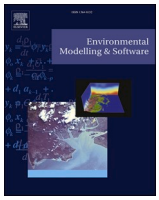
Environmental Modelling and Software 175 (2024) 105979

Contents lists available at ScienceDirect

Environmental Modelling and Software

journal homepage: www.elsevier.com/locate/envsoft



Optimizing sediment transport models by using the Monte Carlo simulation and deep neural network (DNN): A case study of the Riba-Roja reservoir

Danial Dehghan-Souraki [a,e,*], David López-Gómez [b], Ernest Bladé-Castellet [a], Antonia Larese [c,d], Marcos Sanz-Ramos [a,e]

[a] Department of Civil and Environmental Engineering (DECA), Universitat Politècnica de Catalunya, Barcelona, Spain
[b] Centro de Estudios Hidrográficos, Centro de Estudios y Experimentación de Obras Públicas (CEDEX), 28005 Madrid, Spain
[c] Università degli Studi di Padova, Padova, Italy
[d] Institute for Advanced Studies, Technical University of Munich, Garching, Germany
[e] International Center for Numerical Methods in Engineering (CIMNE), 08034, Barcelona, Spain

ARTICLE INFO

Handling Editor: Daniel P Ames

Keywords:
Sediment transport
Monte Carlo method
Artificial intelligence
Machine learning
Hydraulic modeling

ABSTRACT

This study emphasizes the importance of accurate calibration in sediment transport models and highlights the transformative role of artificial intelligence (AI), specifically machine learning, in improving accuracy and computational efficiency. Extensive experiments were carried out in the Riba-Roja reservoir, which is located in the northeastern Iberian Peninsula. The accumulated sediment volume (ASV) curve was used to calibrate these experiments. The optimal ASV curve was found to be very close to the experimental data, with only minor differences in upstream areas. The results revealed a consistent rate of sediment transport and settling. Furthermore, the study investigated the capabilities of deep neural networks (DNNs) in predicting ASV curves and observing variable performance. In essence, the study highlights AI's potential for enhancing sediment transport models.

1. Introduction

Sediments are solid particles that are transported and deposited by natural processes such as erosion, weathering, wind, water, and ice. There are different mechanisms for the transportation of sediments, and these are usually called "modes" of sediment transport. The more important modes are suspension (where sediments are carried within the water column by turbulent flow or currents), traction (where sediments are transported along the bottom of a waterway by rolling, sliding, or dragging), saltation (where sediments are transported in a series of short hops or bounces along the bottom of a waterway), and solution (where sediment are dissolved in water and transported as dissolved load) (van Rijn, 1993). The understanding of sediment transport mechanisms is important in hydraulics as well as other fields such as geology, civil engineering, and environmental management (Wood, 1997).

Sediment control is an important aspect of environmental management that is aimed at reducing soil erosion, sedimentation, and associated environmental impacts. There are several strategies for controlling the process of sediment transport, some of which may include sediment basins, silt fences, erosion control blankets, and grassed waterways. Numerical techniques are considered valid tools for studying such processes in detail. Thanks to the recent advances in computational engineering and their application to hydrodynamics, sediment transport, and morphodynamics, it is possible to understand and control sediments in various environments (Van Binh et al., 2022; Goldstein et al., 2019; Khosronejad et al., 2023). These models can simulate physical processes that govern sediment transport and deposition, predict sediment patterns, and assess sediment control strategies (Kantoush and Sumi, 2010; Minella et al., 2014).

The accurate calibration of the parameters of sediment transport models is essential to ensure a reliable representation of the real-world system being studied. There are several methods for calibrating these models, one of which is the Monte Carlo method. The Monte Carlo method involves the generation of a large number of model simulations with randomly selected parameter values within a specified range. These simulations are then compared to the observed data, after which the parameters that produce the best fit are selected. This process is repeated multiple times to account for the uncertainty in the observed data and improve the accuracy of the model. Although the Monte Carlo

* Corresponding author. Department of Civil and Environmental Engineering (DECA), Universitat Politècnica de Catalunya, Barcelona, Spain.
E-mail address: danial.dehghan@upc.edu (D. Dehghan-Souraki).

https://doi.org/10.1016/j.envsoft.2024.105979
Received 21 September 2023; Received in revised form 18 January 2024; Accepted 12 February 2024
Available online 20 February 2024

method is computationally expensive, it provides a comprehensive approach to model parameter calibration, which is why it is commonly used in sediment transport modeling (Hallouz et al., 2018; Lu and Chiang, 2019; Tonkin and Doherty, 2009). Moreover, the Monte Carlo method can also provide a probabilistic assessment of model uncertainty, which is useful for both decision-making and risk assessment (El Safadi et al., 2015; Zhang et al., 2019).

High-performance computing (HPC), and graphics processing unit (GPU) parallelization in particular, can help in coping with the high computational cost of the Monte Carlo method without losing much accuracy in model simplification. The GPUs are designed to handle massive volumes of data and perform various calculations simultaneously, making them ideal for Monte Carlo simulations. By distributing the workload across multiple GPUs, parallelization has the potential to significantly reduce simulation time, which can enable researchers to solve complex problems more efficiently. Furthermore, if implemented, HPC and GPU parallelization can offer substantial computational advantages over traditional CPU-based systems. In this study, we aimed to demonstrate the effectiveness of HPC and GPU parallelization in accelerating Monte Carlo simulations. The utilization of these technologies allowed us to address increasingly complex problems and achieve faster results. Our findings provide evidence of the benefits and future potential of HPC and GPU parallelization in scientific research. In the context of simulating shallow water flows using GPUs, de la Asunción et al., (2010) introduced a GPU-based model for the simulation of two-layer shallow water flows. The authors utilized CUDA-based GPUs to achieve efficient computations and demonstrated the model's capabilities in accurately solving various test cases. Building upon this work, de la Asunción et al., (2013) further discussed the use of the two-waves total variation diminishing-weighted average flux (TVD-WAF) method for the simulation of two-dimensional, one-layer shallow water systems on structured meshes, presenting an efficient GPU implementation of this method and showcasing its effectiveness in providing accurate solutions. By organizing the references chronologically, this revised section highlighted the progressive development of GPU-based models for shallow water flow simulations. They highlighted the benefits of using GPUs for this type of simulation, including their ability to handle large volumes of data and the speed of GPU processing. Further, they also described the specific techniques that they used to optimize their simulation, including thread and block organization and the use of shared memory. Brodtkorb et al. (2013), who provided a broader overview of GPU programming strategies and trends in GPU computing, discussed the history of GPU computing and its evolution from a specialized technology to a widely used tool for scientific computing. Additionally, they also described various programming paradigms for GPU computing and stressed the importance of optimizing algorithms for the specific architecture of GPUs. Moreover, Bell and Hoberock (2011) described Thrust, which is a productivity-oriented library for CUDA programming. They discussed the features of the library and how it can simplify and speed up GPU programming tasks. Finally, Castro et al. (2011) discussed using GPUs to simulate shallow water flow based on finite volume schemes. Their results demonstrated the efficiency of their GPU-based simulations and stressed the potential of this technology for the simulation of complex flow scenarios.

The integration of artificial intelligence (AI)—and, particularly, machine learning algorithms—in the calibration of 2D sediment transport models using the Monte Carlo method is an emerging and promising field of study. Machine learning algorithms and AI can be used to automate the selection process of the model parameter to reduce the computational time as well as improve the accuracy of the model results. For instance, machine learning algorithms such as neural networks support vector machines, and decision trees can be trained on the observed data and used to predict the best model parameters, along with understanding the model behavior to estimate the possible outcomes. This approach has been present in several recent studies (AlDahoul et al., 2021; Goldstein et al., 2019; Jimeno-Sáez et al., 2022; Sarker, 2021).

Artificial Intelligence has been utilized in various studies to estimate the suspended sediment in rivers. One of the earliest studies of this kind was conducted by Jain (2001), who developed a sediment rating curve model by using a feed-forward neural network (FFNN). The author found that the AI model produced better results than the conventional curve-fitting methods. In another study, Nagy et al. (2002) used a multilayer feed-forward neural network to predict the sediment load in three rivers by using eight input variables, including tractive shear stress and the velocity ratio. They found that the neural network produced satisfactory results compared to conventional equations for sediment load concentration. Kerem Cigizoglu and Kisi, (2006) proposed a method to improve the performance of the neural network by using the K-fold partitioning method to divide the data into sub-groups and train them individually. Alp and Cigizoglu (2007), who investigated the estimation of daily suspended sediment load by using both feed-forward back-propagation and radial basis function algorithms, found that the learning rate directly affected the performance of the algorithm and adding river flow data to rainfall data produced better results. Moreover, Rai and Mathur (2007) developed an event-based model for temporal variation of sediment yield by using a feed-forward back-propagation neural network. The authors found that the artificial neural network (ANN) model performed better than the linear transfer function model. Jothiprakash and Garg (2009), who used a multi-layer perceptron with a back-propagation algorithm to estimate the volume of sediment retained in a reservoir, found that ANN produced better results than conventional regression analysis. In recent years, there have been significant advancements in the application of AI for sediment transport modeling. For instance, a study by Latif et al. (2023) compared the performance of various AI-based models, such as Long Short-Term Memory (LSTM), ANN, and Support Vector Machine (SVM), in predicting sediment transport in the Johor river. The study found that LSTM outperformed other models, suggesting that deep learning models might be more effective than traditional machine learning models for this task. Moreover, a comprehensive review by Gupta et al. (2021) summarized various AI-based models for suspended sediment load (SSL) estimation, including ANN, SVM, and Neuro-Fuzzy (NF). The review found that these models performed well in different circumstances, and hybrid models using ANN with Extreme Learning Machine (ELM) or wavelet analysis showed promising results. Furthermore, a book chapter authored by Heddam et al. (2024) focused on AI modeling of sediment transport in Iranian rivers, demonstrating the global application of these techniques. These recent studies highlight the ongoing evolution and potential of AI in sediment transport modeling, addressing the limitations of earlier models and opening new avenues for future research.

The present study aimed to develop an efficient methodology for calibrating a 2D sediment transport model by utilizing the experimental data from the Riba-Roja Reservoir. The primary focus of this study was to improve the computational efficiency and parameter selection process of the calibration approach. To achieve this goal, the study employed DNNs to reduce computational costs and improve model performance. Unlike a previous study by Bladé Castellet et al. (2019), which utilized a CPU-parallelized sediment transport and a hydraulic model, the present study used a GPU-parallelized version of the same model for faster simulations. DNNs were picked due to their capacity to describe intricate patterns with more precision by capturing complex, nonlinear interactions in sediment transport processes. They were among the best options among other AI algorithms for estimating the ASV curves because of their potential in automated feature extraction from high-dimensional datasets and systematic learning.

One of the major differences between the study by Bladé Castellet et al., (2019) and this one is the approach used for selecting the parameters. In the former, the selection of the parameters was performed "manually" after defining specific objectives; however, in the latter, the selection of the parameters was entirely based on performance evaluation with no manual intervention. This approach allowed for a more

comprehensive evaluation of model performance and reduced the risk of biased parameter selection. Another significant difference between the previous and the present study is the computational domain used for the simulations. In the previous study, a simplified computational domain was used to reduce the computational cost in the first step of the calibration process, whereas the present study employed a more refined computational domain from the first step thanks to the speedup achieved by the GPU-parallelized model and the implementation of AI in the Monte Carlo simulation. This refinement from the beginning of the calibration allowed for a more straightforward and less iterative scheme for the process of calibration.

Using this comprehensive approach, this study aimed to provide an accurate calibration of the sediment transport model and advance the understanding of sediment dynamics in the Riba-Roja Reservoir. This is expected to contribute to the development of improved sediment transport modeling techniques that can be applied to other reservoir and water systems, ultimately supporting better management and conservation of natural resources.

Structurally, this study is divided into the following sections: material and methods, case study, results and discussions, conclusions, acknowledgments, Author CRediT Statement, and references.

2. Materials and methods

This section discusses the governing equations and sediment transport methods used in the Iber software package to simulate hydraulic and sediment transport processes (Bladé et al., 2014). Further, the section mentions the continuous improvements made to Iber, along with briefly introducing two new parallel implementations of the hydraulic module of Iber, namely, R-IBER (Sanz-Ramos et al., 2023), and Iber+ (García-Feal et al., 2018), which utilize the CUDA Fortran code for accelerated simulations. Later in this section, the study area is discussed. This refers to the Ebro River basin in the northeastern Iberian Peninsula. The section briefly mentions two bathymetric campaigns that were conducted to examine the sediment dynamics and bottom evolution in the Riba-Roja Reservoir tail. Finally, the section introduces the methodology that was utilized to calibrate the model by using the given experimental data sets.

2.1. Governing equations

The Iber software package was evaluated to conduct hydraulic and sediment transport simulations. The package was previously utilized to simulate the flow of water and hydrological processes in waterways (Bladé et al., 2014). Notably, the Iber software not only simulates water hydrodynamics processes but also contains multiple modules that are capable of addressing a wide range of processes, such as bedload and suspended sediment dynamics, water quality, and hydrological processes. The sediment transport numerical solver is integrated with the hydrodynamic module, which resolves the two-dimensional shallow water equations by using the finite volume method and the numerical scheme of Roe (1986). These equations, which are also referred to as the 2D Saint-Venant equations, along with the Boussinesq approximation to consider turbulence, can be mathematically expressed as follows:

$$\frac{\partial h}{\partial t}+\frac{\partial hU_x}{\partial x}+\frac{\partial hU_y}{\partial y}=0$$
$$\frac{\partial (hU_x)}{\partial t}+\frac{\partial}{\partial x}\left(hU_x^2+g\frac{h^2}{2}\right)+\frac{\partial}{\partial y}\left(hU_xU_y\right)=gh\left(S_{o,x}-S_{f,x}\right)+\frac{\partial}{\partial x}\left(\nu_t h\frac{\partial U_x}{\partial x}\right)+\frac{\partial}{\partial y}\left(\nu_t h\frac{\partial U_x}{\partial y}\right) \quad [1]$$
$$\frac{\partial \left(hU_y\right)}{\partial t}+\frac{\partial}{\partial y}\left(hU_xU_y\right)+\frac{\partial}{\partial y}\left(hU_y^2+g\frac{h^2}{2}\right)=gh\left(S_{o,y}-S_{f,y}\right)+\frac{\partial}{\partial x}\left(\nu_t h\frac{\partial U_y}{\partial y}\right)+\frac{\partial}{\partial y}\left(\nu_t h\frac{\partial U_y}{\partial y}\right)$$

Here, h is the water depth, U_x and U_y are the x and y components of the depth-averaged velocity, g is the gravitational acceleration, $S_{o,x}=-\partial z_b/\partial x$ and $S_{o,y}=-\partial z_b/\partial y$ are the spatial variation of the bed elevation (z_b) on the x and y directions, $S_{f,x}:=\left(n^2U_x\sqrt{U_x^2+U_y^2}\right)/h^{4/3}$ and $S_{o,y}:=\left(n^2U_y\sqrt{U_x^2+U_y^2}\right)/h^{4/3}$ are the energy dissipation in the x and y directions due to bed friction (which is computed using the Manning formula, where n would be Manning's roughness coefficient), and ν_t is the turbulent viscosity.

Iber has undergone continuous improvements, and its latest iteration features a suite of modules for various free-surface flow processes, including hydrological processes (Cea et al., 2022; Cea and Bladé, 2015; Sanz-Ramos et al., 2021), pollutant diffusion and propagation (Cea et al., 2016), the transport of large wood particles (Ruiz-Villanueva et al., 2014), and the eco-hydraulic model (Sanz-Ramos et al., 2019, 2023).

2.2. Sediment transport methods

Iber includes a sediment transport module that simulates bedload and suspended sediment transport. It is based on the results of velocity, depth, and turbulent viscosity fields computed by the hydrodynamic and turbulence modules. To simulate the suspended sediment transportation, the depth-averaged turbulent convection-diffusion equation was used. The equations concerning the sediment transport model can be formulated as follows (Cea et al., 2016):

$$\frac{\partial hC}{\partial t}+\frac{\partial hU_xC}{\partial x}+\frac{\partial hU_yC}{\partial y}=\frac{\partial}{\partial x_j}\left(\left(\Gamma+\frac{\nu_t}{S_{c,t}}\right)h\frac{\partial C}{\partial x_j}\right)+(Er-De) \quad [2]$$

Here, C is the depth-averaged concentration of suspended solids; Γ is the molecular diffusion coefficient for suspended solids; $S_{c,t}$, is the Schmidt number, which relates the turbulent diffusion coefficient that characterizes the distribution of turbulence intensity with the suspended turbulent diffusion coefficient; and De and Er are the deposition rate and entrainment rate, respectively. Their mathematical definitions are discussed below. First, the sediment conservation equation under the assumption of no bedload transport was used because of the existence of mostly fine and cohesive sediment particles in the study area. This equation computed the changes in the bed level Z_b by using the following terms:

$$(1-p)\frac{\partial Z_b}{\partial t}=De-Er \quad [3]$$

Er was estimated using the linear threshold model, which was initially suggested by Partheniades (1965) and Ariathurai and Arulanandan (1978) for cohesive soils.

$$Er=M\cdot\left(\frac{\tau_b}{\tau_{ce}}-1\right) \quad [4]$$

In this case, the erosion rate was determined by the difference between the shear stress τ_b and the strength (or erosion critical stress) τ_{ce}. Further,

is proportional to M, which represents the rate of erosion constant. The following widely used expression (Einstein and Krone, 1962), which considers a deposition critical stress τ_{cd}, used for the definition of De:

$$De = \left(1 - \frac{\tau_b}{\tau_{cd}}\right) \cdot W_s \cdot C_a \quad [5]$$

Here, C_a is a near-bed concentration derived from the Rouse profile, calculated according to Huybrechts and Villaret (2010). Moreover, the settling velocity W_s is determined by using the Van Rijn formula (Van Rijn, 1987), which is widely used in large-scale applications (Duan and Nanda, 2006).

2.3. CUDA Fortran code: R-IBER

R-IBER is a numerical tool that is based on the code of Iber and was originally developed for hydrodynamic and eco-hydraulic purposes in CUDA Fortran architecture for GPU computing. The existing code has been enhanced by including the sediment transport module of Iber into R-IBER. A code re-ordering and optimization were necessary to follow the main structure of CUDA programming. Unlike the eco-hydraulic module, which is post-process, this new module is fully integrated and works with hydrodynamics. The technical details of the software are provided in section.

The use of GPUs, which were initially designed for computer graphics, has been commended due to their parallel architecture that allows for several TFLOPS (1012 floating-point operations per second) of computational power in a single graphics card. The NVIDIA CUDA, which extends conventional programming languages to expose the capabilities of GPUs to programmers, is widely used in scientific applications (NVIDIA Corporation, 2023). It provides abstractions such as thread hierarchy, shared memory, and barrier synchronization to programmers. The NVIDIA GPUs employ single-instruction, multiple-thread (SIMT) architectures, which distribute threads in groups of 32 called warps. In a stream multiprocessor (SM), each warp is run in parallel and follows the same instructions. However, when a branch instruction is executed, a divergence occurs, resulting in the delay of threads in one pathway while others are executed and vice versa. The threads will converge once the branch is completed. To attain high throughput, it is important to consider this GPU characteristic and update the algorithms to eliminate redundant branching and data recognition, thus avoiding divergence.

Fig. 1 presents a simplified flowchart of R-IBER, which illustrates that the simulation data are transferred before the beginning of the simulation and when they need to be written to the hard disk. During each iterative loop, only the variables of the current timestep are transferred, as discrete GPUs have a memory that is distinct from the system memory. To transfer the data between the CPU and GPU, the PCI-Express bus is used, and the data are transferred from the system memory to the GPU memory. However, these transfers may become a bottleneck due to limited bandwidth and higher latency compared to the system memory accesses. Therefore, it is crucial to minimize the memory transfers between CPU and GPU as much as possible, as recommended by García-Feal et al. (2020).

It is also worth mentioning that to achieve maximum efficiency in variable allocation on GPU processors, it is recommended to use single-precision arithmetic operations whenever possible, which may entail round-off errors in the computation results. The choice between single-precision and double-precision arithmetic depends on memory usage, computational efficiency, and precision requirements. Double-precision arithmetic requires twice as much memory, which can impact memory-intensive applications and large datasets. Conversely, single-precision arithmetic can offer faster performance in scenarios where high computational throughput is crucial, such as real-time simulations or GPU processing. Additionally, single-precision arithmetic can provide sufficient accuracy for applications with less stringent precision

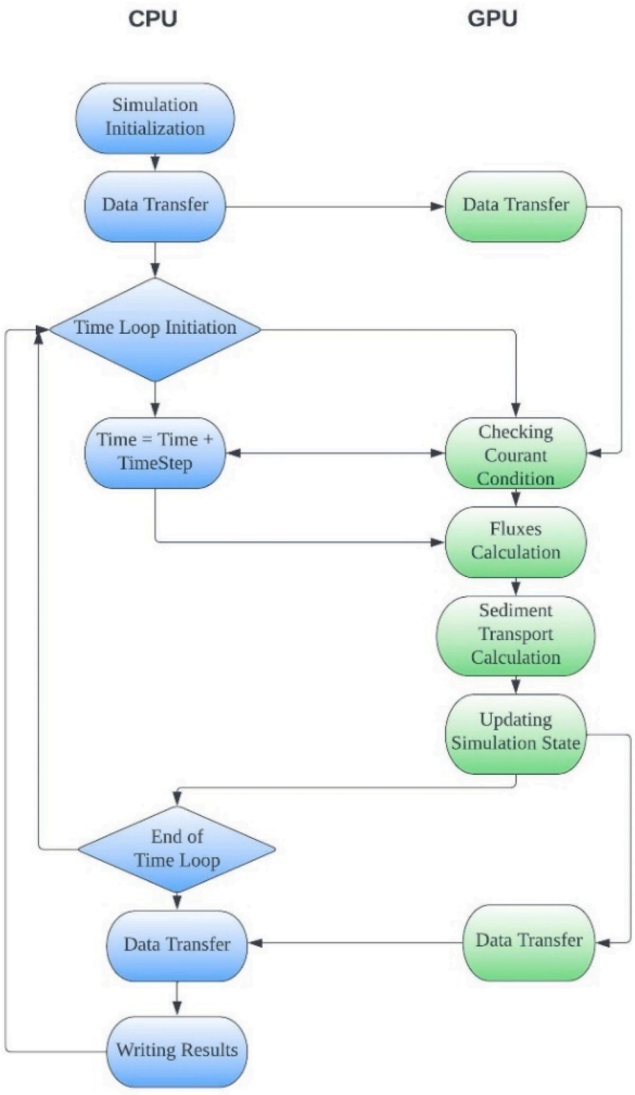


Fig. 1. General flow chart of the R-IBER code for a particular GPU computation.

requirements. Ultimately, the decision should consider the specific application, hardware limitations, and the right balance between memory usage, computational efficiency, and precision needs.

2.4. Hardware and characteristics

The simulations were performed using Iber v3.1, which includes the updated R-IBER code and is available for free distribution on www.iberaula.com. Both algorithms utilized the Roe first-order numerical scheme. The computations were conducted on an AMD RYZEN 5900HX CPU for the sequential version (CPU), while for the CUDA Fortran GPU version (R-IBER), computations were launched on the same CPU along with the GeForce® RTX 3070 laptop GPU, which are widely used devices for mid-range to high-end gaming laptops.

2.5. Accumulated sediment volume (ASV) curve

The accumulated sediment volume (ASV) curve is a graphical representation that shows the cumulative volume of sediment that has been deposited in a particular area over time. The ASV curve is calculated by integrating the sediment accumulation rate over a certain period and plotting the accumulated sediment volume against the distance or time.

The accumulated sediment volume is expressed in units of volume per unit area (typically in cubic meters or cubic feet per square meter or square foot).

The ASV is being used in the field of sedimentology and coastal engineering to evaluate the amount and distribution of sediment that has accumulated in a particular area (Baljyan et al., 2014; Bertrand-Krajewski et al., 2006; Phien and Arbhabhirama, 1979). Furthermore, it is particularly useful for understanding the long-term effects of coastal processes, such as erosion and deposition, as well as for planning coastal management and developmental projects. The ASV curve can also be used to evaluate the impact of human activities such as dam construction and sedimentation patterns (Sedláček et al., 2022). Additionally, the ASV curve can provide important information for the design and management of coastal structures such as breakwaters, jetties, and shoreline stabilization structures. Thus, the ASV curve is a valuable tool for understanding the sediment balance of a coastal system and can help in making informed decisions about the management and protection of these important coastal resources.

2.6. Utilized deep neural network

Deep neural networks (DNNs) are a type of neural network architecture that is used to predict a continuous target variable, given a set of input features. Compared to traditional neural networks, DNNs have more hidden layers and, thus, can learn more complex representations of the input data. This makes them well-suited for applications such as image recognition, speech recognition, and natural language processing, where the input data can be highly complex and varied. The basic equation for a three-layer DNN regression model is as follows:

$$\hat{y} = W_3(\sigma(W_2(\sigma(W_1X + b_1)) + b_2)) + b_3 \quad [6]$$

Here, $\hat{y}$ is the predicted target value, X is the input features, W_1, W_2, and W_3 are the weights for each layer in a three-layer network, b_1, b_2, and b_3 are the biases for each layer, and σ (sigmoid) is the activation function applied element-wise to the intermediate layer outputs. The weights and biases are learned through the training process, during which the network's prediction is compared to the actual target values, and the parameters are updated to minimize the prediction error, which will be further defined later in this section.

The number of hidden layers, the number of neurons in each layer, the activation function, and other hyperparameters are all choices that can be made in designing a DNN regression model. The goal is to find the optimal set of hyperparameters that provide the best prediction performance on the training and test data. To achieve this, techniques such as cross-validation, early stopping, and regularization can be used to prevent overfitting and improve the generalization performance of the model.

Fig. 2 presents a schematic demonstration of a four-layer DNN architecture designed for the purpose of prediction. The DNN comprises two sets of 54 neural nodes and two sets of 27 neural nodes. The decision to use these specific numbers for the layer sizes was influenced by a combination of factors such as the complexity of the problem, the size of the dataset, and architectural considerations. In general, larger layer sizes allow the DNN to learn more complex patterns and potentially capture more intricate relationships in the data. However, the choice of layer sizes often involves striking a balance between model capacity and overfitting risks, as excessively large layers may lead to overfitting on smaller datasets or increase computational complexity. Consequently, the selected layer sizes of 54 and 27 were determined based on prior experimentation and architectural considerations, with the aim to strike an appropriate balance between model capacity and practical feasibility.

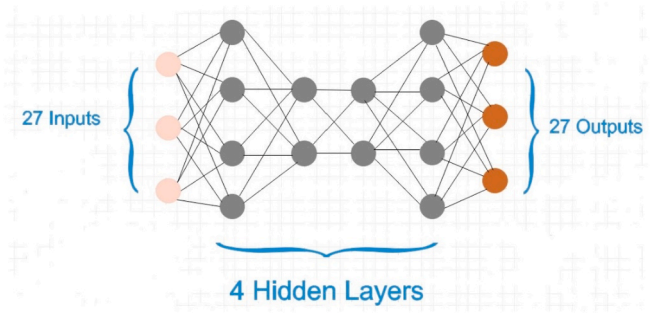


Fig. 2. Utilized DNN model pattern.

For the hidden layers of the DNN, the rectified linear unit (ReLU) activation function, which is commonly employed in regression tasks (Pomerat et al., 2019; Schmidt-Hieber, 2020), was utilized in this study as well. ReLU introduces nonlinearity by setting negative input values to zero while leaving positive values unchanged. This nonlinearity is vital in allowing the DNN to learn complex relationships and patterns within the data. Further, ReLU helps alleviate the vanishing gradient problem where the gradients become extremely small during back-propagation, hindering effective learning. By avoiding negative gradients, ReLU allows for more stable and efficient gradient propagation, enabling the DNN to learn and converge more effectively.

The choice of activation function in the output layer depends on the specific problem and desired output behavior. Linear activation is commonly used for predicting continuous values as it provides a direct mapping from the DNN's output to the desired output space. However, for problems requiring different output behaviors, such as binary classification (sigmoid activation), multi-class classification (SoftMax activation), or specific transformations of the output range (e.g., tanh activation), alternative activation functions can be employed. The selection of the activation functions in the DNN model (including ReLU in the hidden layers) and other appropriate functions in the output layer is based on their compatibility with the nature of the data, problem requirements, and established best practices in the field (Goodfellow et al., 2016; Hinton, 2012).

A maximum number of 1000 iterations, along with the convergence criterion of 1E-06 of relative error, is used for training. This convergence criterion is applied to monitor the change in the loss function, which represents the discrepancy between the predicted outputs of the DNN model and the true values in the training data. The training continues until the change in the loss function falls below the specified threshold, indicating that the model has converged to an acceptable level. In this study, the cross-validation method was used for evaluating the performance of the model. Approximately 20% of the 750 total datasets were excluded from the training process for the sake of the model performance evaluation using these test datasets.

2.7. Monte-Carlo simulations

The benchmark that was used to calibrate the sediment transport model of the R-IBER software was the experimental data extracted from the reservoir that suffered flooding in 2008. This was achieved by subtracting the bathymetric data of 2007 (i.e., before the flood) from the interpolated bathymetric data of 2008 (i.e., after the flood) to determine the depth of the eroded or deposited sediment particles during a flood event. The surface area of the reservoir was divided into 27 segments, and the sediment volume in each segment was calculated, which resulted in an ASV curve. This curve was used as a reference for sediment transport model calibration. The main steps of the calibration process are illustrated in the form of a flowchart in Fig. 3. As regards this flowchart to calibrate the model, the Monte Carlo simulation was used to identify the best combination of uncertain model parameters, including the erosion rate constant (M), the erosion critical shear stress (τ_{ce}), the deposition critical shear stress (τ_{cd}), and the particle diameter (D). Successful optimization of model parameters served as validation against real data, reflecting the iterative nature of the study. The chosen algorithm dynamically adapted based on evolving DNN performance,

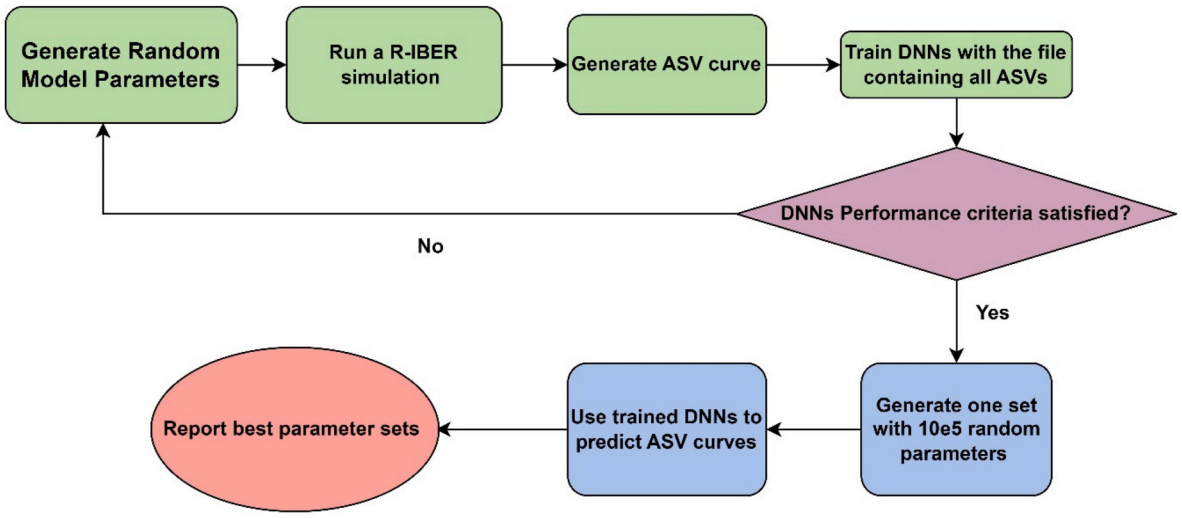


Fig. 3. A simplified process chart of performed sediment transport model calibration.

Table 1
Final values of the parameter ranges used to perform the Monte Carlo simulation.

Parameter	Lower Bound	Upper Bound
Erosion rate (M)	2.5e−08 m/s	8.9e−08 m/s
Erosion critical shear stress (τ_{ce})	0.2 Pa	1 Pa
Deposition critical shear stress (τ_{cd})	0.1 Pa	0.8 Pa
Particle diameter (D)	1e−05 m	3.4e−05 m

allowing flexibility to add more simulations if needed for a robust optimization process. To perform the simulation, a range of values for each parameter was selected based on previous literature and rough estimates. Initially, larger ranges were chosen to train the DNNs and obtain an estimate of the root mean square error (RMSE) values with the input parameters. Subsequently, a smaller range was selected based on the least 30% RMSE values. This smaller range was then used to train the DNNs further in those specific ranges, making the model more robust and accurate. Thanks to its extensive parameter variations, the Monte Carlo simulation combined with a trained DNN ensures an extensive evaluation of a variety of situations and model parameters, eliminating necessity for a separate sensitivity analysis. The selected parameter ranges for the Monte Carlo simulation are presented in Table 1. As Table 1 shows, the erosion rate (M) varied between 2.5e-08 and 8.9-e08 m/s, erosion critical shear stress (τ_{ce}) varied between 0.2 and 1 Pa, deposition critical shear stress (τ_{cd}) varies between 0.1 and 0.8 Pa, and particle diameter (D) varied between 1e-05 and 3.4e-05 m. As previously mentioned, the goal of the Monte Carlo simulation was to select the best set of parameters that fell into these ranges and corresponded with the lowest error metrics, which will be presented shortly.

The Monte Carlo simulation requires a large number of simulations to be run to cover all possible combinations of parameter values. To reduce the computational cost of the simulation, a series of 27 interconnected DNNs were trained to replace the R-IBER software in this specific case of the Riba-Roja Reservoir. The 27 DNNs were trained in sequence, starting with a regression DNN that took the four uncertain model parameters as inputs and predicted the first point on the ASV curve. The subsequent DNN models also took the four uncertain model parameters as inputs, along with the previous points on the ASV curve predicted by the previous DNN models. This interconnection between the DNNs was designed to allow the machine to gradually learn the process of calculating the ASV curve and take it into consideration when making predictions.

2.8. DNNs' performance evaluation

To evaluate the performance of the 27 DNN models, the cross-validation method was used. In this method, approximately 20% of the total datasets were separated from the training dataset to be used as test datasets for the DNNs performance evaluation and were not seen anywhere by the model during the training process. Each DNN model was trained to predict a specific point on the curve. Thus, the evaluation process involved assessing the performance of each DNN, along with the whole DNNs as a unit, to measure the entire ASV curve. Further, the training process involved optimizing the weights of the DNNs by using the training data and then evaluating the performance of the trained DNNs by utilizing the test data. The performance of each DNN model was measured using both mean absolute error (MAE) and RMSE metrics. Notably, MAE and RMSE are two commonly used evaluation metrics for assessing the performance of regression models, including regression DNNs.

MAE is a measure of the average magnitude of the errors between the predicted and the actual values in a dataset, and is defined as the mean of the absolute differences between the predicted and the actual values. The mathematical formula for MAE is as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y_i}| \quad [7]$$

Here, n is the total number of data points, y_i is the actual value of the i-th data point, and $\widehat{y_i}$ is the predicted value of the i-th data point. MAE is a non-negative value, and a lower MAE value indicates better model performance. In this study, for the sake of clarity of the performance evaluation of the DNNs, a modified version of MAE was defined and utilized as follows:

$$ModifiedMAE = 100 \cdot \left|\frac{y_i - \widehat{y_i}}{y_i}\right| \quad [8]$$

Here, y_i represents the actual value of the i-th data point, and $\widehat{y_i}$ represents the predicted value of the i-th data point. The result was multiplied by 100 to express the error as a percentage of the actual value. This modified definition may not be a standard definition of MAE, as MAE is typically defined as the absolute difference between the actual and the predicted values without scaling by a percentage.

Another commonly used evaluation metric for regression models is RMSE. It is a measure of the square root of the average of the squared differences between the predicted and the actual values. The mathematical formula for RMSE is as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad [9]$$

Due to its ease of calculation, RMSE is useful in interpreting differences between actual and predicted values, it has been chosen as the main metric for assessing the performance of DNN models. Additionally, compatibility of units with actual data improves the clarity of error interpretation when comparing DNN models to the global methodology. It is acknowledged that correlation-based measures such as R^2 (coefficient of determination) or NSE (Nash-Sutcliffe Efficiency) are beneficial, but it was realized that most DNNs exhibit consistently high R^2 values, which are generally close to 1. This suggests that the metrics may not offer any new information. The choice to exclude R^2 and NSE values from the overall evaluation of the predictive power of DNN models is supported by the little variances that have been found in particular DNNs.

Eventually, the results of the DNN models were compared with the results of the R-IBER for the same parameter combinations. The comparison was made to determine the accuracy of the DNN models in predicting the ASV curve and assess the feasibility of the potential replacement of the DNN models for the R-IBER in the case of the Riba-Roja Reservoir.

3. Case study

3.1. Study area

The Ebro River basin is approximately 85,000 km^2 in size and covers a large portion of the northeastern Iberian Peninsula. The basin includes around 107 reservoirs with a capacity of more than 1 hm^3, along with an irrigated area of about 800,000 ha. The Riba-Roja Reservoir is located just downstream of the Mequinensa reservoir (Fig. 4), which is the Ebro River's largest reservoir. Both reservoirs are situated in the Ebro River's lower reaches. The Riba-Roja Dam was constructed 115 km from the river mouth.

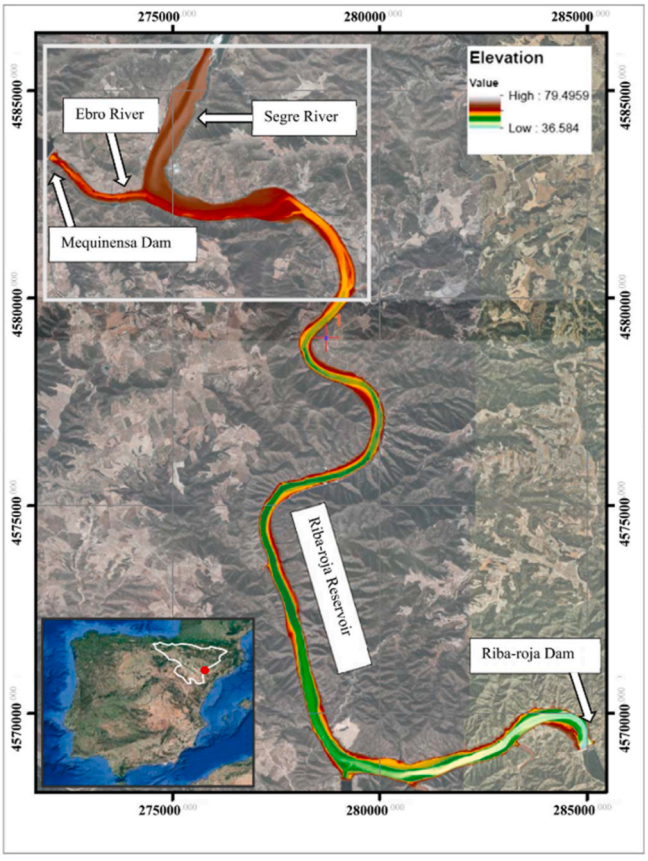


Fig. 4. Riba-Roja Reservoir's location and the 2007 multibeam bathymetry presented as a 2 × 2 m raster (color scale) covering the whole reservoir (source of background image: Google Earth).

Riba-Roja is a monomythic, long, and narrow reservoir in the Ebro River basin with quite a regular topology, encompassing a large portion of the northeastern Iberian Peninsula. Its current storage capacity is 210 hm^3 with a few weeks of resident time. The Riba-Roja Reservoir is closed downstream by a gravity dam with a crest of 76 m above sea level and a base of 16 m. This is supplemented by considerable discharge input from two tributaries, the Segre and Cinca rivers, which flow from the central Pyrenees and converge approximately 2 km before the Ebro River confluence at the reservoir's tail (Bladé Castellet et al., 2019).

The natural sediment discharge of the Segre River has increased dramatically over recent years as a result of the sediment released through the bottom outlets of the Barasona Dam on the Essera River between 1995 and 1997. The Barasona reservoir has had severe siltation difficulties since its formation, with 0.3–0.5 hm^3 of deposited sediment annually (Lobera et al., 2016). Moreover, more than 9 hm^3 of the sediment collected over decades in the reservoir base poured downstream as a result of the successful sediment release operations (Avendaño et al., 2000). This sediment is currently approaching the Riba-Roja Reservoir's tail. The Ebro River, which flows out of the Riba-Roja Reservoir, contributes an average of 8009 hm^3 annually (Dolz et al., 2009). Downstream of the Riba-Roja, the flow of the Segre River increases the contribution to 14,069 hm^3. The Ebro River delivers 59% of the total discharge into the Riba-Roja Reservoir, whereas the Segre River transports the remaining discharge (Dolz et al., 2009). The sediments exhibit a silt-clay texture along nearly the whole reservoir extension, with silt (4 μm < φ < 63 μm) percentages ranging from 56% to 74% and clay (φ < 4 μ m) percentages ranging from 18% to 43%, depending on the region (Roura Carol et al., 2008).

The Riba-Roja Reservoir is an important water resource that provides benefits to the surrounding communities. For this reason, it is extremely important to control its sediment accumulation and erosion processes. Effective sediment transport management is crucial for ensuring the long-term sustainability of the reservoir and the ecosystem it supports. By controlling the movement of sediment, the reservoir's storage capacity can be maintained and its water quality can be preserved. Moreover, a well-designed sediment transport management strategy can reduce the risk of flooding, protect against shoreline erosion, and improve navigation conditions for boats. Furthermore, preserving the health of the ecosystem that relies on the reservoir's water quality is essential for maintaining the biodiversity and the overall ecological balance of the area. Ultimately, a comprehensive sediment transport management strategy is essential for ensuring that the Riba-Roja reservoir continues to provide benefits to the surrounding communities for generations to come.

3.2. The 2008 flood

The present study was based on two bathymetric campaigns that were conducted to investigate the sediment dynamics and bottom evolution in the Riba-Roja Reservoir tail. The first bathymetric campaign, which was conducted in the autumn of 2007, covered the entire reservoir by using a multibeam sensor that provided continuous bathymetry (Fig. 5), while the second campaign was executed in December 2008 after a significant flood event occurred during May–June 2008, for which a single-beam sensor was used due to resource constraints, and the data were gathered along a path following the thalweg of the river and several perpendicular cross-sections (Bladé Castellet et al., 2019). The single-beam sensor, or single-beam echo sounder, measures the water depth using a single sound wave emitted from the sensor and reflected from the seafloor. The time taken for the sound wave to travel to the seafloor and back is used to calculate the water depth. S-Notably,

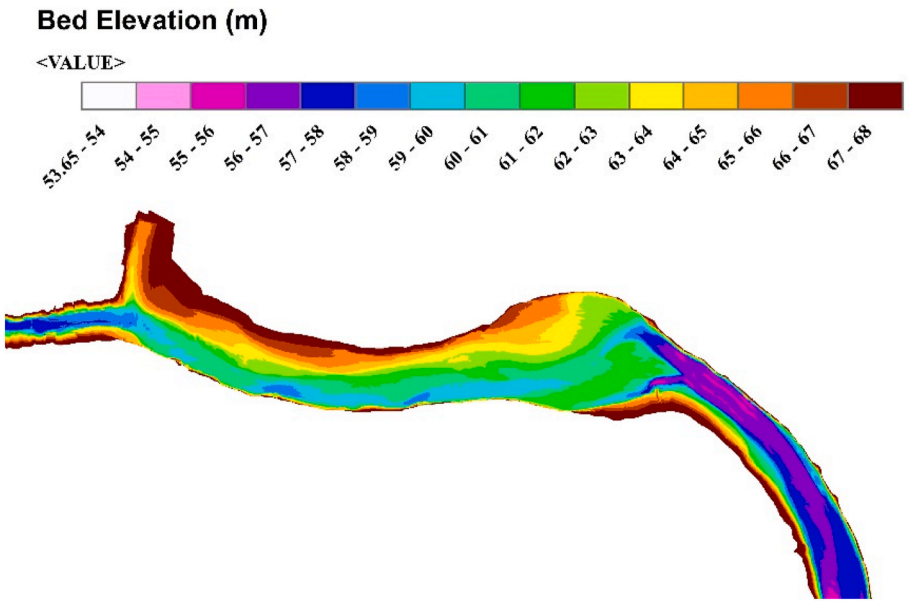


Fig. 5. Bathymetry of the Riba-Roja Reservoir in 2007 (before the flood).

single-beam sensors provide data only along a single line or path, usually along a vessel's track or predefined cross-sections, unlike multibeam sensors that offer continuous bathymetry. Despite providing less dense and continuous data compared to multibeam sensors, single-beam sensors are commonly used in bathymetric surveys due to their simple and cost-effective design.

The May–June 2008 flood entered the Riba-roja reservoir from the Segre river ($Q_{in,1}$) and the Mequinensa dam ($Q_{in,2}$), resulting in a combined hydrograph entering into Riba-roja reservoir with a peak discharge of 1800 m^3/s. The mean discharge into the reservoir during 11 days was greater than 1400 m^3/s, compared to the mean average inflow of around 260 m^3/s. At the beginning of the flood event, the water level was at 59 m. The Riba-roja dam was operated to maintain a constant water level for hydropower generation (Q_{out}), resulting in a roughly similar inflow and outflow to that of the reservoir. These inlet and outlet flow hydrographs were used as boundary conditions in the calibration of the numerical model, as shown in Fig. 6.

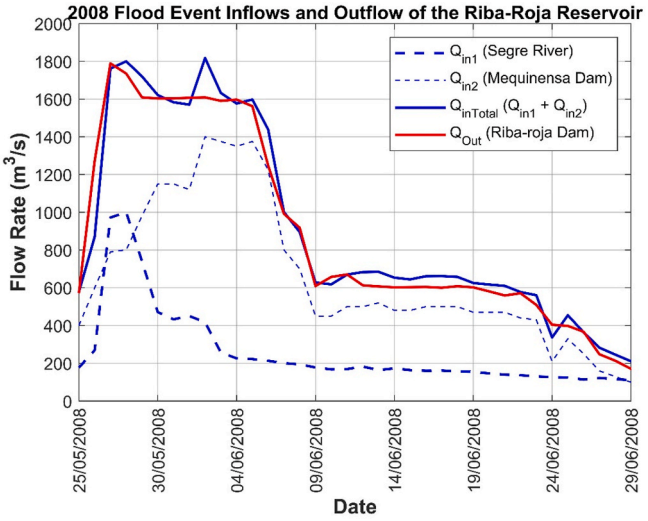


Fig. 6. Hydrographs of the different flows of the Riba-Roja Reservoir during the 2008 flood event (inflows: Segre River and Mequinensa Dam; outflow: Riba-Roja Dam).

The first bathymetric campaign produced a digital terrain model in a raster format with a cell size of 2 × 2 m, while the second campaign's results were converted to a raster format, containing data along a series of transversal sections and a longitudinal profile of the width covered by the sensor (Fig. 7). No other significant flood events occurred between the two campaigns, allowing for an assessment of any morphological changes in the reservoir bottom due to the May–June 2008 flood.

Fig. 7 presents a contour plot of the bed elevation bathymetry of the same parts of the Riba-Roja Reservoir after the 2008 flood. The plot includes a legend that shows the bed elevation value of each color in the plot in meters. However, the plot is not a complete bathymetry, as it only covers some parts of the reservoir as vertical lines and a continuous baseline (which represents the main path of the boat used to collect the data on different vertical lines). The vertical lines represent the sections where the data was collected by the boat perpendicular to the baseline, indicating the flow path on the reservoir. These features are visible in the plot and need to be considered during the interpolation process to obtain a complete and accurate bathymetry of the reservoir.

Several attempts were made to interpolate the given information. Unfortunately, these were affected by the data on the baseline rather than by the data from the vertical lines. Therefore, hardlines were manually generated to perform the manual weighting of the interpolation across the flow path. Fig. 8 presents the same bed elevation bathymetric plot from 2008 but with multiple green hardlines plotted, which connect the vertical lines in the 2008 bathymetry. These hardlines are 3D lines that connect two actual 3D points from the bathymetry of 2008, and the values between the endpoints of each line are interpolated using the bilinear method. This process was performed to obtain the actual data from the 2008 bathymetry along with some 3D hardlines, which interpolate between its vertical sections to have more points in the missing parts of the 2008 bathymetry. Further, Fig. 9 presents a closer look at the upstream part of the reservoir as a sample section, indicating the triangular irregular network (TIN) utilized for the interpolation, whereas Fig. 10 presents the results of the interpolated bathymetry of 2008 with the assistance of the TIN presented in Fig. 8. The TIN was used to interpolate between each of the near three points (i.e., triangles) by using the bilinear method, thereby providing a complete interpolated bed elevation from the 2008 bathymetry (Esri, 2021).

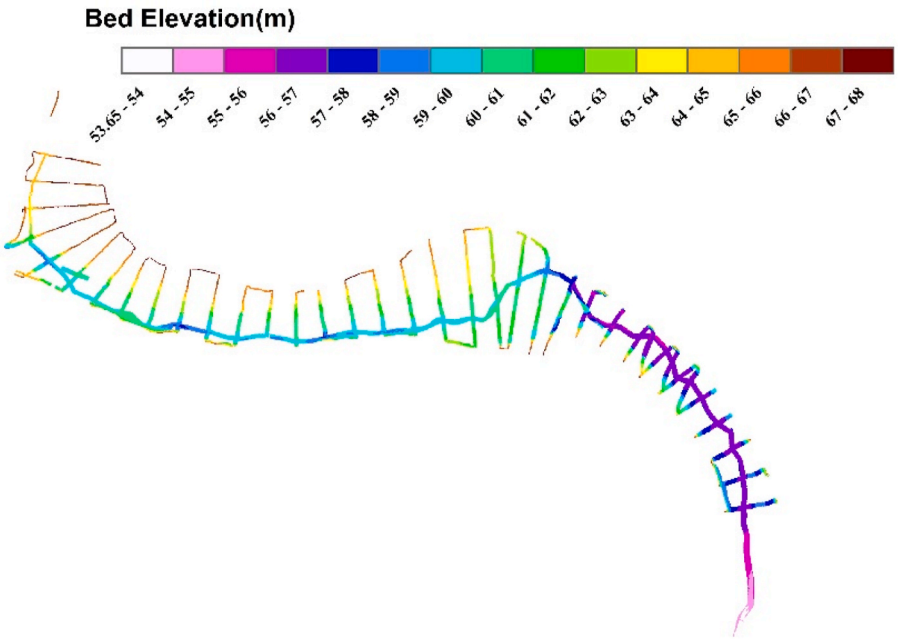


Fig. 7. Bathymetry of the Riba-Roja Reservoir in 2008 (after the flood).

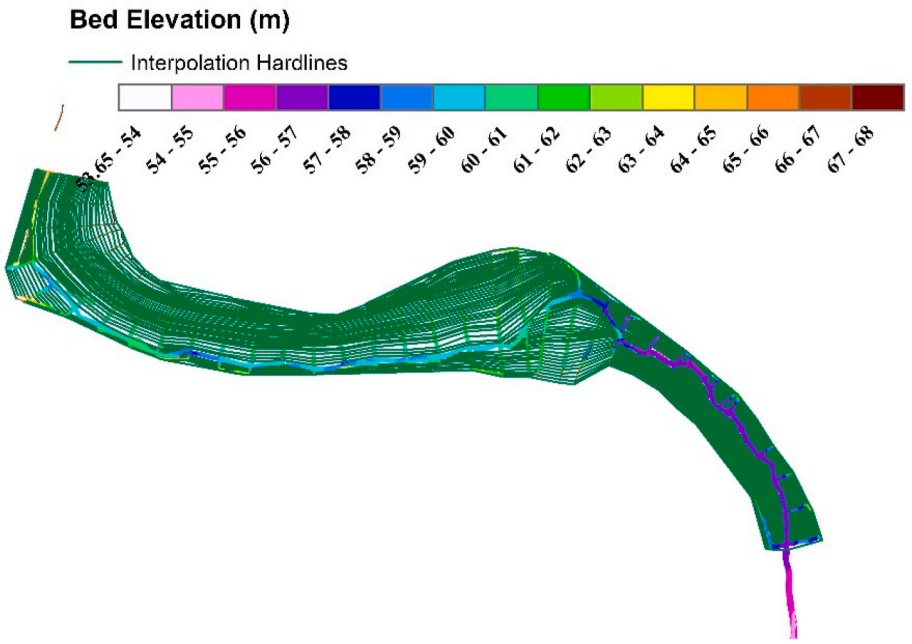


Fig. 8. Schematic of the hardlines generated for the interpolation process of the 2008 bathymetry.

4. Results and discussion

This section presents the results of the Monte Carlo simulation used to calibrate the Riba-Roja sediment transport model. The simulation compared the different accumulated sediment volume (ASV) curves predicted by the model by using random input parameters to an observed values ASV curve. Further, the simulations also provided insights into the accuracy of the model in predicting sediment transport in the upstream areas and emphasized the strengths and weaknesses of the model. The best-fitted ASV curve with the lowest RMSE value was identified, and the optimal parameter combination for the sediment transport model was explored. Additionally, the individual and global performances of the DNN model were evaluated using different graphs, which are presented at the end of the section.

4.1. Best curves predicted by the DNN

Fig. 11 presents the results of the Monte Carlo simulation used to calibrate the Riba-Roja sediment transport model. The main graph shows the comparison between five different ASV curves predicted by the model with random input parameters, along with the experimental ASV curve in the dashed blue line. The x-axis represents the distance from the head (i.e., upstream) of the reservoir, while the y-axis represents the ASV in cubic meters. The ASV curves exhibit different patterns as they progress downstream. The blue experimental curve shows a rapid increase in ASV at the beginning and then a steady growth,

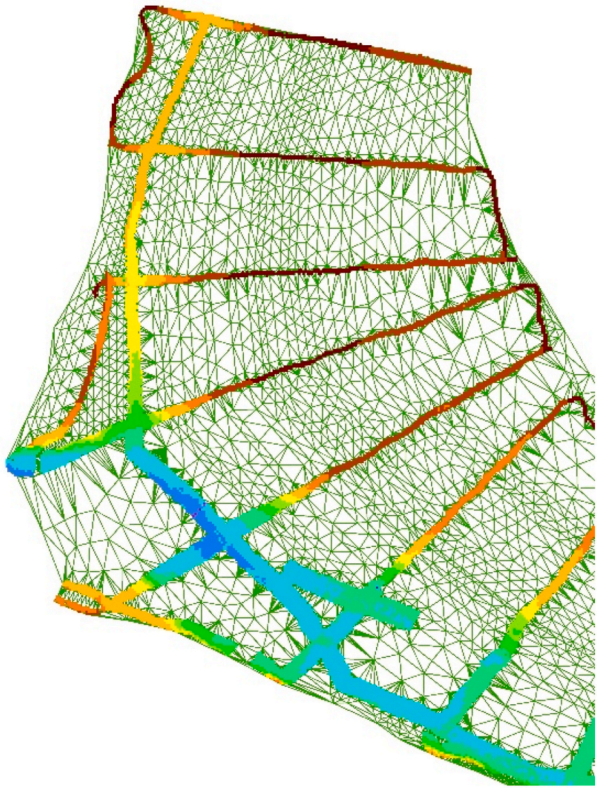


Fig. 9. Close-up image of the triangular irregular network (TIN) created for interpolation of the 2008 bathymetry.

whereas the colored ASV curves show a more gradual increase with some variations. The inset plot in the upper right corner of the figure zooms in on the best-fitted ASV curve with the lowest RMSE value (as shown in Table 2), which is represented as ASV 1 in the main graph and closely approximates the experimental data.

Upon closer inspection of the main graph, it can be observed that the predicted ASV curves generally match well with the experimental curve—especially in the mid-to downstream regions—while exhibiting certain discrepancies in the upstream areas. Specifically, the ASV curves tend to deviate from the experimental curve in the first 3 km from the head of the reservoir where the experimental curve demonstrates a more rapid increase in sediment deposition. This suggests that the model may be less accurate in predicting sediment transport in the upstream areas, which may be due to factors such as sediment characteristics, hydraulic conditions, or measurement errors in the experimental data. Overall, the simulation demonstrated that the model can reasonably predict sediment deposition in the reservoir with some room for improvement in the upstream regions.

Moreover, the variation of the predicted ASV curves with the

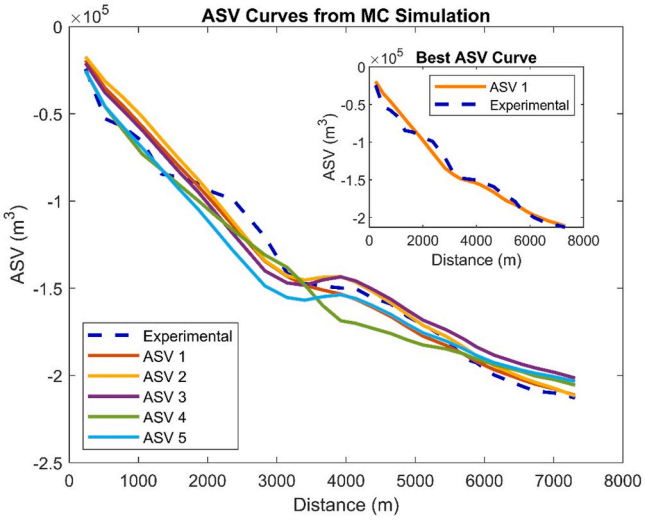


Fig. 11. 5 Sample ASV curves resulted from the Monte Carlo simulation along with the best-fitted ASV curve (inset plot).

Table 2
Model parameter values with the RMSE value for five sampled ASV curves (the best combination being named ASV 1) calculated in the Monte Carlo simulation.

	M [m/s]	τ_{ce} [N/m^2]	τ_{cd} [N/m^2]	D [m]	RMSE [m^3]
ASV 1	3.168e-08	0.100	0.490	1.191e-05	7626
ASV 2	2.791e-08	0.453	0.397	1.823e-05	8853
ASV 3	3.543e-08	0.651	0.501	1.804e-05	9700
ASV 4	5.134e-08	0.134	0.999	1.067e-05	10,422
ASV 5	4.438e-08	0.578	0.630	1.480e-05	11,079

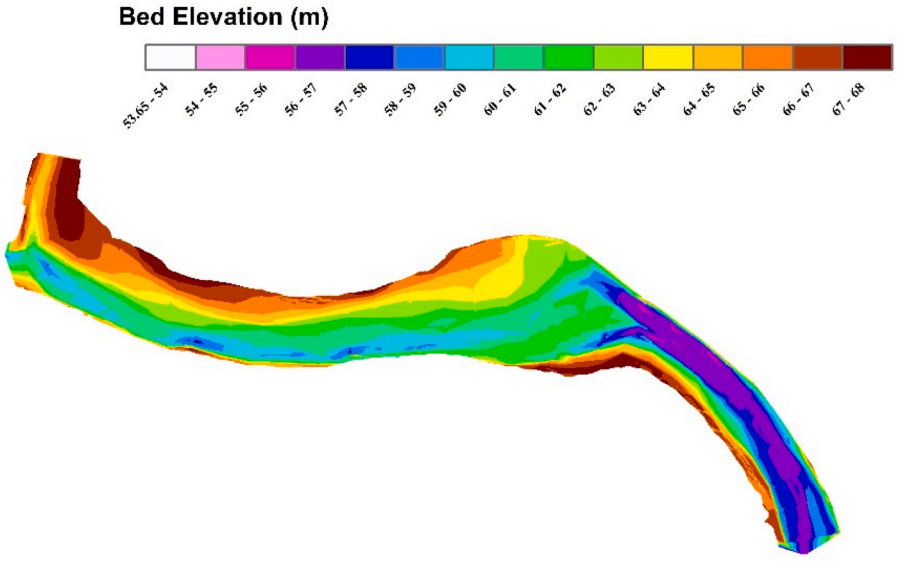


Fig. 10. Bed elevation values calculated using the final interpolation raster of the 2008 bathymetry.

distance from the head of the reservoir can provide insights into the sediment transport dynamics in the system. The gradual increase in ASV from the head towards the tail of the reservoir indicates that sediment deposition is mainly controlled by flow velocity and sediment concentration. However, the observed variations in the ASV curves may suggest the presence of other factors such as turbulence, flow eddies, or sediment supply variations. These factors may affect the sediment transport and deposition differently at different locations, leading to the observed differences in the ASV curves. Overall, the presented plot provides a useful visualization of the sediment transport model calibration results, emphasizing the model's strengths and weaknesses in predicting sediment deposition in the reservoir.

4.2. Choosing the best model parameters

From Table 2, it can be observed that the best parameter set was ASV 1, which had the lowest RMSE value among all ASV curves. The model parameters for ASV 1 were as follows: $M = 3.168\text{e-}08$ m/s, $\tau_{ce} = 0.100$ N/m^2, $\tau_{cd} = 0.490$ N/m^2, and $D = 1.191\text{e-}05$ m. This parameter set was able to capture the sediment transport dynamics more accurately than the other parameter sets. Further, it is noteworthy that the parameter values for ASV 1 fell within the range of the other parameter sets, which suggests that the model can be used for other similar scenarios as well.

The parameter values for ASV 1 show that the sediment transport rate was low ($M = 3.168\text{e-}08$ m/s) and the critical shear stress for erosion τ_{ce} was 0.100 N/m^2, which was also relatively low. Further, the critical shear stress for deposition τ_{cd} was 0.490 N/m^2, which indicates that sediment particles require higher shear stress to be deposited than to be eroded. Additionally, the settling velocity (D) was 1.191e-05 m, which indicates that sediment particles settle at a slow rate. These parameter values suggest that the sediment transport process in the Riba-Roja Reservoir during the 2008 flood was characterized by a low sediment transport rate and slow settling of sediment particles. Overall, the results suggest that the parameter set for ASV 1 can be considered as the optimal parameter combination for the sediment transport model in the Riba-Roja Reservoir during the 2008 flood.

4.3. Evaluating the DNN model performance

Fig. 12 depicts the change in the RMSE values of the predicted ASV curves by using all 27 DNNs versus the number of test datasets used to evaluate the performance of the global regression DNN. The x-axis represents the number of the total dataset used as test datasets, whereas the y-axis represents the RMSE values.

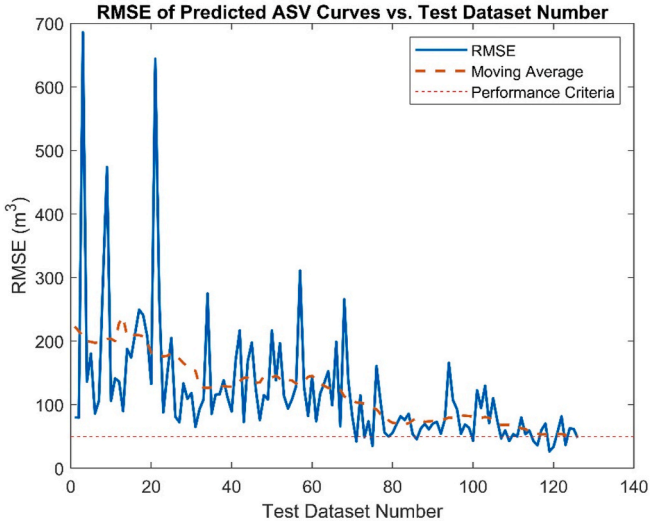


Fig. 12. Global DNN model performance plot.

Fig. 12 illustrates that increasing the number of test datasets results in decreasing RMSE values, which points to the reliable predictive performance of the DNN. However, the rate of decrease in the RMSE values appears to slow down beyond a certain threshold (RMSE = 50 m^3), suggesting a limited benefit from using additional test datasets. Selecting an appropriate number of test datasets is crucial to ensure reliable performance evaluation without unnecessary computational complexity. Additionally, the graph demonstrates that RMSE values stabilized after a few test datasets, indicating the consistent performance of the DNN without overfitting the training data. The threshold of 50 m^3 was chosen based on a safe fraction (1%) of the minimum RMSE value (approximately 7600 m^3) from the Monte Carlo simulation (ASV 1 in Table 2). Furthermore, considering the fluctuation in error values with increasing test dataset numbers (Fig. 12), opting for a smaller value was found to provide even greater safety in light of these variations.

Thus, the presented graph provides valuable insights into the performance evaluation of a regression DNN. By selecting an appropriate number of test datasets and optimizing the DNN network's hyperparameters and features, the predictive accuracy of the network can be further improved, although with a significant computational cost. It is noteworthy that RMSE values are a single numerical measure of prediction accuracy, and it is important to interpret them in the context of the specific simulation and the error being evaluated. Therefore, it may be beneficial to supplement RMSE values with additional evaluation metrics or visualizations to provide a more comprehensive assessment of the simulation accuracy.

Fig. 13 presents boxplots that provide an overview of the performance of individual DNNs. In the upper plot, blue boxes represent the interquartile range (IQR) of each test dataset, horizontal red bars represent the median of the error, and the red crosses outside the box represent the outliers. In the lower plot, each bar represents the RMSE value of the whole test dataset. These boxplots offer a visual representation of the distribution of MAE and RMSE values across the different DNNs and datasets. They stress both the average performance and the variability in performance for each data point separately. By analysing the boxplots, researchers can identify which DNN and dataset exhibit the largest errors, thereby gaining insights into the factors that may be limiting the model's performance. Adjustments to the model architecture or training process can then be made to improve accuracy. These adjustments may involve modifying the layers and their sizes or introducing regularization techniques to address overfitting. For datasets with consistently higher errors, refining data pre-processing, handling outliers, or considering alternative augmentation methods can be beneficial.

In the upper plot presented in Fig. 13, which represents the modified MAE values, the boxplots reveal that the individual DNNs exhibit a wide range of performance across the different datasets. Some DNNs consistently achieve low errors, while others show more variability. Generally, the MAE values are higher for the first three DNNs, with the first DNN having the highest median percentage error of approximately 0.45%, which is considered relatively low but still represents the maximum value among all DNNs. This struggle of the first three DNNs (mostly the first and third) in predicting real values may be due to fewer input parameters involved in their training process. However, as the DNN number increases, in general, the prediction error decreases. Further, the boxplots suggest that there are several datasets where some DNNs perform poorly, with MAE values being significantly higher than the median. These outliers may indicate data points that are particularly challenging for the DNNs to be modeled accurately.

The lower plot presented in Fig. 13 shows the RMSE values of the predicted values for each DNN across all test datasets. Unlike the modified MAE graph, which demonstrates the performance of each DNN on every single test data point separately, the RMSE graph represents the performance of each DNN on the entire test dataset. This means that the RMSE graph is more affected by outliers, as they are included in the final value of the error. Upon examining the RMSE graph, it can be observed

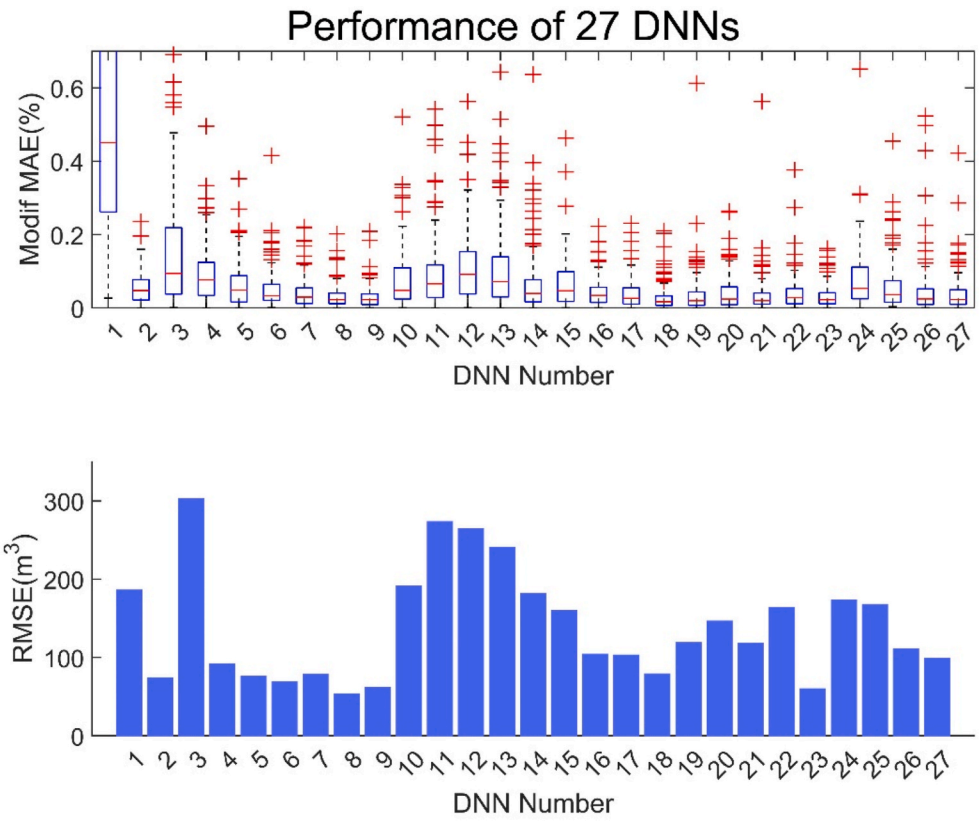


Fig. 13. Individual DNN performance plot.

that the first and third DNNs struggle to predict real values, although to a lesser extent than the modified MAE graph. As mentioned earlier, this may be due to the effect of outliers on other DNNs' performance, which normalizes the RMSE error bars compared to the modified MAE boxplots.

It is worth noting that the maximum RMSE error corresponds to the third DNN, with a value of roughly 300 m^3, which is nearly three orders of magnitude lower than the experimental ASV values presented in Fig. 11. Further, comparing the maximum RMSE value of the individual DNNs with the RMSE value derived from the best curve introduced by the Monte Carlo simulation (as mentioned in Table 2) suggests that the maximum prediction error caused by each DNN would be less than 5% of the error value reported for the best-predicted curve. This indicates that the performance of the DNNs is unlikely to have a significant effect on the Monte Carlo simulation process, as the DNNs are predicted to produce values that are close to the real data reported as R-IBER results.

Overall, based on the comparision of the two DNN performance evaluation plots, it appears that most of the DNNs can meet the performance criterion, with the presence of only a few outliers that cause higher RMSE values. However, it is important to note that the performance criterion is somewhat arbitrary, and the specific threshold value may depend on the context and application of the DNN. Additionally, the performance criterion only considers one aspect of the DNN's performance (i.e., error), and there may be other factors to consider as well (e.g., computational efficiency and interpretability).

4.4. *Performance and the limitaions of the proposed methodology*

The accuracy of the current methodology utilizing DNNs for calibrating a sediment transport model can be justified based on the results presented in Fig. 13, which displays boxplots demonstrating the distribution of MAE and RMSE values for individual DNNs across the different datasets. These boxplots provide valuable insights into the variability in the performance of the DNNs, with some DNNs consistently achieving low errors while others exhibit more variability. Thus, adjustments to the model architecture or training process can be made based on these insights to improve accuracy.

As for computational time, the current methodology requires approximately 750 simulations to train a robust DNN model, with each simulation taking about 35 min using R-IBER, which is a GPU-parallelized code. However, after training, the Monte Carlo simulations, which consist of approximately 100,000 simulations to cover every possible uncertainty in model parameter combinations, take almost no time. This indicates that the computational time for performing the Monte Carlo simulations using the trained DNNs is significantly reduced compared to traditional methods that rely solely on HPC clusters for Monte Carlo simulations. To further compare the computational time of the current methodology with other possible approaches, Table 3 is provided below, considering a common GPU cluster with 20 NVIDIA A100s as a reference for comparison.

The computational time estimates provided in Table 3 are approximate and may vary depending on various factors such as hardware configuration, simulation setup, and dataset size. As Table 3 shows, the

Table 3
Computational time comparison for the calibration of the sediment transport model between the different methodologies.

Methodology	Training Time (hours)	Typical Monte Carlo Simulation (20,000 cases) (hours)	Intensive Monte Carlo Simulation (100,000 cases) (hours)
Current Methodology	437	Negligible	Negligible
HPC Cluster (Consisting of 20 NVIDIA A100 GPUs)	N/A	582 (estimated)	2910 (estimated)

current methodology utilizing trained DNNs for Monte Carlo simulations significantly reduces the computational time compared to traditional methods that rely solely on HPC clusters. The training time for the DNNs is relatively low (about 437 h) compared to the estimated time required for performing typical and intensive Monte Carlo simulations (with 20,000 and 100,000 cases, respectively) by using HPC clusters (582 and 2910 h, respectively). Depending on their approach and implementation, other possible methodologies may also require similar or higher computational time.

In summary, the proposed methodology leveraging Deep Neural Networks (DNNs) for sediment transport model calibration exhibits promising performance in terms of accuracy and computational efficiency. By significantly reducing the computational time required for Monte Carlo simulations, the methodology offers a more efficient approach compared to traditional methods relying solely on High-Performance Computing (HPC) clusters. The potential for further acceleration exists by combining the strengths of both traditional and current methodologies—training DNNs using HPC clusters and subsequently utilizing the trained DNNs for Monte Carlo simulations on local machines or edge devices. This hybrid approach presents an avenue for achieving faster results without compromising accuracy. However, it is imperative to acknowledge inherent limitations in the methodology.

The current methodology, while demonstrating generality and applicability across various modeling scenarios, faces challenges rooted in its two-dimensional representation of sediment particle transport. Real-world sediment transport is inherently three-dimensional, involving complex interactions and motions, such as turbulence and temperature stratification. The simplification of assuming uniform spatiotemporal distribution of parameters introduces a degree of abstraction that might lead to inaccuracies in predicting sediment dynamics. The use of single-beam bathymetry for ASV curve extraction, accompanied by interpolation to fill information gaps, adds complexity and potential discrepancies. Additionally, the calibration procedure's reliance on a specific extreme event—the 2009 flood—may introduce a level of case specificity. Deviations from these conditions could impact the model's predictive accuracy, highlighting a potential limitation of the study.

Despite these challenges, the hybrid methodology shows promise in achieving significant computational time savings and improved efficiency compared to traditional approaches. Careful validation and performance assessment are essential to ensure the reliability of results, considering the outlined limitations in accurately capturing the complexity of real-world sediment transport phenomena.

5. Conclusions

This study aimed to improve sediment transport modeling and calibration by integrating the Monte Carlo method with AI and machine learning algorithms. The methodology involved the use of DNNs to replicate the results of the Iber software package and then incorporate them into the calibration process using the Monte Carlo method.

The case study conducted in the Riba-Roja reservoir provided valuable insights into sediment dynamics and the performance of the developed models. The Monte Carlo simulation, coupled with the DNNs, facilitated the generation of ASV curves, which were then compared with the experimental data. The results showed a high degree of accuracy, especially in the middle and downstream regions of the reservoir. However, discrepancies were observed in the upstream regions, indicating the need for further refinement and calibration to improve model performance in these regions.

In addition, the successful application of AI algorithms and machine learning techniques, in conjunction with the Monte Carlo method, demonstrated their potential to improve the accuracy and computational efficiency of sediment transport models. By replicating Iber's results using DNNs, the study demonstrated the capabilities of AI in simulating sediment transport processes. The subsequent integration of these AI-driven models into the calibration process further improved the reliability and predictive power of the sediment transport model. To illustrate this, Table 4 is provided, with an overview of the computational time variations across different methodologies for sediment transport modeling in the Riba-Roja Reservoir during the 2008 flood event. The table compares the simulation times for 1 simulation, 20,000 simulations, and 100,000 simulations using different methodologies: sequential code (CPU), sequential code combined with AI (CPU + DNN), GPU parallelized code (CPU + GPU), and GPU parallelized code combined with AI (CPU + GPU + DNN). The results show that the use of AI-driven models, especially in combination with GPU acceleration, leads to significant reductions in computation time. The CPU + GPU methodology resulted in speed-ups higher than 10 when compared to the sequential code methodology. Moreover, the integration of DNNs into the modeling process further improved computational efficiency, with seed-ups between 26 and 131 with CPU + GPU + DNN compared to CPU + GPU methodology, which represents overall speed-ups between 355 and 1740. These results highlight the benefits of using AI and GPU methods to achieve more efficient and time-saving procedures to model sediment transport.

Table 4
Computational time variations across different methodologies (sequential code (CPU), sequential code combined with AI (CPU + DNN), GPU parallelized code (CPU + GPU), and GPU parallelized code combined with AI (CPU + GPU + DNN)) for sediment transport modeling in Riba-Roja Reservoir during the 2008 flood event.

Utilized Methodology	1 simulation time (s)	20,000 simulations time (s)	100,000 simulations time (s)
CPU	2.8E04	5.7E08	2.8E09
CPU + DNN	N/A	2.2E07	2.2E07
CPU + GPU	2.1E03	4.2 E07	2.1E08
CPU + GPU + DNN	N/A	1.6E06	1.6E06

In terms of directions for future research, the results have shown that there is still a field to improve the sediment transport model in the upstream end of the Riba-Roja reservoir, where discrepancies were observed. This can be achieved by incorporating additional field data and continuously improving the model parameters. In addition, exploring the application of AI and machine learning algorithms to address calibration processes associated with hydro-environmental modeling holds great promise for advancing the field.

In summary, this work presents a novel combination of machine learning methods, AI algorithms, and the Monte Carlo approach to improve sediment transport modeling. Through efficient use of DNNs to reproduce model output and optimize calibration procedures, the methodology provides innovative understandings of sediment dynamics. Even though there are slight variations in certain regions, the results provide useful information for practical uses. This work contributes to theoretical understanding and has the potential to simplify sediment transport modeling in real-world circumstances due to its increased computing efficiency and accuracy. The study's repercussions are wide-ranging, influencing knowledgeable techniques for managing water resources and streamlining decision-making procedures in difficult situations concerning sediment transport processes.

Software availability

- Name of tool: R-IBER
- Developers: David López-Gómez, Marcos Sanz-Ramos, Ernest Bladé
- First year of availability: 2021
- Hardware required: basic computer with a graphical power unit (GPU) based on CUDA architecture (mainly, any NVIDIA GPU)
- Requirements: Windows OS ×64
- Source code availability: the numerical tool is freely distributed through www.iberaula.com

- Data availability: the authors do not have permission to share the data
- Cost: free
- Program languages: CUDA Fortran

CRediT authorship contribution statement

Danial Dehghan-Souraki: Formal analysis, Investigation, Methodology, Writing – original draft. **David López-Gómez:** Software, Supervision. **Ernest Bladé-Castellet:** Conceptualization, Resources, Supervision. **Antonia Larese:** Supervision, Writing – review & editing. **Marcos Sanz-Ramos:** Resources, Software.

Declaration of competing interest

Antonia Larese reports financial support was provided by Technical University of Munich. Danial Dehghan-Souraki reports financial support was provided by International Center for Numerical Methods in Engineering.

Data availability

The authors do not have permission to share data.

Acknowledgments

The contract of the Danial Dehghan-Souraki is funded by the International Center for Numerical Methods in Engineering (CIMNE-VAC-2021-1).

The authors are grateful for the support offered by the Institute of Advanced Studies of the Technical University of Munich through the Hans Fischer Fellowship program as well as Gruppo Nazionale per il Calcolo Scientifico (GNCS) by Indam. This study received funding from the European Union - Next Generation EU National Recovery and Resilience Plan [NRRP], Mission 4, Component 2, Investment 1.3–D.D. 1243 2/082022, PE0000005 Extended Partnership "RETURN: Multi-Risk sciEnce for resilienT commUnities underR a changiNg climate" and Mission 4, Component 2, Investment 1.5 - Call for tender No. 3277 of 30 dicembre 2021 ECS00000043, No. 1058 of June 23, 2023, CUP C43C22000340006, "iNEST: Interconnected Nord-Est Innovation Ecosystem".

References

AlDahoul, N., Essam, Y., Kumar, P., Ahmed, A.N., Sherif, M., Sefelnasr, A., Elshafie, A., 2021. Suspended sediment load prediction using long short-term memory neural network. Sci. Rep. 11, 7826. https://doi.org/10.1038/s41598-021-87415-4.

Alp, M., Cigizoglu, H., 2007. Suspended sediment load simulation by two artificial neural network methods using hydrometeorological data. Environ. Model. Software 22, 2–13. https://doi.org/10.1016/j.envsoft.2005.09.009.

Ariathurai, R., Arulanandan, K., 1978. Erosion rates of cohesive soils. J. Hydraul. Div. 104, 279–283.

Avendaño, C., Sanz-Montero, M.E., Cobo, R., 2000. State of the art of reservoir sedimentation management in Spain. In: Proceedings of the International Workshop and Symposium on Reservoir Sedimentation Management. Toyama, Japan, pp. 27–34.

Baljyan, P., Sedrakyan, S., Manukyan, A., 2014. Determination of sediment accumulation volume and its deposition in the Khachen reservoir. Adv. Mater. Res. 1020, 756–759. https://doi.org/10.4028/www.scientific.net/AMR.1020.756.

Bell, N., Hoberock, J., 2011. Thrust: A Productivity-Oriented Library for CUDA 26.

Bertrand-Krajewski, J.-L., Bardin, J.-P., Gibello, C., 2006. Long term monitoring of sewer sediment accumulation and flushing experiments in a man-entry sewer. Water Sci. Technol. 54, 109–117. https://doi.org/10.2166/wst.2006.619.

Van Binh, D., Kantoush, S.A., Ata, R., Tassi, P., Nguyen, T.V., Lepesqueur, J., Abderrezzak, K.E.K., Bourban, S.E., Nguyen, Q.H., Phuong, D.N.L., Trung, L.V., Tran, D.A., Letrung, T., Sumi, T., 2022. Hydrodynamics, sediment transport, and morphodynamics in the Vietnamese Mekong Delta: field study and numerical modelling. Geomorphology 413, 108368. https://doi.org/10.1016/j.geomorph.2022.108368.

Bladé Castellet, E., Cea, L., Corestein, G., 2014. Numerical modelling of river inundations. Ingeniería del agua 18, 68. https://doi.org/10.4995/ia.2014.3144.

Bladé Castellet, E., Sánchez-Juny, M., Arbat Bofill, M., Dolz Ripollés, J., 2019. Computational modeling of fine sediment relocation within a dam reservoir by means of artificial flood generation in a reservoir Cascade. Water Resour. Res. 55, 3156–3170. https://doi.org/10.1029/2018WR024434.

Bladé, E., Cea, L., Corestein, G., Escolano, E., Puertas, J., Vázquez-Cendón, E., Dolz, J., Coll, A., 2014. Iber: herramienta de simulación numérica del flujo en ríos. Rev. Int. Métodos Numéricos Cálculo Diseño Ing. 30, 1–10. https://doi.org/10.1016/j.rimni.2012.07.004.

Brodtkorb, A.R., Hagen, T.R., Schulz, C., Hasle, G., 2013. GPU computing in discrete optimization. Part I: introduction to the GPU. EURO J. Transport. Logist. 2, 129–157. https://doi.org/10.1007/s13676-013-0025-1.

Castro, M.J., Ortega, S., de la Asunción, M., Mantas, J.M., Gallardo, J.M., 2011. GPU computing for shallow water flow simulation based on finite volume schemes. Compt. Rendus Mec. 339, 165–184. https://doi.org/10.1016/j.crme.2010.12.004.

Cea, L., Bladé, E., 2015. A simple and efficient unstructured finite volume scheme for solving the shallow water equations in overland flow applications. Water Resour. Res. 51, 5464–5486. https://doi.org/10.1002/2014WR016547.

Cea, L., Bermúdez, M., Puertas, J., Bladé, E., Corestein, G., Escolano, E., Conde, A., Bockelmann-Evans, B., Ahmadian, R., 2016. IberWQ: new simulation tool for 2D water quality modelling in rivers and shallow estuaries. J. Hydroinf. 18, 816–830. https://doi.org/10.2166/hydro.2016.235.

Cea, L., Bladé, E., García-Feal, O., Sanz-Ramos, M., Gómez-Gesteira, M., Farfán, F., 2022. Extension of 2D shallow water models to hydrological modelling. In: International Association for Hydro-Environment Engineering and Research (IAHR). https://doi.org/10.3850/IAHR-39WC2521-71192022SS1085.

Dolz, J., Armengol, J., Roura, M., De Pourcq, K., Arbat, M., López, P., Arbat-Bofill, M., López, P., 2009. Estudio de la dinámica sedimentaria y batimetría de precisión del embalse de Ribarroja. Barcelona.

Duan, J.G., Nanda, S.K., 2006. Two-dimensional depth-averaged model simulation of suspended sediment concentration distribution in a groyne field. J. Hydrol. (Amst.) 327, 426–437. https://doi.org/10.1016/j.jhydrol.2005.11.055.

Einstein, H.A., Krone, R.B., 1962. Experiments to determine modes of cohesive sediment transport in salt water. J. Geophys. Res. 67, 1451–1461. https://doi.org/10.1029/JZ067i004p01451.

Esri, 2021. Arc GIS Pro Documentation [WWW Document]. URL. https://pro.arcgis.com/en/pro-app/latest/help/data/tin/tin-in-arcgis-pro.htm#ESRI_SECTION1_AC64261379754A2A92CCCF97D823C170 (accessed 4 December.23).

García-Feal, O., González-Cao, J., Gómez-Gesteira, M., Cea, L., Domínguez, J., Formella, A., 2018. An accelerated tool for flood modelling based on Iber. Water 10, 1459. https://doi.org/10.3390/w10101459.

García-Feal, O., Cea, L., González-Cao, J., Domínguez, J.M., Gómez-Gesteira, M., 2020. IberWQ: a GPU accelerated tool for 2D water quality modeling in rivers and estuaries. Water 12, 413. https://doi.org/10.3390/w12020413.

Goldstein, E.B., Coco, G., Plant, N.G., 2019. A review of machine learning applications to coastal sediment transport and morphodynamics. Earth Sci. Rev. 194, 97–108. https://doi.org/10.1016/j.earscirev.2019.04.022.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

Gupta, D., Hazarika, B.B., Berlin, M., Sharma, U.M., Mishra, K., 2021. Artificial intelligence for suspended sediment load prediction: a review. Environ. Earth Sci. 80 (9), 346. https://doi.org/10.1007/s12665-021-09625-3.

Hallouz, F., Meddi, M., Mahé, G., Alirahmani, S., Keddar, A., 2018. Modeling of discharge and sediment transport through the SWAT model in the basin of Harraza (Northwest of Algeria). Water Sci. 32, 79–88. https://doi.org/10.1016/j.wsj.2017.12.004.

Heddam, S., Naghibi, A., Khosravi, K., Singh, S.K., 2024. Chapter 15 - suspended sediment load prediction and tree-based algorithms. In: Melesse, A.M., Rahmati, O., Khosravi, K., Petropoulos, G.P. (Eds.), Remote Sensing of Soil and Land Surface Processes. Elsevier, pp. 257–269. https://doi.org/10.1016/B978-0-443-15341-9.00013-7.

Hinton, G.E., 2012. A Practical Guide to Training Restricted Boltzmann Machines, pp. 599–619. https://doi.org/10.1007/978-3-642-35289-8_32.

Huybrechts, N., Villaret, C., 2010. Comparison between 2D and 3D modelling of sediment transport: application to the dune evolution. In: Dittrich, A. (Ed.), River Flow 2010. Karlsruhe, Bundesanstalt für Wasserbau, Braunschweig, Germany, pp. 887–893.

Jain, S.K., 2001. Development of integrated sediment rating curves using ANNs. J. Hydraul. Eng. 127, 30–37. https://doi.org/10.1061/(ASCE)0733-9429(2001)127:1(30).

Jimeno-Sáez, P., Martínez-España, R., Casalí, J., Pérez-Sánchez, J., Senent-Aparicio, J., 2022. A comparison of performance of SWAT and machine learning models for predicting sediment load in a forested Basin, Northern Spain. Catena 212, 105953. https://doi.org/10.1016/j.catena.2021.105953.

Jothiprakash, V., Garg, V., 2009. Reservoir sedimentation estimation using artificial neural network. J. Hydrol. Eng. 14, 1035–1040. https://doi.org/10.1061/(ASCE)HE.1943-5584.0000075.

Kantoush, S., Sumi, T., 2010. River Morphology and Sediment Management Strategies for Sustainable Reservoir in Japan and European Alps 53.

Kerem Cigizoglu, H., Kisi, Ö., 2006. Methods to improve the neural network performance in suspended sediment estimation. J. Hydrol. 317, 221–238. https://doi.org/10.1016/j.jhydrol.2005.05.019.

Khosronejad, A., Limaye, A.B., Zhang, Z., Kang, S., Yang, X., Sotiropoulos, F., 2023. On the morphodynamics of a wide class of large-scale meandering rivers: insights gained by coupling LES with sediment-dynamics. J. Adv. Model. Earth Syst. 15 https://doi.org/10.1029/2022MS003257.

Latif, S.D., Chong, K.L., Ahmed, A.N., Huang, Y.F., Sherif, M., El-Shafie, A., 2023. Sediment load prediction in Johor river: deep learning versus machine learning models. Appl. Water Sci. 13 (3), 79. https://doi.org/10.1007/s13201-023-01874-w.

Lobera, G., Batalla, R.J., Vericat, D., López-Tarazón, J.A., Tena, A., 2016. Sediment transport in two mediterranean regulated rivers. Sci. Total Environ. 540, 101–113. https://doi.org/10.1016/j.scitotenv.2015.08.018.

Lu, C.-M., Chiang, L.-C., 2019. Assessment of Sediment Transport Functions with the Modified SWAT-Twn Model for a Taiwanese Small Mountainous Watershed. Water 11, 1749. https://doi.org/10.3390/w11091749.

Minella, J.P.G., Walling, D.E., Merten, G.H., 2014. Establishing a sediment budget for a small agricultural catchment in southern Brazil, to support the development of effective sediment management strategies. J. Hydrol. (Amst.) 519, 2189–2201. https://doi.org/10.1016/j.jhydrol.2014.10.013.

Nagy, H.M., Watanabe, K., Hirano, M., 2002. Prediction of sediment load concentration in rivers using artificial neural network model. J. Hydraul. Eng. 128, 588–595. https://doi.org/10.1061/(ASCE)0733-9429(2002)128:6(588).

NVIDIA Corporation, 2023. NVIDIA CUDA fortran programming guide [WWW Document]. Available online: https://docs.nvidia.com/hpc-sdk/compilers/cuda-fortran-prog-guide/.

Partheniades, E., 1965. Erosion and deposition of cohesive soils. J. Hydraul. Div. 91, 105–139.

Phien, H.N., Arbhabhirama, A., 1979. A statistical analysis of the sediment volume accumulated in reservoirs. J. Hydrol. (Amst.) 44, 231–240. https://doi.org/10.1016/0022-1694(79)90133-1.

Pomerat, J., Segev, A., Datta, R., 2019. On neural network activation functions and optimizers in relation to polynomial regression. In: 2019 IEEE International Conference on Big Data (Big Data). IEEE, pp. 6183–6185. https://doi.org/10.1109/BigData47090.2019.9005674.

Rai, R.K., Mathur, B.S., 2007. Event-based soil erosion modeling of small watersheds. J. Hydrol. Eng. 12, 559–572. https://doi.org/10.1061/(ASCE)1084-0699(2007)12:6(559).

Roe, P.L., 1986. Discrete models for the numerical analysis of time-dependent multidimensional gas dynamics. J. Comput. Phys. 63, 458–476. https://doi.org/10.1016/0021-9991(86)90204-4.

Roura Carol, M., Armengol, J., Fernando, J., Dolz Ripollès, J., 2008. Incidencia en los embalses de Mequinenza y Ribarroja en el transporte sólido en suspensión del río Ebro. Ingen. agua 15, 221. https://doi.org/10.4995/ia.2008.2936.

Ruiz-Villanueva, V., Bladé, E., Sánchez-Juny, M., Marti-Cardona, B., Díez-Herrero, A., Bodoque, J.M., 2014. Two-dimensional numerical modeling of wood transport. J. Hydroinf. 16, 1077–1096. https://doi.org/10.2166/hydro.2014.026.

El Safadi, E.A., Adrot, O., Flaus, J.-M., 2015. Advanced Monte Carlo Method for model uncertainty propagation in risk assessment. IFAC-PapersOnLine 48, 529–534. https://doi.org/10.1016/j.ifacol.2015.06.135.

Sanz-Ramos, M., Bladé Castellet, E., Palau Ibars, A., Vericat Querol, D., Ramos-Fuertes, A., 2019. IberHABITAT: evaluación de la Idoneidad del Hábitat Físico y del Hábitat Potential Útil para peces. Apl. Eume. Ribagua 6, 158–167. https://doi.org/10.1080/23863781.2019.1664273.

Sanz-Ramos, M., Bladé, E., González-Escalona, F., Olivares, G., Aragón-Hernández, J.L., 2021. Interpreting the Manning roughness coefficient in overland flow simulations with coupled hydrological-hydraulic distributed models. Water 13, 3433. https://doi.org/10.3390/w13233433.

Sanz-Ramos, M., López-Gómez, D., Bladé, E., Dehghan-Souraki, D., 2023. A CUDA Fortran GPU-parallelised hydrodynamic tool for high-resolution and long-term eco-hydraulic modelling. Environ. Model. Software 161, 105628. https://doi.org/10.1016/j.envsoft.2023.105628.

Sarker, I.H., 2021. Machine learning: algorithms, real-world applications and research directions. SN Comput. Sci. 2, 160. https://doi.org/10.1007/s42979-021-00592-x.

Schmidt-Hieber, J., 2020. Nonparametric regression using deep neural networks with ReLU activation function. Ann. Stat. 48 https://doi.org/10.1214/19-AOS1875.

Sedláček, J., Bábek, O., Grygar, T.M., Lenďáková, Z., Pacina, J., Štojdl, J., Hošek, M., Elznicová, J., 2022. A closer look at sedimentation processes in two dam reservoirs. J. Hydrol. (Amst.) 605, 127397. https://doi.org/10.1016/j.jhydrol.2021.127397.

Tonkin, M., Doherty, J., 2009. Calibration-constrained Monte Carlo analysis of highly parameterized models using subspace techniques. Water Resour. Res. 45 https://doi.org/10.1029/2007WR006678.

Van Rijn, L.C., 1987. Mathematical Modelling of Morphological Processes in the Case of Suspended Sediment Transport. Delft Tech. University, Delft, The Netherlands.

van Rijn, L.C., 1993. Principles of Sediment Transport in Rivers, Estuaries and Coastal Seas.

Wood, P.J., 1997. Biological effects of fine sediment in the lotic environment. Environ. Manag. 21, 203–217. https://doi.org/10.1007/s002679900019.

Zhang, L., Xue, B., Yan, Y., Wang, G., Sun, W., Li, Z., Yu, J., Xie, G., Shi, H., 2019. Model uncertainty analysis methods for semi-arid watersheds with different characteristics: a comparative swat case study. Water 11, 1177. https://doi.org/10.3390/w11061177.

de la Asunción, M., Mantas, J.M., Castro, M.J. (2010). Programming CUDA-Based GPUs to Simulate Two-Layer Shallow Water Flows. In: D'Ambra, P., Guarracino, M., Talia, D. (eds) Euro-Par 2010 - Parallel Processing. Euro-Par 2010. Lecture Notes in Computer Science, vol 6272. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-15291-7_32.

de la Asunción M, Castro MJ, Fernández-Nieto ED, Mantas JM, Acosta SO, González-Vida JM. Efficient GPU implementation of a two waves TVD-WAF method for the two-dimensional one layer shallow water system on structured meshes. Computers & Fluids. 2013 Jul 10;80:441-52.