p. 127297. DOI: 10.1016/j.jhydrol.2021.127297.

Clark, M. P., A. G. Slater, D. E. Rupp, R. A. Woods, J. A. Vrugt, H. V. Gupta, et al. (2008). "Framework for Understanding Structural Errors (FUSE): A modular framework to diagnose differences between hydrological models". In: *Water Resources Research* 44.12. DOI: 10.1029/2007WR006735.

Colleoni, F., N. N. T. Huynh, P.-A. Garambois, M. Jay-Allemand, D. Organde, B. Renard, et al. (2025). "SMASH v1.0: A Differentiable and Regionalizable High-Resolution Hydrological Modeling and Data Assimilation Framework". In: *EGUsphere* 2025, pp. 1–36. DOI: 10.5194/egusphere-2025-690.

Colleoni, F., P.-A. Garambois, P. Javelle, M. Jay-Allemand, and P. Arnaud (2022). "Adjoint-based spatially distributed calibration of a grid GR-based parsimonious hydrological model over 312 French catchments with SMASH platform". In: *EGUsphere* 2022, pp. 1–37. DOI: 10.5194/egusphere-2022-506.

Crawford, N. H. and R. K. Linsley (1966). "Digital Simulation in Hydrology'Stanford Watershed Model 4". In.

De Lavenne, A., V. Andréassian, G. Thirel, M.-H. Ramos, and C. Perrin (2019). "A regularization approach to improve the sequential calibration of a semidistributed hydrological model". In: *Water Resources Research* 55.11, pp. 8821–8839. DOI: 10.1029/2018WR024266.

Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan (2002). "A fast and elitist multiobjective genetic algorithm: NSGA-II". In: *IEEE transactions on evolutionary computation* 6.2, pp. 182–197. DOI: 10.1109/4235.996017.

Delaigue, O., B. Génot, L. Lebecherel, P. Brigode, and P.-Y. Bourgin (2020). *Database of watershed-scale hydroclimatic observations in France*. URL: https://webgr.inrae.fr/base-de-donnees.

Dennis Jr., J. E. and J. J. Moré (1977). "Quasi-Newton Methods, Motivation and Theory". In: *SIAM Review* 19.1, pp. 46–89. DOI: 10.1137/1019005.

Ding, Q. and P. Arnaud (2024). "Using supervised machine learning for regional hydrological hazard estimation in metropolitan France". In: *Journal of Hydrology: Regional Studies* 54, p. 101872. DOI: 10.1016/j.ejrh.2024.101872.

Dooge, J. (1973). *Linear theory of hydrologic systems*. 1468. Agricultural Research Service, US Department of Agriculture.

— (1986). "Looking for hydrologic laws". In: *Water Resources Research* 22.9S, 46S–58S. DOI: 10.1029/WR022i09Sp0046S.

Douinot, A., H. Roux, P.-A. Garambois, and D. Dartus (2018). "Using a multi-hypothesis framework to improve the understanding of flow dynamics during flash floods". In: *Hydrology and Earth System Sciences* 22.10, pp. 5317–5340. DOI: 10.5194/hess-22-5317-2018.

Duan, Q., S. Sorooshian, and V. Gupta (1992). "Effective and efficient global optimization for conceptual rainfall-runoff models". In: *Water Resources Research* 28.4, pp. 1015–1031. DOI: 10.1029/91WR02985.

Duarte, M. and R. N. Watanabe (2021). *Notes on Scientific Computing for Biomechanics and Motor Control*. Version v0.0.2. DOI: 10.5281/zenodo.4599319.

Durand, Y., M. Laternser, G. Giraud, P. Etchevers, B. Lesaffre, and L. Mérindol (2009). "Reanalysis of 44 yr of climate in the French Alps (1958–2002): Methodology, model validation, climatology, and trends for air temperature and precipitation". In: *Journal of Applied Meteorology and Climatology* 48.3, pp. 429–449. DOI: 10.1175/2008JAMC1808.1.

Elman, J. L. (1990). "Finding structure in time". In: *Cognitive Science* 14.2, pp. 179–211. DOI: 10.1016/0364-0213(90)90002-E.

Emmanuel, I., H. Andrieu, E. Leblois, N. Janey, and O. Payrastre (2015). "Influence of rainfall spatial variability on rainfall–runoff modelling: benefit of a simulation approach?" In: *Journal of hydrology* 531, pp. 337–348. DOI: 10.1016/j.jhydrol.2015.04.058.

Ettalbi, M., P.-A. Garambois, N. N. T. Huynh, P. Arnaud, E. Ferreira, and N. Baghdadi (2025). "Improving parameter regionalization learning for spatialized differentiable hydrological models by assimilation of satellite-based soil moisture data". In: *Journal of Hydrology* 660, p. 133300. DOI: 10.1016/j.jhydrol.2025.133300.

Ettalbi, M., P.-A. Garambois, N. N. T. Huynh, E. Ferreira, and N. Baghdadi (2025). "GPU-Enabled Cell-to-Cell Routing in a High Resolution Hybrid Distributed Hydrological Model with Multi-Source Remote Sensing Data Assimilation: A Continental-Scale Computational Approach". In: *EGU General Assembly 2025*. DOI: 10.5194/egusphere-egu25-9176.

Fablet, R., B. Chapron, L. Drumetz, E. Mémin, O. Pannekoucke, and F. Rousseau (2021). "Learning variational data assimilation models and solvers". In: *Journal of Advances in Modeling Earth Systems* 13.10, e2021MS002572. DOI: 10.1029/2021MS002572.

Fekete, B. M. and C. J. Vörösmarty (2007). "The current status of global river discharge monitoring and potential new technologies complementing traditional discharge measurements". In: *IAHS publ* 309, pp. 129–136.

Feng, D., K. Fang, and C. 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. DOI: 10.1029/2019WR026793.

Feng, D., J. Liu, K. Lawson, and C. 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. DOI: 10.1029/2022WR032404.

Ficchì, A., C. Perrin, and V. Andréassian (2019). "Hydrological modelling at multiple sub-daily time steps: Model improvement via flux-matching". In: *Journal of Hydrology* 575, pp. 1308–1327. DOI: 10.1016/j.jhydrol.2019.05.084.

Finke, P., R. Hartwich, R. Dudal, J. Ibanez, M. Jamagne, D. King, et al. (1998). *Geo-referenced soil database for Europe. Manual of procedures, version 1.0*. European Communities.

Fleischmann, A. S., R. C. D. Paiva, W. Collischonn, V. A. Siqueira, A. Paris, D. M. Moreira, et al. (2020). "Trade-Offs Between 1-D and 2-D Regional River Hydrodynamic Models". In: *Water Resources Research* 56.8, e2019WR026812. DOI: 10.1029/2019WR026812.

Flipo, N., N. Gallois, and J. Schuite (2023). "Regional coupled surface–subsurface hydrological model fitting based on a spatially distributed minimalist reduction of frequency domain discharge data". In: *Geoscientific Model Development* 16.1, pp. 353–381. DOI: 10.5194/gmd-16-353-2023.

Fortin, F.-A., F.-M. D. Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné (2012). "DEAP: Evolutionary Algorithms Made Easy". In: *Journal of Machine Learning Research* 13.70, pp. 2171–2175.

Frame, J. M., F. Kratzert, A. Raney II, M. Rahman, F. R. Salas, and G. Nearing (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.

Freeze, R. and R. Harlan (1969). "Blueprint for a physically-based, digitally-simulated hydrologic response model". In: *Journal of Hydrology* 9.3, pp. 237–258. DOI: 10.1016/0022-1694(69)90020-1.

Garambois, P.-A., K. Larnier, J. Monnier, P. Finaud-Guyot, J. Verley, A.-S. Montazem, et al. (2020). "Variational estimation of effective channel and ungauged anabranching river discharge from multi-satellite water heights of different spatial sparsity". In: *Journal of Hydrology* 581, p. 124409. DOI: 10.1016/j.jhydrol.2019.124409.

Garambois, P.-A. (Mar. 2025). "Étude d'écoulements en hydrologie et hydraulique de surface par modélisation numérique et assimilation de données multi-sources". Habilitation à diriger des recherches. URL: https://hal.science/tel-05015362.

Garambois, P.-A., S. Calmant, H. Roux, A. Paris, J. Monnier, P. Finaud-Guyot, et al. (2017). "Hydraulic visibility: Using satellite altimetry to parameterize a hydraulic model of an ungauged reach of a braided river". In: *Hydrological Processes* 31.4, pp. 756–767. DOI: 10.1002/hyp.11033.

Garambois, P.-A., K. Larnier, H. Roux, D. Labat, and D. Dartus (2014). "Analysis of flash flood-triggering rainfall for a process-oriented hydrological model". In: *Atmospheric research* 137, pp. 14–24. DOI: 10.1016/j.atmosres.2013.09.016.

Garambois, P.-A. and J. Monnier (2015). "Inference of effective river properties from remotely sensed observations of water surface". In: *Advances in Water Resources* 79, pp. 103–120. DOI: 10.1016/j.advwatres.2015.02.007.

Garambois, P.-A., H. Roux, K. Larnier, W. Castaings, and D. Dartus (2013). "Characterization of process-oriented hydrologic model behavior with temporal sensitivity analysis for flash floods in Mediterranean catchments". In: *Hydrology and Earth System Sciences*. DOI: 10.5194/hess-17-2305-2013.

Garambois, P.-A., H. Roux, K. Larnier, D. Labat, and D. Dartus (2015a). "Characterization of catchment behaviour and rainfall selection for flash flood hydrological model calibration: catchments of the eastern Pyrenees". In: *Hydrological sciences journal* 60.3, pp. 424–447. DOI: 10.1080/02626667.2014.909596.

— (2015b). "Parameter regionalization for a process-oriented distributed model dedicated to flash floods". In: *Journal of Hydrology* 525, pp. 383–399. DOI: 10.1016/j.jhydrol.2015.03.052.

Gaume, E., V. Bain, P. Bernardara, O. Newinger, M. Barbuc, A. Bateman, et al. (2009). "A compilation of data on European flash floods". In: *Journal of Hydrology* 367.1, pp. 70–78. DOI: 10.1016/j.jhydrol.2008.12.028.

Gejadze, I., P.-O. Malaterre, H. Oubanas, and V. Shutyaev (2022). "A new robust discharge estimation method applied in the context of SWOT satellite data processing". In: *Journal of Hydrology* 610, p. 127909. DOI: 10.1016/j.jhydrol.2022.127909.

Glorot, X. and Y. Bengio (2010). "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Y. W. Teh and M. Titterington. Vol. 9. Proceedings of Machine Learning Research. Chia Laguna Resort, Sardinia, Italy: PMLR, pp. 249–256. URL: https://proceedings.mlr.press/v9/glorot10a.html.

Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep Learning*. MIT Press. URL: http://www.deeplearningbook.org.

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, et al. (2020). "Generative adversarial networks". In: *Commun. ACM* 63.11, pp. 139–144. DOI: 10.1145/3422622.

Gordon, R. L. (1989). "Acoustic Measurement of River Discharge". In: *Journal of Hydraulic Engineering* 115.7, pp. 925–936. DOI: 10.1061/(ASCE)0733-9429(1989)115:7(925).

Götzinger, J. and A. Bárdossy (2007). "Comparison of four regionalisation methods for a distributed hydrological model". In: *Journal of Hydrology* 333.2-4, pp. 374–384. DOI: 10.1016/j.jhydrol.2006.09.008.

Gupta, H., K. Beven, and T. Wagener (2006). "Model calibration and uncertainty estimation". In: *Encyclopedia of hydrological sciences*. DOI: 10.1002/0470848944.hsa138.

Gupta, H., H. Kling, K. K. Yilmaz, and G. F. Martinez (2009). "Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling". In: *Journal of hydrology* 377.1-2, pp. 80–91. DOI: 10.1016/j.jhydrol.2009.08.003.

Gupta, H. and S. Razavi (2018). "Revisiting the basis of sensitivity analysis for dynamical earth system models". In: *Water Resources Research* 54.11, pp. 8692–8717. DOI: 10.1029/2018WR022668.

Hannah, D. M., S. Demuth, H. A. van Lanen, U. Looser, C. Prudhomme, G. Rees, et al. (2011). "Large-scale river flow archives: importance, current status and future needs". In: *Hydrological Processes* 25.7, pp. 1191–1200. DOI: 10.1002/hyp.7794.

Hascoet, L. and V. Pascual (2013). "The Tapenade automatic differentiation tool: principles, model, and specification". In: *ACM Transactions on Mathematical Software (TOMS)* 39.3, pp. 1–43. DOI: 10.1145/2450153.2450158.

Hashemi, R., P. Brigode, P.-A. Garambois, and P. Javelle (2022). "How can we benefit from regime information to make more effective use of long short-term memory (LSTM) runoff models?" In: *Hydrology and Earth System Sciences* 26.22, pp. 5793–5816. DOI: 10.5194/hess-26-5793-2022.

He, B., X. Huang, M. Ma, Q. Chang, Y. Tu, Q. Li, et al. (2018). "Analysis of flash flood disaster characteristics in China from 2011 to 2015". In: *Natural Hazards* 90, pp. 407–420. DOI: 10.1007/s11069-017-3052-7.

He, K., X. Zhang, S. Ren, and J. Sun (2016). "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

Herman, J. and W. Usher (2017). "SALib: An open-source Python library for Sensitivity Analysis". In: *The Journal of Open Source Software* 2.9. DOI: 10.21105/joss.00097.

Hersbach, H., B. Bell, P. Berrisford, S. Hirahara, A. Horányi, J. Muñoz-Sabater, 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.

Hettiarachchi, S., C. Wasko, and A. Sharma (2018). "Increase in flood risk resulting from climate change in a developed urban watershed - the role of storm temporal patterns". In: *Hydrology and Earth System Sciences* 22.3, pp. 2041–2056. DOI: 10.5194/hess-22-2041-2018.

Hochreiter, S. (1998). "The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 06.02, pp. 107–116. DOI: 10.1142/S0218488598000094.

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

Hoffman, M. D. and A. Gelman (2014). "The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo". In: *Journal of Machine Learning Research* 15, pp. 1593–1623.

Höge, 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 Discussions*, pp. 1–29. DOI: 10.5194/hess-26-5085-2022.

Horner, I., B. Renard, J. Le Coz, F. Branger, H. K. McMillan, and G. Pierrefeu (2018). "Impact of Stage Measurement Errors on Streamflow Uncertainty". In: *Water Resources Research* 54.3, pp. 1952–1976. DOI: 10.1002/2017WR022039.

Horner, I. (2020). "Design and Evaluation of hydrological signatures for the diagnosis and improvement of a process-based distributed hydrological model". PhD thesis. Université Grenoble Alpes.

Hrachowitz, M., O. Fovet, L. Ruiz, T. Euser, S. Gharari, R. Nijzink, et al. (2014). "Process consistency in models: The importance of system signatures, expert knowledge, and process complexity". In: *Water Resources Research* 50.9, pp. 7445–7469. DOI: 10.1002/2014WR015484.

Hrachowitz, M., H. Savenije, G. Blöschl, J. McDonnell, M. Sivapalan, J. Pomeroy, et al. (2013). "A decade of Predictions in Ungauged Basins (PUB)—a review". In: *Hydrological Sciences Journal* 58.6, pp. 1198–1255. DOI: 10.1080/02626667.2013.803183.

Hsu, K.-l., H. V. Gupta, and S. Sorooshian (1995). "Artificial Neural Network Modeling of the Rainfall-Runoff Process". In: *Water Resources Research* 31.10, pp. 2517–2530. DOI: 10.1029/95WR01955.

Huang, S., S. Eisner, J. O. Magnusson, C. Lussana, X. Yang, and S. Beldring (2019). "Improvements of the spatially distributed hydrological modelling using the HBV model at 1 km resolution for Norway". In: *Journal of Hydrology* 577, p. 123585. DOI: 10.1016/j.jhydrol.2019.03.051.

Huffman, G. J., D. T. Bolvin, D. Braithwaite, K.-L. Hsu, R. J. Joyce, C. Kidd, et al. (2020). "Integrated Multi-satellite Retrievals for the Global Precipitation Measurement (GPM) Mission (IMERG)". In: *Satellite Precipitation Measurement: Volume 1*. Ed. by V. Levizzani, C. Kidd, D. B. Kirschbaum, C. D. Kummerow, K. Nakamura, and F. J. Turk. Cham: Springer International Publishing, pp. 343–353. DOI: 10.1007/978-3-030-24568-9_19.

Hundecha, Y. and A. Bárdossy (2004). "Modeling of the effect of land use changes on the runoff generation of a river basin through parameter regionalization of a watershed model". In: *Journal of hydrology* 292.1-4, pp. 281–295. DOI: 10.1016/j.jhydrol.2004.01.002.

Huynh, N. N. T., P.-A. Garambois, B. Renard, F. Colleoni, J. Monnier, and H. Roux (2025). "A Distributed Hybrid Physics-AI Framework for Learning Corrections of Internal Hydrological Fluxes and Enhancing High-Resolution Regionalized Flood Modeling". In: *EGUsphere* 2025, pp. 1–34. DOI: 10.5194/egusphere-2024-3665.

Huynh, N. N. T., P.-A. Garambois, F. Colleoni, and P. Javelle (2023). "Signatures-and-sensitivity-based multi-criteria variational calibration for distributed hydrological modeling applied to Mediterranean floods". In: *Journal of Hydrology* 625, p. 129992. DOI: 10.1016/j.jhydrol.2023.129992.

Huynh, N. N. T., P.-A. Garambois, F. Colleoni, B. Renard, J. Monnier, and H. Roux (2024). "Multiscale Learnable Physical Modeling and Data Assimilation Framework: Application to High-Resolution Regionalized Hydrological Simulation of Flash Flood". In: *Authorea Preprints*. DOI: 10.22541/au.170709054.44271526/v2.

Huynh, N. N. T., P.-A. Garambois, F. Colleoni, B. Renard, and H. Roux (2023). "Multi-gauge Hydrological Variational Data Assimilation: Regionalization Learning with Spatial Gradients

using Multilayer Perceptron and Bayesian-Guided Multivariate Regression". In: *arXiv*. DOI: 10.48550/arXiv.2307.02497.

Huynh, N. N. T., P.-A. Garambois, F. Colleoni, B. Renard, H. Roux, J. Demargne, et al. (2024). "Learning Regionalization Using Accurate Spatial Cost Gradients Within a Differentiable High-Resolution Hydrological Model: Application to the French Mediterranean Region". In: *Water Resources Research* 60.11, e2024WR037544. DOI: 10.1029/2024WR037544.

Huynh, N. N. T., P.-A. Garambois, B. Renard, and J. Monnier (2025). "A General Framework for Integrating Neural Networks into Numerical Resolution Methods for Spatially Distributed Hydrological Models". In: *EGU General Assembly 2025*. DOI: 10.5194/egusphere-egu25-2740.

Iwanaga, T., W. Usher, and J. Herman (2022). "Toward SALib 2.0: Advancing the accessibility and interpretability of global sensitivity analyses". In: *Socio-Environmental Systems Modelling* 4, p. 18155. DOI: 10.18174/sesmo.18155.

Javelle, P., C. Saint-Martin, L. Garandeau, and B. Janet (2019). "Flash flood warnings: Recent achievements in France with the national Vigicrues Flash system". In: *United Nations Office for Disaster Risk Reduction, Contributing Paper to the Global Assessment Report on Disaster Risk Reduction (GAR 2019)* 60.

Jay-Allemand, M., J. Demargne, P.-A. Garambois, P. Javelle, I. Gejadze, F. Colleoni, et al. (2024). "Spatially distributed calibration of a hydrological model with variational optimization constrained by physiographic maps for flash flood forecasting in France". In: *Proceedings of IAHS* 385, pp. 281–290. DOI: 10.5194/piahs-385-281-2024.

Jay-Allemand, M. (2020). "Estimation variationnelle des paramètres d'un modèle hydrologique distribué". PhD thesis. Aix-Marseille.

Jay-Allemand, M., P. Javelle, I. Gejadze, P. Arnaud, P.-O. Malaterre, J.-A. Fine, et al. (2020). "On the potential of variational calibration for a fully distributed hydrological model: application on a Mediterranean catchment". In: *Hydrology and Earth System Sciences* 24.11, pp. 5519–5538. DOI: 10.5194/hess-24-5519-2020.

Jiang, S., Y. Zheng, and D. Solomatine (2020). "Improving AI system awareness of geoscience knowledge: Symbiotic integration of physical approaches and deep learning". In: *Geophysical Research Letters* 47.13, e2020GL088229. DOI: 10.1029/2020GL088229.

Kalman, R. E. (1960). "A New Approach to Linear Filtering and Prediction Problems". In: *Journal of Basic Engineering* 82.1, pp. 35–45. DOI: 10.1115/1.3662552.

Kavetski, D., G. Kuczera, and S. W. Franks (2006). "Bayesian analysis of input uncertainty in hydrological modeling: 1. Theory". In: *Water resources research* 42.3. DOI: 10.1029/2005WR004368.

Kennedy, E. (1984). *Discharge ratings at gaging stations*. DOI: 10.3133/twri03A10.

Kermode, J. R. (2020). "f90wrap: an automated tool for constructing deep Python interfaces to modern Fortran codes". In: *J. Phys. Condens. Matter*. DOI: 10.1088/1361-648X/ab82d2.

Kiang, J. E., C. Gazoorian, H. McMillan, G. Coxon, J. Le Coz, I. K. Westerberg, et al. (2018). "A Comparison of Methods for Streamflow Uncertainty Estimation". In: *Water Resources Research* 54.10, pp. 7149–7176. DOI: 10.1029/2018WR022708.

Kidd, C., A. Becker, G. J. Huffman, C. L. Muller, P. Joe, G. Skofronick-Jackson, et al. (2017). "So, How Much of the Earth's Surface Is Covered by Rain Gauges?" In: *Bulletin of the American Meteorological Society* 98.1, pp. 69–78. DOI: 10.1175/BAMS-D-14-00283.1.

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

Kirchner, J. 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.

Kirkby, M. and A. K. Biswas (1971). "History of Hydrology". In: *The Geographical Journal* 137.2, p. 243.

Kirstetter, G., O. Delestre, P.-Y. Lagrée, S. Popinet, and C. Josserand (2021). "B-flood 1.0: an open-source Saint-Venant model for flash-flood simulation using adaptive refinement". In: *Geoscientific Model Development* 14.11, pp. 7117–7132. DOI: 10.5194/gmd-14-7117-2021.

Kollet, S. J. and R. M. Maxwell (2006). "Integrated surface–groundwater flow modeling: A free-surface overland flow boundary condition in a parallel groundwater flow model". In: *Advances in Water Resources* 29.7, pp. 945–958. DOI: 10.1016/j.advwatres.2005.08.006.

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, F., D. Klotz, M. Herrnegger, A. K. Sampson, S. Hochreiter, and G. Nearing (2019). "Toward Improved Predictions in Ungauged Basins: Exploiting the Power of Machine Learning". In: *Water Resources Research* 55.12, pp. 11344–11354. DOI: 10.1029/2019WR026065.

Kratzert, F., G. Nearing, N. Addor, T. Erickson, M. Gauch, O. Gilon, et al. (2023). "Caravan-A global community dataset for large-sample hydrology". In: *Scientific Data* 10.1, p. 61. DOI: 10.1038/s41597-023-01975-w.

Krizhevsky, A., I. Sutskever, and G. E. Hinton (2012). "ImageNet Classification with Deep Convolutional Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira, C. Burges, L. Bottou, and K. Weinberger. Vol. 25. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Kuang, X., J. Liu, B. R. Scanlon, J. J. Jiao, S. Jasechko, M. Lancia, et al. (2024). "The changing nature of groundwater in the global water cycle". In: *Science* 383.6686, eadf0630. DOI: 10.1126/science.adf0630.

Kuczera, G., B. Renard, M. Thyer, and D. Kavetski (2010). "There are no hydrological monsters, just models and observations with large uncertainties!" In: *Hydrological Sciences Journal* 55.6, pp. 980–991. DOI: 10.1080/02626667.2010.504677.

Kumanlioglu, A. A. and O. Fistikoglu (2019). "Performance Enhancement of a Conceptual Hydrological Model by Integrating Artificial Intelligence". In: *Journal of Hydrologic Engineering* 24.11, p. 04019047. DOI: 10.1061/(ASCE)HE.1943-5584.0001850.

Laloy, E. and J. A. Vrugt (2012). "High-dimensional posterior exploration of hydrologic models using multiple-try DREAM(ZS) and high-performance computing". In: *Water Resources Research* 48.1. DOI: 10.1029/2011wr010608.

Lamboni, M., B. Iooss, A.-L. Popelin, and F. Gamboa (2013). "Derivative-based global sensitivity measures: General links with Sobol'indices and numerical tests". In: *Mathematics and Computers in Simulation* 87, pp. 45–54. DOI: 10.1016/j.sbspro.2010.05.208.

Lane, R. A., J. E. Freer, G. Coxon, and T. Wagener (2021). "Incorporating uncertainty into multi-scale parameter regionalization to evaluate the performance of nationally consistent parameter fields for a hydrological model". In: *Water Resources Research* 57.10, e2020WR028393. DOI: 10.1029/2020WR028393.

Larnier, K., J. Monnier, P.-A. Garambois, and J. V. and (2021). "River discharge and bathymetry estimation from SWOT altimetry measurements". In: *Inverse Problems in Science and Engineering* 29.6, pp. 759–789. DOI: 10.1080/17415977.2020.1803858.

Larnier, K. and J. Monnier (2020). "Hybrid Neural Network–Variational Data Assimilation algorithm to infer river discharges from SWOT-like data". In: *Nonlinear Processes in Geophysics Discussions*, pp. 1–30. DOI: 10.5194/npg-2020-32.

Larnier, K., P.-A. Garambois, C. Emery, L. Pujol, J. Monnier, L. Gal, et al. (2025). "Estimating channel parameters and discharge at river network scale using hydrological-hydraulic models, SWOT and multi-satellite data". In: working paper or preprint. URL: https://hal.inrae.fr/hal-04681079.

Le Coz, J., B. Renard, L. Bonnifait, F. Branger, and R. Le Boursicaud (2014). "Combining hydraulic knowledge and uncertain gaugings in the estimation of hydrometric rating curves: A Bayesian approach". In: *Journal of Hydrology* 509, pp. 573–587. DOI: 10.1016/j.jhydrol.2013.11.016.

Le Mesnil, M. (2021). "Signatures Hydrologiques des Bassins Karstiques". Theses. Montpellier SupAgro. URL: https://tel.archives-ouvertes.fr/tel-03578569.

Leblois, E. and E. Sauquet (1999). *Grid elevation models in hydrology - Part 1: Principales and literrature review; Part 2: HydroDEM, User's manual*. Tech. rep. Cemagref.

LeCun, Y., Y. Bengio, and G. Hinton (2015). "Deep learning". In: *Nature* 521, pp. 436–444. DOI: 10.1038/nature14539.

LeCun, Y., B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, et al. (1989). "Backpropagation Applied to Handwritten Zip Code Recognition". In: *Neural Computation* 1.4, pp. 541–551. DOI: 10.1162/neco.1989.1.4.541.

LeCun, Y., L. Bottou, Y. Bengio, and P. Haffner (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324. DOI: 10.1109/5.726791.

LeCun, Y. (2018). "The Power and Limits of Deep Learning". In: *Research-Technology Management* 61.6, pp. 22–27. DOI: 10.1080/08956308.2018.1516928.

— (2022). "A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27". In: *Open Review* 62.1, pp. 1–62.

LeCun, Y., K. Kavukcuoglu, and C. Farabet (2010). "Convolutional networks and applications in vision". In: *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, pp. 253–256. DOI: 10.1109/ISCAS.2010.5537907.

Lee, H., D.-J. Seo, Y. Liu, V. Koren, P. McKee, and R. Corby (2012). "Variational assimilation of streamflow into operational distributed hydrologic models: effect of spatiotemporal scale of adjustment". In: *Hydrology and Earth System Sciences* 16.7, pp. 2233–2251. DOI: 10.5194/hess-16-2233-2012.

Li, K., G. Huang, S. Wang, S. Razavi, and X. Zhang (2022). "Development of a Joint Probabilistic Rainfall-Runoff Model for High-to-Extreme Flow Projections Under Changing Climatic Conditions". In: *Water Resources Research* 58.6, e2021WR031557. DOI: 10.1029/2021wr031557.

Li, Z., M. Chen, S. Gao, X. Luo, J. J. Gourley, P. Kirstetter, et al. (2021). "CREST-iMAP v1. 0: A fully coupled hydrologic-hydraulic modeling framework dedicated to flood inundation mapping and prediction". In: *Environmental Modelling & Software* 141, p. 105051. DOI: 10.1016/j.envsoft.2021.105051.

Liang, X., D. P. Lettenmaier, E. F. Wood, and S. J. Burges (1994). "A simple hydrologically based model of land surface water and energy fluxes for general circulation models". In: *Journal of Geophysical Research: Atmospheres* 99.D7, pp. 14415–14428. DOI: 10.1029/94JD00483.

Lions, J. L. (1971). *Optimal Control of Systems Governed by Partial Differential Equations*. 1st ed. Vol. 170. Grundlehren der mathematischen Wissenschaften. Springer Berlin, Heidelberg, pp. XI, 400.

Liu, Y. and H. Gupta (2007). "Uncertainty in hydrologic modeling: Toward an integrated data assimilation framework". In: *Water resources research* 43.7. DOI: 10.1029/2006WR005756.

Lyne, V. and M. Hollick (1979). "Stochastic time-variable rainfall-runoff modelling". In: *Institute of Engineers Australia National Conference*. Vol. 79. 10. Institute of Engineers Australia Barton, Australia, pp. 89–93.

MacQueen, J. (1967). "Some methods for classification and analysis of multivariate observations". In: URL: https://api.semanticscholar.org/CorpusID:6278891.

Malou, T., P.-A. Garambois, A. Paris, J. Monnier, and K. Larnier (2021). "Generation and analysis of stage-fall-discharge laws from coupled hydrological-hydraulic river network model integrating sparse multi-satellite data". In: *Journal of Hydrology* 603, p. 126993. DOI: 10.1016/j.jhydrol.2021.126993.

McCabe, M. F., M. Rodell, D. E. Alsdorf, D. G. Miralles, R. Uijlenhoet, W. Wagner, et al. (2017). "The future of Earth observation in hydrology". In: *Hydrology and Earth System Sciences* 21.7, pp. 3879–3914. DOI: 10.5194/hess-21-3879-2017.

McCulloch, W. S. and W. Pitts (1943). "A logical calculus of the ideas immanent in nervous activity". In: *The bulletin of mathematical biophysics* 5.4, pp. 115–133. DOI: 10.1007/BF02478259.

McMillan, H. K. (2021). "A review of hydrologic signatures and their applications". In: *Wiley Interdisciplinary Reviews: Water* 8.1, e1499. DOI: 10.1002/wat2.1499.

Meyer Oliveira, A., A. S. Fleischmann, and R. C. D. Paiva (2021). "On the contribution of remote sensing-based calibration to model hydrological and hydraulic processes in tropical regions". In: *Journal of hydrology (Amsterdam)* 597, p. 126184. DOI: 10.1016/j.jhydrol.2021.126184.

Michel, C. (1989). "Hydrologie appliquée aux petits bassins ruraux". In: *Hydrology handbook (in French), Cemagref, Antony, France.*

Mizukami, N., M. P. Clark, A. J. Newman, A. W. Wood, E. D. Gutmann, B. Nijssen, et al. (2017). "Towards seamless large-domain parameter estimation for hydrologic models". In: *Water Resources Research* 53.9, pp. 8020–8040. DOI: 10.1002/2017WR020401.

Mizukami, N., O. Rakovec, A. J. Newman, M. P. Clark, A. W. Wood, H. V. Gupta, et al. (2019). "On the choice of calibration metrics for "high-flow" estimation using hydrologic models". In: *Hydrology and Earth System Sciences* 23.6, pp. 2601–2614. DOI: 10.5194/hess-23-2601-2019.

Mnih, V., K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, et al. (2015). "Human-level control through deep reinforcement learning". In: *Nature* 518.7540, pp. 529–533. DOI: 10.1038/nature14236.

Molnar, C., G. Casalicchio, and B. Bischl (2020). "Interpretable Machine Learning – A Brief History, State-of-the-Art and Challenges". In: *ECML PKDD 2020 Workshops*. Ed. by I. Koprinska, M. Kamp, A. Appice, C. Loglisci, L. Antonie, A. Zimmermann, et al. Cham: Springer International Publishing, pp. 417–431. DOI: 10.1007/978-3-030-65965-3_28.

Monnier, J. (2021). "Variational Data Assimilation and Model Learning". Master. Lecture. France. URL: https://hal.science/hal-03040047.

Monnier, J., F. Couderc, D. Dartus, K. Larnier, R. Madec, and J.-P. Vila (2016). "Inverse algorithms for 2D shallow water equations in presence of wet dry fronts: Application to flood plain dynamics". In: *Advances in Water Resources* 97, pp. 11–24. DOI: 10.1016/j.advwatres.2016.07.005.

Mulvaney, T. (1850). "On the use of self-registering rain and flood gauges". In: *Proceedings-Institution of Civil Engineers* 4, pp. 1–8.

Mustafa, S. M. T., J. Nossent, G. Ghysels, and M. Huysmans (2018). "Estimation and Impact Assessment of Input and Parameter Uncertainty in Predicting Groundwater Flow With a Fully Distributed Model". en. In: *Water Resources Research* 54.9, pp. 6585–6608. DOI: 10.1029/2017WR021857.

Nash, J. (1957). "The form of the instantaneous unit hydrograph". In: *Comptes Rendus et Rapports Assemblee Generale de Toronto* 3, pp. 114–121.

Nathan, R. J. and T. A. McMahon (1990). "Evaluation of automated techniques for base flow and recession analyses". In: *Water resources research* 26.7, pp. 1465–1473. DOI: 10.1029/WR026i007p01465.

Nearing, G., F. Kratzert, A. K. Sampson, C. S. Pelissier, D. Klotz, J. M. Frame, et al. (2021). "What Role Does Hydrological Science Play in the Age of Machine Learning?" In: *Water Resources Research* 57.3. DOI: 10.1029/2020WR028091.

Ochoa-Rodriguez, S., L.-P. Wang, P. Willems, and C. Onof (2019). "A Review of Radar-Rain Gauge Data Merging Methods and Their Potential for Urban Hydrological Applications". In: *Water Resources Research* 55.8, pp. 6356–6391. DOI: 10.1029/2018WR023332.

Odry, J. (2017). "Prédétermination des débits de crues extrêmes en sites non jaugés : régionalisation de la méthode par simulation SHYREG". Thèse de doctorat dirigée par Arnaud, Patrick

Géosciences de l'environnement. Hydrologie Aix-Marseille 2017. PhD thesis. URL: http://www.theses.fr/2017AIXM0424.

Oki, T. and S. Kanae (2006). "Global Hydrological Cycles and World Water Resources". In: *Science* 313.5790, pp. 1068–1072. DOI: 10.1126/science.1128845.

Organde, D., P. Arnaud, J.-A. Fine, C. Fouchier, N. Folton, and J. Lavabre (2013). "Régionalisation d'une méthode de prédétermination de crue sur l'ensemble du territoire français: la méthode SHYREG". In: *Revue des Sciences de l'Eau* 26.1, pp. 65–78. DOI: 10.7202/1014920ar.

Oudin, L., V. Andréassian, C. Perrin, C. Michel, and N. Le Moine (2008). "Spatial proximity, physical similarity, regression and ungaged catchments: A comparison of regionalization approaches based on 913 French catchments". In: *Water Resources Research* 44.3. DOI: 10.1029/2007WR006240.

Oudin, L., F. Hervieu, C. Michel, C. Perrin, V. Andréassian, F. Anctil, et al. (2005). "Which potential evapotranspiration input for a lumped rainfall–runoff model?: Part 2 Towards a simple and efficient potential evapotranspiration model for rainfall–runoff modelling". In: *Journal of hydrology* 303.1-4, pp. 290–306. DOI: 10.1016/j.jhydrol.2004.08.026.

Oudin, L., A. Kay, V. Andréassian, and C. Perrin (2010). "Are seemingly physically similar catchments truly hydrologically similar?" In: *Water Resources Research* 46.11. DOI: 10.1029/2009WR008887.

Paiva, R. C., W. Collischonn, and C. E. Tucci (2011). "Large scale hydrologic and hydrodynamic modeling using limited data and a GIS based approach". In: *Journal of Hydrology* 406.3-4, pp. 170–181. DOI: 10.1016/j.jhydrol.2011.06.007.

Pan, L. and L. Wu (1998). "A hybrid global optimization method for inverse estimation of hydraulic parameters: Annealing-Simplex Method". In: *Water Resources Research* 34.9, pp. 2261–2269. DOI: 10.1029/98WR01672.

Papamarkou, T., J. Hinkle, M. T. Young, and D. Womble (2022). "Challenges in Markov Chain Monte Carlo for Bayesian Neural Networks". In: *Statistical Science* 37.3. DOI: 10.1214/21-STS840.

Parajka, J., A. Viglione, M. Rogger, J. Salinas, M. Sivapalan, and G. Blöschl (2013). "Comparative assessment of predictions in ungauged basins–Part 1: Runoff-hydrograph studies". In: *Hydrology and Earth System Sciences* 17.5, pp. 1783–1795. DOI: 10.5194/hess-17-1783-2013.

Parajka, J., R. Merz, and G. Blöschl (2005). "A comparison of regionalisation methods for catchment model parameters". In: *Hydrology and Earth System Sciences* 9.3, pp. 157–171. DOI: 10.5194/hess-9-157-2005.

Paszke, A., S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, et al. (2019). "PyTorch: an imperative style, high-performance deep learning library". In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc.

Pelletier, A. and V. Andréassian (2020). "Hydrograph separation: an impartial parametrisation for an imperfect method". In: *Hydrology and earth system sciences* 24.3, pp. 1171–1187. DOI: 10.5194/hess-24-1171-2020.

Perrin, C., C. Michel, and V. Andrèassian (2003). "Improvement of a parsimonious model for streamflow simulation". In: *Journal of hydrology* 279.1-4, pp. 275–289. DOI: 10.1016/S0022-1694(03)00225-7.

Peterson, P. (2009). "F2PY: a tool for connecting Fortran and Python programs". In: *International Journal of Computational Science and Engineering* 4.4, pp. 296–305. DOI: 10.1504/IJCSE.2009.029165.

Piotte, O., T. Montmerle, C. Fouchier, A. Belleudy, L. Garandeau, B. Janet, et al. (2020). "Les évolutions du service d'avertissement sur les pluies intenses et les crues soudaines en France". In: *La Houille Blanche* 106.6, pp. 75–84. DOI: 10.1051/lhb/2020055.

Poncelet, C. (2016). "Du bassin au paramètre : jusqu'où peut-on régionaliser un modèle hydrologique conceptuel ?" Thèse de doctorat dirigée par Andréassian, Vazken et Oudin, Ludovic Hydrologie Paris 6 2016. PhD thesis. URL: http://www.theses.fr/2016PA066550.

Pujol, L., P.-A. Garambois, and J. Monnier (2022). "Multi-dimensional hydrological–hydraulic model with variational data assimilation for river networks and floodplains". In: *Geoscientific Model Development* 15.15, pp. 6085–6113. DOI: 10.5194/gmd-15-6085-2022.

Pujol, L., P.-A. Garambois, P. Finaud-Guyot, J. Monnier, K. Larnier, R. Mose, et al. (2020). "Estimation of multiple inflows and effective channel by assimilation of multi-satellite hydraulic signatures: The ungauged anabranching Negro river". In: *Journal of Hydrology* 591, p. 125331. DOI: 10.1016/j.jhydrol.2020.125331.

Puy, A., W. Becker, S. L. Piano, and A. Saltelli (2022). "A comprehensive comparison of total-order estimators for global sensitivity analysis". In: *International Journal for Uncertainty Quantification* 12.2. DOI: 10.1615/Int.J.UncertaintyQuantification.2021038133.

Quintana-Seguí, P., P. Le Moigne, Y. Durand, E. Martin, F. Habets, M. Baillon, et al. (2008). "Analysis of Near-Surface Atmospheric Variables: Validation of the SAFRAN Analysis over France". In: *Journal of Applied Meteorology and Climatology* 47.1, p. 92. DOI: 10.1175/2007JAMC1636.1.

Rackauckas, C., Y. Ma, J. Martensen, C. Warner, K. Zubov, R. Supekar, et al. (2021). *Universal Differential Equations for Scientific Machine Learning*. DOI: 10.48550/arXiv.2001.04385.

Raissi, M., P. Perdikaris, and G. Karniadakis (2019). "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *Journal of Computational Physics* 378, pp. 686–707. DOI: 10.1016/j.jcp.2018.10.045.

Ramesh, A., M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, et al. (2021). "Zero-Shot Text-to-Image Generation". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by M. Meila and T. Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, pp. 8821–8831. URL: https://proceedings.mlr.press/v139/ramesh21a.html.

Rantz, S. E. (1982). *Measurement and computation of streamflow*. DOI: 10.3133/wsp2175.

Razavi, S. and H. Gupta (2019). "A multi-method Generalized Global Sensitivity Matrix approach to accounting for the dynamical nature of earth and environmental systems models". In: *Environmental modelling & software* 114, pp. 1–11. DOI: 10.1016/j.envsoft.2018.12.002.

Razavi, T. and P. Coulibaly (2013). "Streamflow prediction in ungauged basins: review of regionalization methods". In: *Journal of hydrologic engineering* 18.8, pp. 958–975. DOI: 10.1061/(ASCE)HE.1943-5584.0000690.

Reichl, J. P. C., A. W. Western, N. R. McIntyre, and F. H. S. Chiew (2009). "Optimization of a similarity measure for estimating ungauged streamflow". In: *Water Resources Research* 45.10. DOI: 10.1029/2008WR007248.

Reichstein, M., G. Camps-Valls, B. Stevens, M. Jung, J. Denzler, N. Carvalhais, 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.

Renard, B., V. Garreta, and M. Lang (2006). "An application of Bayesian analysis and Markov chain Monte Carlo methods to the estimation of a regional trend in annual maxima". In: *Water Resources Research* 42.12. DOI: 10.1029/2005WR004591.

Renard, B., D. Kavetski, G. Kuczera, M. Thyer, and S. W. Franks (2010). "Understanding predictive uncertainty in hydrologic modeling: The challenge of identifying input and structural errors". In: *Water Resources Research* 46.5. DOI: 10.1029/2009WR008328.

Renard, B., D. Kavetski, E. Leblois, M. Thyer, G. Kuczera, and S. W. Franks (2011). "Toward a reliable decomposition of predictive uncertainty in hydrological modeling: Characterizing rainfall errors using conditional simulation". In: *Water Resources Research* 47.11. DOI: 10.1029/2011WR010643.

Ronneberger, O., P. Fischer, and T. Brox (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi. Springer International Publishing, pp. 234–241.

Rosenblatt, F. (1958). "The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6, p. 386.

Roux, H., D. Labat, P.-A. Garambois, M.-M. Maubourguet, J. Chorda, and D. Dartus (2011). "A physically-based parsimonious hydrological model for flash floods in Mediterranean catchments". In: *Natural Hazards & Earth System Sciences* 11.9, pp. 2567–2582. DOI: 10.5194/nhess-11-2567-2011.

Rumelhart, D. E., G. E. Hinton, and R. J. Williams (1986). "Learning representations by back-propagating errors". In: *Nature* 323.6088, pp. 533–536. DOI: 10.1038/323533a0.

Saadi, M., L. Oudin, and P. Ribstein (2019). "Random forest ability in regionalizing hourly hydrological model parameters". In: *Water* 11.8, p. 1540. DOI: 10.3390/w11081540.

Saharia, M., P.-E. Kirstetter, H. Vergara, J. J. Gourley, I. Emmanuel, and H. Andrieu (2021). "On the impact of rainfall spatial variability, geomorphology, and climatology on flash floods". In: *Water Resources Research* 57.9, e2020WR029124. DOI: 10.1029/2020WR029124.

Saltelli, A. (2002). "Making best use of model evaluations to compute sensitivity indices". In: *Computer physics communications* 145.2, pp. 280–297. DOI: 10.1016/S0010-4655(02)00280-1.

Samaniego, L., R. Kumar, S. Thober, O. Rakovec, M. Zink, N. Wanders, et al. (2017). "Toward seamless hydrologic predictions across spatial scales". In: *Hydrology and Earth System Sciences* 21.9, pp. 4323–4346. DOI: 10.5194/hess-21-4323-2017.

Samaniego, L., R. Kumar, and S. Attinger (2010). "Multiscale parameter regionalization of a grid-based hydrologic model at the mesoscale". In: *Water Resources Research* 46.5. DOI: 10.1029/2008WR007327.

Santos, L., G. Thirel, and C. Perrin (2018). "Continuous state-space representation of a bucket-type rainfall-runoff model: a case study with the GR4 model using state-space GR4 (version 1.0)". In: *Geoscientific Model Development* 11.4, pp. 1591–1605. DOI: 10.5194/gmd-11-1591-2018.

Seibert, J. (1999). "Regionalisation of parameters for a conceptual rainfall-runoff model". In: *Agricultural and forest meteorology* 98, pp. 279–293. DOI: 10.1016/S0168-1923(99)00105-7.

Shen, C. (2018). "A Transdisciplinary Review of Deep Learning Research and Its Relevance for Water Resources Scientists". In: *Water Resources Research* 54.11, pp. 8558–8593. DOI: 10.1029/2018WR022643.

Shen, C., A. P. Appling, P. Gentine, T. Bandai, H. Gupta, A. Tartakovsky, et al. (2023). "Differentiable modelling to unify machine learning and physical models for geosciences". In: *Nature Reviews Earth & Environment* 4.8, pp. 552–567. DOI: 10.1038/s43017-023-00450-9.

Shen, C. and K. Lawson (2021). "Applications of Deep Learning in Hydrology". In: *Deep Learning for the Earth Sciences*. John Wiley & Sons, Ltd. Chap. 19, pp. 283–297. DOI: 10.1002/9781119646181.ch19.

Sherman, L. K. (1932). "Streamflow from rainfall by the unit-graph method". In: *Eng. News Record* 108, pp. 501–505.

Singh, V. P. (2018). "Hydrologic modeling: progress and future directions". In: *Geoscience letters* 5.1, pp. 1–18. DOI: 10.1186/s40562-018-0113-z.

Sit, M., B. Z. Demiray, Z. Xiang, G. J. Ewing, Y. Sermet, and I. Demir (2020). "A comprehensive review of deep learning applications in hydrology and water resources". In: *Water Science and Technology* 82.12, pp. 2635–2670. DOI: 10.2166/wst.2020.369.

Sivapalan, M. (2003). "Prediction in ungauged basins: a grand challenge for theoretical hydrology". In: *Hydrological Processes* 17.15, pp. 3163–3170. DOI: 10.1002/hyp.5155.

Skahill, B. E. and J. Doherty (2006). "Efficient accommodation of local minima in watershed model calibration". In: *Journal of Hydrology* 329.1, pp. 122–139. DOI: 10.1016/j.jhydrol.2006.02.005.

Sobol, I. and S. Kucherenko (2010). "Derivative based global sensitivity measures". In: *Procedia-Social and Behavioral Sciences* 2.6, pp. 7745–7746.

Solomatine, D. and A. Ostfeld (2008). "Data-driven modelling: some past experiences and new approaches". In: *Journal of Hydroinformatics* 10.1, pp. 3–22. DOI: 10.2166/hydro.2008.015.

Song, Y., W. J. M. Knoben, M. P. Clark, D. Feng, K. Lawson, K. Sawadekar, et al. (2024). "When ancient numerical demons meet physics-informed machine learning: adjoint-based gradients for implicit differentiable modeling". In: *Hydrology and Earth System Sciences* 28.13, pp. 3051–3077. DOI: 10.5194/hess-28-3051-2024.

Song, Z., J. Xia, G. Wang, D. She, C. Hu, and S. Hong (2022). "Regionalization of hydrological model parameters using gradient boosting machine". In: *Hydrology and Earth System Sciences* 26.2, pp. 505–524. DOI: 10.5194/hess-26-505-2022.

Špitalar, M., J. J. Gourley, C. Lutoff, P.-E. Kirstetter, M. Brilly, and N. Carr (2014). "Analysis of flash flood parameters and human impacts in the US from 2006 to 2012". In: *Journal of Hydrology* 519, pp. 863–870. DOI: 10.1016/j.jhydrol.2014.07.004.

Syed, S., A. Bouchard-Côté, G. Deligiannidis, and A. Doucet (2022). "Non-Reversible Parallel Tempering: A Scalable Highly Parallel MCMC Scheme". en. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 84.2, pp. 321–350. DOI: 10.1111/rssb.12464.

Tarasova, L., S. Basso, M. Zink, and R. Merz (2018). "Exploring controls on rainfall-runoff events: 1. Time series-based event separation and temporal dynamics of event runoff response in Germany". In: *Water Resources Research* 54.10, pp. 7711–7732. DOI: 10.1029/2018WR022587.

Te Chow, V., D. R. Maidment, and L. W. Mays (1988). *Applied Hydrology*. McGraw-Hill Series in Water Resources and Environmental Engineering.

Tikhonov, A. N. et al. (1943). "On the stability of inverse problems". In: *Dokl. akad. nauk sssr*. Vol. 39. 5, pp. 195–198.

Toosi, A., A. G. Bottino, B. Saboury, E. Siegel, and A. Rahmim (2021). "A Brief History of AI: How to Prevent Another Winter (A Critical Review)". In: *PET Clinics* 16.4, pp. 449–469. DOI: 10.1016/j.cpet.2021.07.001.

Tripathy, K. P. and A. K. Mishra (2024). "Deep learning in hydrology and water resources disciplines: concepts, methods, applications, and research directions". In: *Journal of Hydrology* 628, p. 130458. DOI: 10.1016/j.jhydrol.2023.130458.

Turing, A. M. (1950). "I.—COMPUTING MACHINERY AND INTELLIGENCE". In: *Mind* LIX.236, pp. 433–460. DOI: 10.1093/mind/LIX.236.433.

Turnipseed, D. P. and V. B. Sauer (2010). *Discharge measurements at gaging stations*. Reston, VA. DOI: 10.3133/tm3A8.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al. (2023). *Attention Is All You Need*. DOI: 10.48550/arXiv.1706.03762.

Vergara, H., P.-E. Kirstetter, J. J. Gourley, Z. L. Flamig, Y. Hong, A. Arthur, et al. (2016). "Estimating a-priori kinematic wave model parameters based on regionalization for flash flood forecasting in the Conterminous United States". In: *Journal of Hydrology* 541, pp. 421–433. DOI: 10.1016/j.jhydrol.2016.06.011.

Vidal, J.-P., E. Martin, L. Franchistéguy, M. Baillon, and J.-M. Soubeyroux (2010). "A 50-year high-resolution atmospheric reanalysis over France with the Safran system". In: *International Journal of Climatology* 30.11, pp. 1627–1644. DOI: 10.1002/joc.2003.

Vinet, F. (2008). "Geographical analysis of damage due to flash floods in southern France: The cases of 12–13 November 1999 and 8–9 September 2002". In: *Applied Geography* 28.4, pp. 323–336. DOI: 10.1016/j.apgeog.2008.02.007.

Vrugt, J. A., C. J. Ter Braak, M. P. Clark, J. M. Hyman, and B. A. Robinson (2008). "Treatment of input uncertainty in hydrologic modeling: Doing hydrology backward with Markov chain Monte Carlo simulation". In: *Water Resources Research* 44.12. DOI: 10.1029/2007WR006720.

Vrugt, J. A. and B. A. Robinson (2007). "Improved evolutionary optimization from genetically adaptive multimethod search". In: *Proceedings of the National Academy of Sciences* 104.3, pp. 708–711. DOI: 10.1073/pnas.0610471104.

Wagener, T., M. Sivapalan, P. Troch, and R. Woods (2007). "Catchment Classification and Hydrologic Similarity". In: *Geography Compass* 1.4, pp. 901–931. DOI: 10.1111/j.1749-8198.2007.00039.x.

Wang, C., S. Jiang, Y. Zheng, F. Han, R. Kumar, O. Rakovec, et al. (2024). "Distributed Hydrological Modeling With Physics-Encoded Deep Learning: A General Framework and Its Application in the Amazon". In: *Water Resources Research* 60.4, e2023WR036170. DOI: 10.1029/2023WR036170.

Wang, N., W.-j. Zhao, N. Wu, and D. Wu (2017). "Multi-objective optimization: a method for selecting the optimal solution from Pareto non-inferior solutions". In: *Expert Systems with Applications* 74, pp. 96–104. DOI: 10.1016/j.eswa.2017.01.004.

Wang, Z. and G. P. Rangaiah (2017). "Application and analysis of methods for selecting an optimal solution from the Pareto-optimal front obtained by multiobjective optimization". In: *Industrial & Engineering Chemistry Research* 56.2, pp. 560–574. DOI: 10.1021/acs.iecr.6b03453.

Widén-Nilsson, E., S. Halldin, and C.-y. Xu (2007). "Global water-balance modelling with WASMOD-M: Parameter estimation and regionalisation". In: *Journal of Hydrology* 340.1-2, pp. 105–118. DOI: 10.1016/j.jhydrol.2007.04.002.

Wilde, S. A., J. W. Valley, W. H. Peck, and C. M. Graham (2001). "Evidence from detrital zircons for the existence of continental crust and oceans on the Earth 4.4 Gyr ago". In: *Nature* 409.6817, pp. 175–178. DOI: 10.1038/35051550.

Winsemius, H. C., J. C. Aerts, L. P. Van Beek, M. F. Bierkens, A. Bouwman, B. Jongman, et al. (2016). "Global drivers of future river flood risk". In: *Nature Climate Change* 6.4, pp. 381–385. DOI: 10.1038/nclimate2893.

Xiang, Z., J. Yan, and I. Demir (2020). "A Rainfall-Runoff Model With LSTM-Based Sequence-to-Sequence Learning". In: *Water Resources Research* 56.1, e2019WR025326. DOI: 10.1029/2019WR025326.

Yeh, W. W.-G. (1986). "Review of parameter identification procedures in groundwater hydrology: The inverse problem". In: *Water resources research* 22.2, pp. 95–108. DOI: 10.1029/WR022i002p00095.

Yin, Y., V. L. Guen, J. Dona, E. de Bézenac, I. Ayed, N. Thome, et al. (2021). "Augmenting physical models with deep networks for complex dynamics forecasting". In: *Journal of Statistical Mechanics: Theory and Experiment* 2021.12, p. 124012. DOI: 10.1088/1742-5468/ac3ae5.

Zhao, R.-J., Y.-L. Zuang, L.-R. Fang, X.-R. Liu, and Q. Zhang (1980). "The Xinanjiang Model". In: IAHS-AISH Publ. 129.

Zhu, C., R. H. Byrd, P. Lu, and J. Nocedal (1997). "Algorithm 778: L-BFGS-B: Fortran Subroutines for Large-Scale Bound-Constrained Optimization." In: *ACM Trans. Math. Softw.* 23.4, pp. 550–560. DOI: 10.1145/279232.279236.

Zoccatelli, D., M. Borga, A. Viglione, G. Chirico, and G. Blöschl (2011). "Spatial moments of catchment rainfall: rainfall spatial organisation, basin morphology, and flood response". In: *Hydrology and Earth System Sciences* 15.12, pp. 3767–3783. DOI: 10.5194/hess-15-3767-2011.

Appendices

Appendix A

Hydrological Evaluation Metrics and Properties

Let $\boldsymbol{Q}(t)$ and $\boldsymbol{Q}^*(t)$ denote the simulated and observed discharge time series, respectively, over a time period $t \in [t_0, T]$. The hydrological performance metrics analyzed in this study are:

A.1 Nash-Sutcliffe Efficiency

The Nash-Sutcliffe Efficiency (NSE) is defined as:

$$NSE = 1 - \frac{\sum_{t=t_0}^{T} \left(\boldsymbol{Q}^*(t) - \boldsymbol{Q}(t)\right)^2}{\sum_{t=t_0}^{T} \left(\boldsymbol{Q}^*(t) - \bar{\boldsymbol{Q}}^*\right)^2} \tag{A.1}$$

where:

— $\bar{\boldsymbol{Q}}^*$ is the mean of the observed discharge time series:

$$\bar{\boldsymbol{Q}}^* = \frac{1}{T - t_0 + 1} \sum_{t=t_0}^{T} \boldsymbol{Q}^*(t) \tag{A.2}$$

— The numerator represents the sum of squared errors between observed and simulated discharges.

— The denominator represents the variance of the observed discharge.

NSE values range from $-\infty$ to 1, where values close to 1 indicate a better fit. This function is fully differentiable.

A.2 Normalized Nash-Sutcliffe Efficiency

The Normalized Nash-Sutcliffe Efficiency (NNSE) is defined as:

$$NNSE = \frac{1}{1 - NSE} \tag{A.3}$$

where NSE is given by Equation A.1.

NNSE is always positive, with a value of 1 indicating perfect agreement. This function is not differentiable at $NSE = 1$.

A.3 Kling-Gupta Efficiency

The Kling-Gupta Efficiency (KGE) is given by:

$$KGE = 1 - \sqrt{a_1 (r-1)^2 + a_2 (\beta - 1)^2 + a_3 (\alpha - 1)^2} \tag{A.4}$$

where:

— r is the Pearson correlation coefficient between the observed and simulated discharge time series:

$$r = \frac{\sum_{t=t_0}^{T} \left(\boldsymbol{Q}^*(t) - \bar{\boldsymbol{Q}}^*\right)\left(\boldsymbol{Q}(t) - \bar{\boldsymbol{Q}}\right)}{\sqrt{\sum_{t=t_0}^{T} \left(\boldsymbol{Q}^*(t) - \bar{\boldsymbol{Q}}^*\right)^2} \sqrt{\sum_{t=t_0}^{T} \left(\boldsymbol{Q}(t) - \bar{\boldsymbol{Q}}\right)^2}} \tag{A.5}$$

— β is the bias ratio:

$$\beta = \frac{\bar{\boldsymbol{Q}}}{\bar{\boldsymbol{Q}}^*} \tag{A.6}$$

where $\bar{\boldsymbol{Q}}$ is the mean of the simulated discharge time series.

— α is the variability ratio:

$$\alpha = \frac{\sigma_{\boldsymbol{Q}}}{\sigma^*_{\boldsymbol{Q}}} \tag{A.7}$$

where $\sigma_{\boldsymbol{Q}}$ and $\sigma^*_{\boldsymbol{Q}}$ are the standard deviations of the simulated and observed discharge time series, respectively.

— a_1, a_2, a_3 are non-negative weighting parameters such that:

$$\sum_{i=1}^{3} a_i = 1 \tag{A.8}$$

KGE values range from $-\infty$ to 1, where values closer to 1 indicate better model performance. This function is not differentiable at points where the square root argument is zero.

A.4 Modified Kling-Gupta Efficiency

The Modified Kling-Gupta Efficiency (KGE_2) is a differentiable version of KGE, making it more suitable for gradient-based optimization methods. It is defined as:

$$KGE_2 = 1 - \left[a_1 (r-1)^2 + a_2 (\beta - 1)^2 + a_3 (\alpha - 1)^2\right] \tag{A.9}$$

where $r, \beta, \alpha, (a_i)_{i=1..3}$ are defined in Equations A.5, A.6, A.7, A.8.

KGE_2 values range from $-\infty$ to 1, where values closer to 1 indicate better model performance. This function is fully differentiable.

A.5 Mean Square Error

The Mean Square Error (MSE) is given by:

$$MSE = \frac{1}{T - t_0 + 1} \sum_{t=t_0}^{T} \left(\boldsymbol{Q}^*(t) - \boldsymbol{Q}(t)\right)^2 \tag{A.10}$$

where:

— The squared differences between observed and simulated discharges penalize large errors more heavily than small errors.
— The division by the number of time steps normalizes the metric.

MSE values are always non-negative, with lower values indicating better model performance. This function is fully differentiable.

A.6 Root Mean Square Error

The Root Mean Square Error (RMSE) is defined as:

$$RMSE = \sqrt{\frac{1}{T - t_0 + 1} \sum_{t=t_0}^{T} \left(\boldsymbol{Q}^*(t) - \boldsymbol{Q}(t)\right)^2} \tag{A.11}$$

RMSE measures the average magnitude of the error in the same units as the discharge values. Lower RMSE values indicate better performance. This function is not differentiable at zero.

Appendix B

Multi-Objective Optimization with Spatially Uniform Model Parameters

We look into multi-objective optimization for a global calibration of spatially uniform parameters, i.e. a low dimensional control $\overline{\boldsymbol{\theta}}$. The multi-objective calibration is simply defined as the optimization problem:

$$\min_{\boldsymbol{\theta}\in\mathcal{O}\subset\mathbb{R}^n}(j_1(\boldsymbol{\theta}),...,j_m(\boldsymbol{\theta})) \tag{B.1}$$

where $\boldsymbol{\theta}$ is the n-dimensional vector of model parameters in the feasible space $\mathcal{O}\subset\mathbb{R}^n$ and $j_1,...,j_m$ are the m single-objective functions to be simultaneously minimized.

B.1 Pareto front

In single-objective optimization, the Pareto optimal solution is unique (in terms of objective space) but in multi-objective problem, it common to have several solutions that cannot be defined which one is the best. If the optimization problem is non-dominated, or non-inferior (each objective function is its own entity, so no individual can be better off without making at least one individual worse off), then we call that Pareto optimality, or Pareto efficiency. A Pareto front (in terms of parameter space) is a set of all Pareto efficient solutions that need to be estimated. Let us consider two feasible solutions: $\boldsymbol{\theta}_1,\boldsymbol{\theta}_2\in\mathcal{O}$. Then, $\boldsymbol{\theta}_1$ is said to Pareto dominate $\boldsymbol{\theta}_2$ if the following properties hold:

1. $\forall i\in\{1,...,m\}, j_i(\boldsymbol{\theta}_1)\leq j_i(\boldsymbol{\theta}_2)$;
2. $\exists i\in\{1,...,m\}, j_i(\boldsymbol{\theta}_1)< j_i(\boldsymbol{\theta}_2)$.

We call $\mathcal{P}$ the Pareto set representing all of Pareto solutions. By definition, a Pareto solution $\boldsymbol{\theta}^*\in\mathcal{P}$ of problem B.1 must fill the two following conditions:

1. $\nexists\boldsymbol{\theta}'\in\mathcal{O}\setminus\mathcal{P}, \exists i\in\{1,...,m\}, j_i(\boldsymbol{\theta}')< j_i(\boldsymbol{\theta}^*)$;
2. $\nexists\boldsymbol{\theta}''\in\mathcal{P}, \boldsymbol{\theta}''$ dominates $\boldsymbol{\theta}^*$.

The first statement indicates that there does not exist other point in the feasible space that reduces at least one objective function while keeping others unchanged, so the Pareto set is the optimal set. The second says that, no other point exists in the Pareto set that decreases one objective function without increasing another one, so it is impossible to distinguish any solution as being better than the other in the Pareto set. Figure B.1 illustrates this for a simple problem where we have 2-objective functions j_1, j_2. The Pareto front (in terms of objective space) represents all of non-dominated optimal solutions. It implies that, it is impossible to move from any point in the feasible space and simultaneously decrease the two objective functions without violating a constraint.

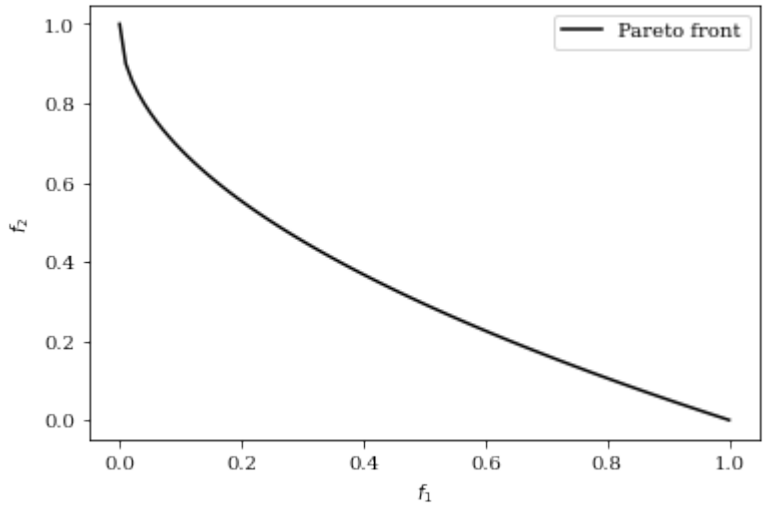


Figure B.1 – Illustration of Pareto front in terms of objective space.

B.2 Overview of GA and NSGA

Genetic algorithm (GA) is a "heuristic algorithm" (or search heuristic) in optimization, inspired by the Theory of Natural Evolution, whose selection operators include "crossover" and "mutation". Basically, the process of a GA consists of the following three phases:

1. *Population initialization.* The population is randomly initialized based on the problem range and constraint. The size of the population determines also the number of solutions, called "pop-size".
2. *Parents selection (sorting).* A fitness function is defined to calculate the fitness score (also called Pareto ranking in multi-objective optimization) that determines how fit an individual is to the problem. Then, the fitness score decides the probability of selecting an individual as a parent to reproduce offspring population.
3. *Mating.* For each pair of parent to be mated, new offspring are created by exchanging the genes of parents among themselves (crossover operator). To maintain the diversity within the population and prevent premature convergence, some of the bits in the gene of certain new offspring can be flipped with a low random probability (mutation operator). Offspring are created until their pop-size is equal to the pop-size of previous generation.

Suggested by Deb et al. (2002), NSGA is a well known multi-objective genetic algorithm for solving multi/many-objective optimization problems, including fast and elitist approach (Deb et al., 2002). Namely, a fast sorting algorithm helps optimizing the computational complexity (even with a large population size) arising from the non-dominated sorting procedure in every generation. Into the bargain, NSGA possesses a diversity preservation property, based on a sharing function method, that prevents the loss of good solutions involved in the mating process. Recently, NSGA has also been implemented in the *pymoo* Python library (Blank et al., 2020), that is used in the present study thanks to the Python interface of the `smash` platform.

B.3 Selection of an optimal solution from Pareto front

We aim to select an optimal solution that is acceptable for every objective within a constraint on principal objective function. Many strategies can be chosen to perform such a selection (e.g. based

on the sensitivity ratio that is the ratio of the average variabilities of a certain non-inferior solution to the corresponding value of the objective function in the Pareto front (N. Wang et al., 2017), or the Euclidean distance from the ideal solution (Z. Wang et al., 2017)). A simple additive weighting (SAW) method in Z. Wang et al. (2017) can be used in our case by adding a normalization operator and assigned weightage for the objective functions.

Considering an objective matrix $(j_{ij})_{1\leq i\leq m, 1\leq j\leq n}$, where m is the number of non-dominated solutions, n is the number of objective functions. Then each row i represents the i^{th} solution set of the Pareto front and each column j represents all non-inferior solutions of the j^{th} objective function. Denote c be the index of the classical objective function (for example $1-NSE$ or $1-KGE$), which is the most constrained function to find a unique optimal solution from Pareto front. This algorithm is detailed in the following three phases:

1. Computing normalized objective matrix $(F_{ij})_{1\leq i\leq m, 1\leq j\leq n}$: $F_{ij} = \frac{f_j^+ - f_{ij}}{f_j^+ - f_j^-}$
   where $f_j^+ = \max_{1\leq i\leq m} f_{ij}$ and $f_j^- = \min_{1\leq i\leq m} f_{ij}$
2. Assigning weightage for normalized objective matrix $(G_{ij})_{1\leq i\leq m, 1\leq j\leq n}$: $G_{ij} = w_j \times F_{ij}$
   where $w_j = \begin{cases} e^d, \text{ if } j = c \\ e - e^d, \text{ otherwise} \end{cases}$ and $d = f_c^+ - f_c^-$
3. Finding optimal solution: $\boldsymbol{\theta} = (f_{k1}, ..., f_{kn})$ where $k = \arg\max_{1\leq i\leq m} \left(\sum_{j=1}^n G_{ij}\right)$.

Appendix C

Incorporating Regionalization Neural Networks into the Differentiable Hydrological Model

This appendix details the neural network design and the derivation of hydrological cost gradients for the ANN-based regionalization algorithm.

A simple ANN denoted $\mathcal{N}n$, consisting of N_L fully connected (dense) layers, intends to learn the descriptors-to-parameters field mapping in the 2D spatial domain, from $\boldsymbol{D}(x) \in \mathbb{R}^{N_D}$ to $\boldsymbol{\theta}(x) \in \mathbb{R}^{N_\theta}, \forall x \in \Omega$ (Figure C.1).

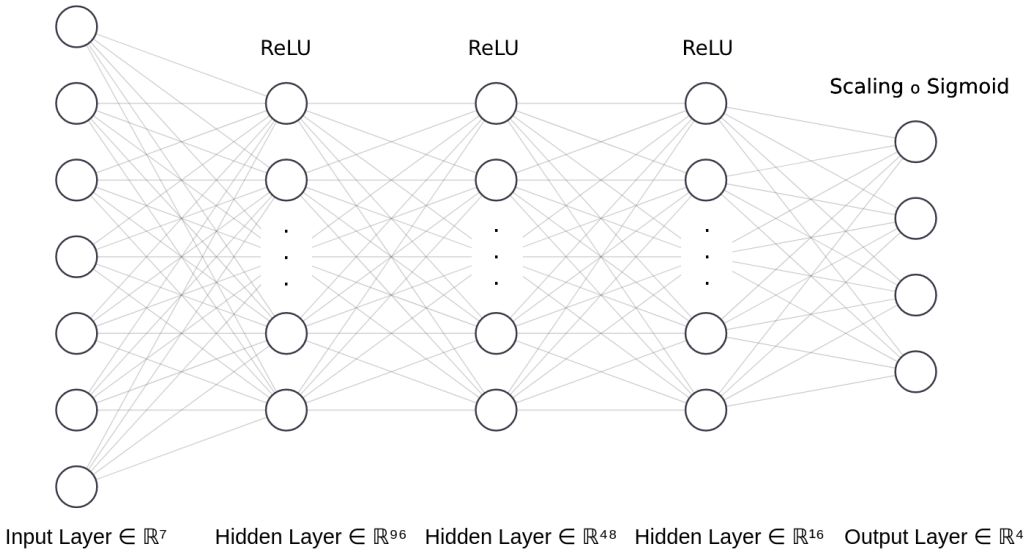


Figure C.1 – The architecture of the ANN consists of three hidden layers followed by the ReLU activation function and an output layer that uses the Sigmoid activation function in combination with a scaling function. In this particular case, we have $N_D = 7$, $N_L = 4$ and $N_\theta = 4$. The total number of trainable parameters is calculated in Table C.1.

Let us consider an ensemble of layers where each layer is associated with its weight $\boldsymbol{W}_j$ and bias $\boldsymbol{b}_j$. Then, an input I of each layer is mapped to the input of the next layer by a linear function $\phi_j(I) = \boldsymbol{W}_j I + \boldsymbol{b}_j$, and followed by the ReLU activation function denoted δ, except for the last layer, which is followed by the Sigmoid activation function denoted σ, ensuring that its outputs are

Table C.1 – Number of parameters of the ANN where $N_D = 7$, $N_L = 4$ and $N_\theta = 4$.

	Hidden layer 1	Hidden layer 2	Hidden layer 3	Output layer
Input shape	$(N_D,\ )$	$(96,\ )$	$(48,\ )$	$(16,\ )$
Number of neurons	96	48	16	N_θ
Number of parameters	$N_D.96 + 96 = 768$	$96.48 + 48 = 4656$	$48.16 + 16 = 784$	$16.N_\theta + N_\theta = 68$

Total parameters: 6276.

between 0 and 1. Now an output $\boldsymbol{O}_x = \sigma \circ \phi_{N_L}(.,x) \in [0,1]^{N_\theta}$ of the last layer is mapped to the range of the hydrological model parameters by a differentiable scaling function s:

$$\boldsymbol{\theta}(x) = s(\boldsymbol{O}_x) = l + (u - l) \odot \boldsymbol{O}_x \tag{C.1}$$

where $l = (l_1, ..., l_{N_\theta})$ and $u = (u_1, ..., u_{N_\theta})$ with the lower and upper bounds $l_k \in \mathbb{R}$ and $u_k \in \mathbb{R}$, assumed spatially uniform, defining the bound constraints of $\theta_k(x), \forall (k,x) \in [1..N_\theta] \times \Omega$, in the direct hydrological model. The notation "$\odot$" denotes the Hadamard product. Noting $\Psi_j \equiv \begin{cases} \delta \circ \phi_j,\ j = 1..N_L - 1 \\ \sigma \circ \phi_j,\ j = N_L \end{cases}$, the forward propagation of the neural network $\mathcal{N}n$ is defined as Equation C.2.

$$\boldsymbol{\theta}(x) = \mathcal{N}n(\boldsymbol{D}(x), .) = s \circ \Psi_{N_L} \circ \Psi_{N_L - 1} \circ ... \circ \Psi_1(\boldsymbol{D}(x)), \forall x \in \Omega. \tag{C.2}$$

Here, the notation "$\circ$" denotes the function composition operator.

Recall that our objective is the calibration problem of Equation 3.9 with respect to the regional control vector $\boldsymbol{\rho} \coloneqq [\boldsymbol{W}, \boldsymbol{b}]$, using the cost function of Equation 3.10. In such manner, different variants of stochastic gradient descent algorithm are used and thus require the gradients of the cost function with respect to the weights and biases $\frac{\partial J}{\partial \boldsymbol{\rho}_j}$ for each layer, where $\boldsymbol{\rho}_j \coloneqq [\boldsymbol{W}_j, \boldsymbol{b}_j]$. Since the forward model $\mathcal{M} \equiv \mathcal{M}_{rr}(., \mathcal{N}n(.))$ with $\boldsymbol{\theta}$ being both the output of $\mathcal{N}n$ and the input of $\mathcal{M}_{rr}$, we can write $\frac{\partial J}{\partial \boldsymbol{\rho}_j} = \frac{\partial J}{\partial \boldsymbol{\theta}} \frac{\partial \boldsymbol{\theta}}{\partial \boldsymbol{\rho}_j}$. Then these two gradients are obtained as follows:

- The gradients $\frac{\partial J}{\partial \boldsymbol{\theta}}$ of the cost function with respect to the hydrological model parameters, computed by solving the numerical adjoint model of $\mathcal{M}_{rr}$;
- The gradients $\frac{\partial \boldsymbol{\theta}}{\partial \boldsymbol{\rho}_j}$ of the network output with respect to the weight and bias, computed using the chain rule of composite functions of $\mathcal{N}n$.

Eventually, the backward propagation for updating the weights and biases, using for instance Adam optimizer, is described in Algorithm 2.

Algorithm 2 Adapted backpropagation using Adam optimizer

▷ Randomly initialized weights and biases $\boldsymbol{\rho}^{(0)} = \left(\boldsymbol{\rho}_1^{(0)}, ..., \boldsymbol{\rho}_{N_L}^{(0)}\right)$
▷ Number of training epochs N_{epo}

for $i = 1..N_{epo}$ **do**
  ▷ Forward propagation over the spatial domain $\boldsymbol{\theta} \leftarrow \left[\left(\mathcal{N}n\left(\boldsymbol{D}(x), \boldsymbol{\rho}^{(i-1)}\right)\right)_{x\in\Omega}\right]^T$
  ▷ Initial gradient accumulation $\nabla A \leftarrow \nabla_{\boldsymbol{\theta}} J = \left(\frac{\partial J}{\partial \theta_1}, ..., \frac{\partial J}{\partial \theta_{N_\theta}}\right)$
  for $j = N_L..1$ **do**
    ▷ Gradient computation $\frac{\partial J}{\partial \boldsymbol{\rho}_j} \leftarrow \left(\frac{\partial \boldsymbol{\theta}}{\partial \boldsymbol{\rho}_j}\right)^T \nabla A$
    ▷ Updated gradient accumulation $\nabla A \leftarrow \nabla A.\left[W_j^{(i-1)}\right]^T$
    ▷ Updated weights and biases $\boldsymbol{\rho}_j^{(i)} \leftarrow \boldsymbol{\rho}_j^{(i-1)} - \eta \frac{m^{(i)}}{(1-\beta_1)\left(\sqrt{\frac{v^{(i)}}{1-\beta_2}}+\epsilon\right)}$ where:
    $m^{(i)} \leftarrow \beta_1 m^{(i-1)} + (1-\beta_1)\frac{\partial J}{\partial \boldsymbol{\rho}_j}\left(\boldsymbol{\rho}_j^{(i-1)}\right)$; $v^{(i)} \leftarrow \beta_2 v^{(i-1)} + (1-\beta_2)\left(\frac{\partial J}{\partial \boldsymbol{\rho}_j}\left(\boldsymbol{\rho}_j^{(i-1)}\right)\right)^2$
    $\beta_1 = 0.9$ and $\beta_2 = 0.999$ are the decay rates for first and second moments of gradients
    $\epsilon = 10^{-8}$ is a small scalar
    η is the learning rate that is a tuning parameter determining the step size of the optimization problem
  end for
end for