Environmental Modelling and Software 198 (2026) 106851

Contents lists available at ScienceDirect

Environmental Modelling and Software

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

hydroMOPSO: A flexible and model-independent multi-objective optimisation R package for environmental and hydrological models ☆

Rodrigo Marinao [a], Mauricio Zambrano-Bigiarini [b,c,*], Oscar M. Baez-Villanueva [d]

[a] Master of Engineering Science, Universidad de La Frontera, Temuco, Chile
[b] Department of Civil Engineering, Universidad de La Frontera, Temuco, Chile
[c] Center for Climate and Resilience Research (CR2), Universidad de Chile, Santiago, Chile
[d] Hydro-Climate Extremes Lab, Ghent University, Ghent, Belgium

ARTICLE INFO

Dataset link: https://doi.org/10.5281/zenodo.15743400, https://doi.org/10.5281/zenodo.17273537, https://doi.org/10.5281/zenodo.17273949

Keywords:
Multi-objective
Optimisation
Pareto front
Calibration
R package
SWAT+

ABSTRACT

This article introduces hydroMOPSO, a multi-objective, model-independent R package for the calibration of hydrological and environmental models. It supports both R-based and R-external models through wrapper functions, providing flexibility for a wide range of optimisation problems. The package includes fine-tuning options to generate a Pareto-optimal front. The performance of hydroMOPSO was compared to the caRamel R package using benchmark functions and case studies involving two R-based hydrological models in an Andean catchment. hydroMOPSO outperformed caRamel on benchmarks, with faster convergence in the two hydrological models. An R-external case study demonstrated the flexibility and ease of use of hydroMOPSO, through its application to the calibration of the SWAT+ model. The package also enables the generation of informative outputs for modellers, with particular emphasis on hydrographs and parameter sets from the Pareto-optimal front. hydroMOPSO constitutes a valuable tool for researchers and practitioners seeking to implement multi-objective optimisation in environmental and hydrological modelling.

1. Introduction

The availability of optimisation software able to be used across a wide range of hydrological and environmental models is crucial for improving the representation of physical processes through robust parameter calibration (Savenije, 2009; Kavetski and Fenicia, 2011; Wallner et al., 2012; Demirel et al., 2018), which is particularly important for applications such as assessing the impacts of climate change (Royer-Gaspard et al., 2021). Calibrated model parameters are considered robust or transferable if they maintain good performance across different time periods and climatic conditions beyond those used during model calibration (Perrin et al., 2007; Bárdossy and Singh, 2008; Coron et al., 2012, 2014; Minville et al., 2014; Bisselink et al., 2016; Royer-Gaspard et al., 2021). Robust model parameters are essential to provide, at a minimum, a reasonable level of credibility that matches the complexity of the modelling task (Klemeš, 1986a,b). On the other hand, the lack of robustness in calibrated model parameters can substantially undermine the reliability of the insights generated by these models. This has critical implications for decision-making in areas such as (*i*) ecosystem services and land-use planning (e.g., McColl and Aggett, 2007; Guswa et al., 2014; Sun et al., 2020); (*ii*) flood forecasting (e.g., Kauffeldt et al., 2016; Dasgupta et al., 2023); and (*iii*) water resources management (e.g., Jayakrishnan et al., 2005; Schuol et al., 2008; Hartmann et al., 2014; Paul et al., 2021; Barría et al., 2021). However, despite advancements in optimisation software, estimating robust model parameters remains a complex challenge (Bárdossy and Singh, 2008; Qi et al., 2019), partly due to the highly non-linear nature of environmental processes (Bárdossy and Singh, 2008; Herrera et al., 2022).

In the search for robust model parameters, the computational procedure adopted for model calibration is crucial. This process involves defining the lower and upper bounds for each parameter to be calibrated, selecting a goodness-of-fit function to assess how well model simulations align with (multi-variable) real-world observations, and choosing an appropriate algorithm to estimate the parameter values (Refsgaard and Henriksen, 2004; Efstratiadis and Koutsoyiannis, 2010). Traditionally, the majority of model-calibration studies have relied on single-objective optimisation (SOO) algorithms, which assume that a unique set of parameter values provides the "*best*" model performance by comparing simulations and observations of a single variable (Efstratiadis and Koutsoyiannis, 2010). Commonly used SOO

☆ This article is part of a Special issue entitled: 'SWAT-ML-RS' published in Environmental Modelling and Software.

* Correspondence to: Av. Francisco Salazar 01145, Temuco, Chile.
E-mail address: mauricio.zambrano@ufrontera.cl (M. Zambrano-Bigiarini).

https://doi.org/10.1016/j.envsoft.2025.106851
Received 26 June 2025; Received in revised form 19 December 2025; Accepted 23 December 2025
Available online 2 January 2026

algorithms include the Shuffled Complex Evolution (SCE-UA; Duan et al., 1992), Differential Evolution (DE; Storn and Price, 1997), Dynamically Dimensioned Search (DDS; Tolson and Shoemaker, 2007), Simulated Annealing (SA; Kirkpatrick et al., 1983) and the Particle Swarm Optimisation (PSO; Kennedy and Eberhart, 1995). However, SOO approaches have been criticised for their limitations in adequately capturing all important characteristics of the observations (Yapo et al., 1998; Vrugt et al., 2003; Wagener, 2003). This often results in a trade-off scenario where improving the fit of one model output variable may increase the error in another (Mostafaie et al., 2018). Additionally, this contributes to the equifinality problem, where different parameter sets lead to similar objective function values (Beven and Binley, 1992; Beven, 1993, 2006). The early works of Gupta et al. (1998) and Yapo et al. (1998) in the late 1990s highlighted the multi-objective nature of model calibration, advocating for the inclusion of additional constraints and consideration of trade-offs between objective functions, which was later recognised as an important measure to mitigate equifinality and uncertainty (Wagener and Gupta, 2005; Tang et al., 2006; Her and Seong, 2018). Consequently, there has been a growing call to adopt multi-objective optimisation approaches that more effectively address model errors and provide a more robust quantification of both parametric and predictive uncertainties in environmental modelling (Yapo et al., 1996; Efstratiadis and Koutsoyiannis, 2010).

The development of multi-objective evolutionary algorithms (MOEAs) has been an active research area for at least three decades, gaining popularity across various engineering disciplines (Coello and Lamont, 2004; Coello, 2005; Deb, 2015). Notable examples of widely applied MOEAs include the Vector Evaluated Genetic Algorithm (VEGA; Schaffer, 1985), Multi-Objective Genetic Algorithm (MOGA; Fonseca and Fleming, 1993), Non-dominated Sorting Genetic Algorithm (NSGA; Srinivas and Deb, 1994), Strength Pareto Evolutionary Algorithm (SPEA; Zitzler and Thiele, 1999), SPEA2 (Zitzler et al., 2001), NSGA-II (Deb et al., 2002), Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D; Zhang and Li, 2007), and NSGA-III (Deb and Jain, 2014). Additionally, other algorithm families, particularly the so-called Multi-objective Particle Swarm Optimisers (MOPSOs; Moore et al., 2000; Coello and Lechuga, 2002), which are based on PSO (Kennedy and Eberhart, 1995), have attracted increasing attention from the scientific community. Notable examples within this family include OMOPSO (Sierra and Coello, 2005), SMPSO (Nebro et al., 2009), CMPSO (Zhan et al., 2013), pccsAMOPSO (Hu and Yen, 2015), and NMPSO (Lin et al., 2016).

The application of MOO for the automatic calibration of hydrological models began with the development of the Multi-Objective Complex Evolution (MOSCOM-UA; Yapo et al., 1998) to calibrate a conceptual rainfall-runoff model using two objectives. Since then, various methods have been introduced, including the Multi-objective Shuffled Complex Evolution Metropolis (MOSCEM; Vrugt et al., 2003), the enhanced Epsilon Dominance NSGA-II (ϵ-NSGA-II; Reed et al., 2003; Kollat and Reed, 2005), the Multi-objective Evolutionary Annealing-Simplex method (MEAS; Efstratiadis and Koutsoyiannis, 2008) or the hybrid algorithm `caRamel` (Monteil et al., 2020). Recent studies have explored using MOO algorithms for time-consistent model parameter identification through sub-period calibration (Gharari et al., 2013), enhancing model consistency by integrating goodness-of-fit measures with hydrological signatures (Shafii and Tolson, 2015), using remote sensing observations of vegetation (Madsen, 2003; Khu et al., 2008); and calibrating/optimising coupled (hydrogeological and hydrogeochemical) models (e.g., Naghdi et al., 2021; Rafiei et al., 2022).

Over the last decades, MOO has shown great advances in parameter estimation and uncertainty identification for complex models (Efstratiadis and Koutsoyiannis, 2010). However, the development of open-source and flexible multi-objective calibration tools has lagged behind, hindering the widespread adoption of MOO approaches. Currently, the implementation of MOO algorithms available in diverse programming languages, such as Python (Benítez-Hidalgo et al., 2019; Blank and Deb, 2020) and R (Binois and Picheny, 2019; Benitez and Pinto-Roa, 2022), often lack the flexibility required to be used in a wide range of models. Among the existing algorithms, the `caRamel` R package (Monteil et al., 2020) stands out for its flexibility and has been successfully tested with R-based hydrological models. However, it is not designed to work with R-external models, limiting its broader applicability in real-world applications.

This work presents `hydroMOPSO`, the first R package specifically designed for the multi-objective optimisation of R-external and R-based hydrological and environmental models, with minimal user intervention. Unlike existing tools, `hydroMOPSO` enables simultaneous optimisation of multiple performance criteria, multiple state variables (e.g., discharges, soil moisture), model outputs in several file formats (e.g., text files, raster files) and temporal resolutions (e.g., hourly, daily, annual). Users familiar with the widely used single-objective `hydroPSO` R package (Zambrano-Bigiarini and Rojas, 2013) will recognise several features that have been extended and enhanced in `hydroMOPSO`: (*i*) multi-platform compatibility (GNU/Linux, Windows, macOS); (*ii*) model independence through console-based execution; (*iii*) support for parallel processing to substantially reduce calibration times; (*iv*) flexible fine-tuning options with default settings informed by Marinao-Rivas and Zambrano-Bigiarini (2021); and (*v*) automated, high-quality graphics to support transparent multi-objective decision-making. Additionally, `hydroMOPSO` adheres to the FAIR guiding principles for scientific data management and stewardship (Wilkinson et al., 2016).

The rest of this paper is organised as follows. Section 2 presents key concepts of multi-objective optimisation, highlighting the differences between single and multi-objective optimisation approaches for environmental models. Section 3 offers an overview of the main functions within the `hydroMOPSO` R package, along with the general guidelines for calibrating a model. Section 4 outlines the methodology for benchmarking `hydroMOPSO`, evaluating its performance with R-based functions and models, as well as an extended R-external case study. Finally, Section 5 presents the results from the case studies, and Section 6 provides concluding remarks.

2. Multi-objective optimisation

In this section, we summarise the key concepts for applying multi-objective optimisation in Earth sciences, with emphasis on users transitioning from traditional single-objective approaches.

2.1. Parameter and objective space

In contrast to single-objective optimisation where the main goal is to identify a single parameter set that optimises a single objective function to maximise model performance, multi-objective optimisation seeks to optimise multiple (N) objective functions simultaneously, resulting in an N-dimensional *objective space* (Ringuest, 1992; Yapo et al., 1998; Deb, 2014). Each objective function evaluates different parameter sets in a D-dimensional parameter space, where D is the number of model parameters under optimisation. The collection of all evaluated parameter sets is comprised in the *parameter space* (e.g., Zitzler and Thiele, 1999; Monteil et al., 2020). Fig. 1 illustrates the relationship between these concepts, emphasising that each point in the parameter space corresponds to a point in the objective space. Whether or not these points are considered optimal depends on the concepts described below.

2.2. Pareto-optimal front

In MOO, multiple, often conflicting, goals are analysed simultaneously. A non-dominated parameter set, which defines a point (f) in the objective space, is that for which no other parameter set performs better across all objective functions simultaneously (Miettinen, 1998; Coello,

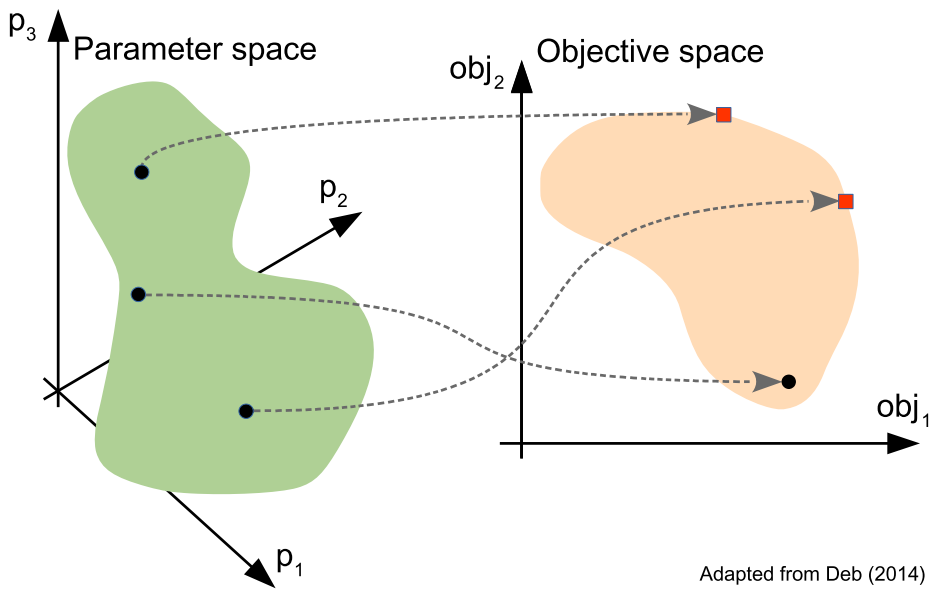

Fig. 1. Schematic representation of parameter space (left panel) and objective space (right panel) for MOO problems. In the right panel, assuming a problem with maximising objectives, non-dominated solutions that satisfy Pareto optimality are represented as red squares.
Source: This figure is adapted from Deb (2014).

1999; Ehrgott, 2005; Zitzler et al., 2004; Coello et al., 2007; Deb, 2014). In other words, to improve one objective function, one must accept a trade-off, i.e., a decrease in at least one other objective function. The collection of all non-dominated solutions (parameter sets) forms what is known as the Pareto front (Coello et al., 2007) or *Pareto-optimal front* (POF; Deb, 2014). This front represents a curve in the objective space (or a surface in higher dimensions), indicating that each point on the front is equally optimal for the given optimisation problem. Every parameter set on the Pareto front represents the best possible compromise among all objectives. In a 2-D maximisation scenario, the right panel in Fig. 1 illustrates two non-dominated solutions (red squares) and one dominated solution (black dot), with the latter excluded from the POF.

2.3. Best compromise solution

The multiple parameter sets obtained in the final POF contribute to overcome challenges such as equifinality and parametric uncertainty, often missed in single-objective methods. However, the existence of multiple non-unique solutions can overwhelm decision-makers (Luo et al., 2014; Beykal et al., 2018), particularly when dealing with a large number of conflicting objectives which require an extensive number of model runs, such as climate change impact studies, environmental management, and economic planning. As a result, it might be necessary to identify a single *best compromise solution* (BCS), which provides the best balance between all objectives in the MOO problem (Coello et al., 2007). This BCS simplifies decision-making by reducing the cognitive burden of evaluating numerous trade-offs, allowing for quicker and more efficient decisions (Velea and Lache, 2015; Shafii and De Smedt, 2009; Monteil et al., 2020; Jung et al., 2017; Efstratiadis and Koutsoyiannis, 2010). Additionally, selecting a single BCS enhances clarity and communication between stakeholders by providing a clear focal point, helping to prevent confusion and fostering consensus in collaborative settings with diverse priorities (Kogiso et al., 2014).

Fig. 2 illustrates four key points in the objective space that are used to select a best compromise solution from the parameter space. The first two points, shown in the left panel, are the ideal (Ide) and nadir (Nad) points, which define the boundaries of the Pareto Optimal Front (POF). These are essential for normalising the POF and ensuring comparability across objectives with different scales or units (Deb, 2014). The Ide point represents a hypothetical best-case scenario, where each objective independently reaches its optimal value (i.e., minimum or maximum) in the absence of trade-offs. In contrast, the Nad point corresponds to the worst values of each objective among the Pareto-optimal solutions, thereby reflecting the trade-offs inherent in multi-objective optimisation. These two points are fundamental for identifying the best compromise solution on the POF, typically defined as the solution with the minimum Euclidean distance to the Ide point in the normalised objective space.

In addition to these, two further points, depicted in the right panel of Fig. 2, extend beyond the POF and are used for normalisation and solution selection. These points are defined using user-defined acceptable ranges for each objective function. The first point is the *utopia* (Uto), which theoretically represents the intersection of the best attainable values across all objectives within the POF (Deb, 2014; Miettinen, 1998). The second is the *reference* point (Ref), which reflects the minimum acceptable performance for each objective from the decision maker's perspective (Miettinen, 1998). Together, Uto and Ref enable a more decision-oriented normalisation framework, allowing the identification of compromise solutions that align better with stakeholder preferences and practical constraints.

In this work, the best compromise solution is selected by normalising all the solutions in the POF using the Uto and Ref points. These points are relevant for real-world practical problems, such as calibration of hydrological and environmental models through goodness-of-fit functions, where they represent the best (Uto) and minimally acceptable (Ref) model performance values. As illustrated in the left panel of Fig. 2, the minimum Euclidean distance from Uto is computed within the normalised objective space defined by Uto and Ref. Each non-dominated solution in the POF is thus normalised according to Eq. (1):

$$f_i^{norm} = \frac{f_i - Ref_i}{Uto_i - Ref_i} \tag{1}$$

where i represents each objective of the optimisation problem.

2.4. Performance metrics for MOO

When evaluating the quality of the POF obtained by a MOO algorithm, three main aspects should be considered (Riquelme et al., 2015): (i) *cardinality*, which refers to the number of solutions within the POF,

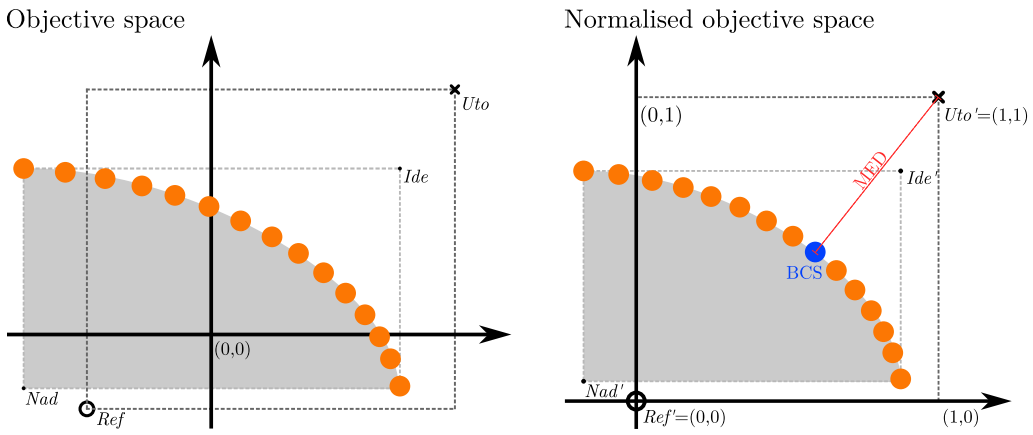

Fig. 2. Pareto-optimal front in the objective space for a maximisation problem. In the left panel black symbols indicate the utopia point (Uto), reference point (Ref), ideal point (Ide) and nadir point (Nad). The right panel illustrate the same POF in an objective space with objective values normalised using the Ref and Uto points. In both panels, the shaded area in the POF corresponds to the hypervolume of the POF.

with a larger cardinality indicating a more detailed representation of the trade-offs between objectives and, therefore, a better performance; (*ii*) *accuracy*, which measures how closely the evaluated POF approximates the theoretical true POF (or a reference POF if the true POF is unknown); and (*iii*) *diversity*, which encompasses both the distribution (relative distances between solutions) and spread (the range of values represented by the solutions).

In this work, we use two widely adopted metrics to evaluate the quality of the POF obtained through MOO optimisation. The first metric is the *hypervolume* (HV; Zitzler and Thiele, 1999), which quantifies the volume of the objective space dominated by the POF relative to a reference point, fixed to the Nad point for all case studies. The HV simultaneously assesses both accuracy and diversity of the solutions, with higher values indicating better model performance. The second metric is the *generational distance* (GD; Van Veldhuizen, 1999), which focuses solely on accuracy. The GD calculates the average Euclidean distance between each solution in the POF obtained during the optimisation and the nearest point to the true POF, which is typically known for benchmark problems. A lower GD value indicates that the solutions are closer to the true POF, reflecting better convergence towards optimal solutions (Riquelme et al., 2015). Cardinality is not explicitly considered in `hydroMOPSO` during the multi-objective optimisation. However, it is implicitly considered in each iteration by collecting all non-dominated solutions identified throughout the optimisation process to construct the most comprehensive Pareto front achievable, which is subsequently evaluated through accuracy (hypervolume) and diversity (generational distance) metrics.

For real-world applications, obtaining the true POF is impractical or impossible. In these cases, a reference POF, which includes all the known non-dominated solutions obtained during an exhaustive application of the optimisation algorithm, is used instead as an approximation. Therefore, instead of referring to it as the true or global POF, we use the term *final POF* to describe the POF obtained after the completion of all algorithm iterations (Riquelme et al., 2015). This final POF represents a local POF and may or may not be close to the true (global) POF.

2.5. The NMPSO algorithm

The novel multi-objective particle swarm optimiser (NMPSO; Lin et al., 2016) was specifically designed to address Many-Objective Optimisation Problems, which involve more than three objectives. During each iteration, NMPSO maintains an external archive that stores a limited number of particles representing the current POF. Particles are prioritised for retention in the external archive based on non-dominance and a metric introduced by Lin et al. (2016) called Balanceable Fitness Estimation (BFE). This metric guides particles toward the true POF while preserving their diversity.

Similar to the original PSO (Kennedy and Eberhart, 1995), the NMPSO algorithm starts by randomly initialising all the particle positions $\vec{X}_i = x_{i1}, x_{i2}, \ldots, x_{iD}$ in a D-dimensional parameter space. The method used for initialisation has an important impact on both the convergence speed and the quality of the optimal solutions found by PSO (Cazzaniga et al., 2015). Therefore, it is crucial to apply robust initialisation strategies, such as Latin Hypercube Sampling (LHS; McKay et al., 1979) or low-discrepancy Sobol′ sequences (Sobol, 1967) to improve performance. After initialisation, particles update their positions using problem-specific goodness-of-fit measures to identify two key points in the swarm. First, the personal best position of the ith particle, which is the best solution found by the ith particle based on its individual experience, and denoted by $\vec{P}_i = p_{i1}, p_{i2}, \ldots, p_{iD}$. Second, the global best position of the ith particle, which represents best solution found in the whole search space by that particle, and denoted by $\vec{G}_i = g_{i1}, g_{i2}, \ldots, g_{iD}$. Unlike traditional PSO, NMPSO assigns each particle a global best position randomly selected from the top 10% of the particles with the highest BFE values in the external archive (Lin et al., 2016).

After computing $\vec{P}_i$ and $\vec{G}_i$, the algorithm updates the particle's velocity in each iteration using Eq. (2).[1]

$$\vec{V}_i^{(t)} = \begin{cases} 0, & \text{if } t = 1 \\ \omega\vec{V}_i^{t-1} + c_1\vec{U}_1^{t-1} \otimes (\vec{P}_i^{t-1} - \vec{X}_i^{t-1}) + c_2\vec{U}_2^{t-1} \otimes (\vec{G}^{t-1} - \vec{X}_i^{t-1}), & \text{if } t \geq 2 \end{cases} \tag{2}$$

In this equation, $i = 1, 2, \ldots, N$ and $t = 1, 2, \ldots, T$, N is the swarm size and T is the maximum number of iterations. The parameters ω, c_1, and c_2 represent the inertia weight, the cognitive and social acceleration coefficients, respectively. The vectors $\vec{U}_1$, $\vec{U}_2$ are random vectors, each independently and uniformly distributed between 0 and 1. Note that $\otimes$ indicates element-wise vector multiplication. After updating their velocity (Eq. (2)), the position of the ith particle is updated as follows:

$$\vec{X}_i^{(t)} = \vec{X}_i^{(t-1)} + \vec{V}_i^{(t)} \tag{3}$$

To track the evolution of the POF across iterations, NMPSO uses an external archive to store the non-dominated particles identified during the optimisation process. At the initial iteration ($t = 1$), this archive is populated with N_1 non-dominated particles. Once the PSO

[1] In addition to the cognitive and social components, Lin et al. (2016) introduced a third term that blends these two components. However, in conceptual tests conducted in this research with up to three objective functions, this term was found to introduce disturbances that delay convergence.

search is complete ($t = T$), the archive holds the final set of non-dominated solutions. If the archive size exceeds its maximum allowed capacity (N_e), the BFE method (Lin et al., 2016) is applied to identify which non-dominated solutions to retain. After updating the archive, NMPSO performs a hybrid evolutionary search using all the stored non-dominated particles. This is achieved through the usage of two genetic operators: Simulated Binary Crossover (SBX) and Polynomial Mutation (PM). The first genetic operation (SBX) promotes diversity by recombining gene segments, while the second (PM) enhances local exploration (Lin et al., 2015). In each iteration, newly generated particles are evaluated, and their dominance is assessed relative to the current archive. The archive is then updated to a new size N_2. If $N_2 > N_e$, the BFE method is again used to reduce the archive size, and the resulting set of non-dominated solutions represents the current POF. At the end of each iteration, the BFE algorithm is applied once more to the updated archive (with size N_2, where $N_2 < N_e$), and the 10% of particles with the highest BFE values are selected as the global best. These elite particles are used in the subsequent iteration to update the velocity vector (Eq. (2)). For a more detailed explanation of the NMPSO algorithm, the readers are referred to Lin et al. (2016).

3. The `hydroMOPSO` R package

The `hydroMOPSO` R package is a flexible, model-independent, and multi-objective optimisation tool that implements the NMPSO algorithm (Lin et al., 2016), with a particular emphasis on environmental and hydrological applications. It can be used to calibrate any model whose input/output files can be accessed from the system console, whether implemented directly in R, referred to hereafter as an *R-based model*, or as a standalone executable run via the command line using input and output files, referred to as an *R-external model*. Coupling `hydroMOPSO` with a model requires minimal programming effort and does not require modifying the model's source code. Built upon the `hydroPSO` package (Zambrano-Bigiarini and Rojas, 2013), `hydroMOPSO` inherits several features designed to address a wide range of calibration challenges. In particular, `hydroMOPSO` supports parallel execution on multicore machines and network clusters, helping to mitigate the computational cost of calibrating complex and time-consuming models.

3.1. Main functions

The main functions of `hydroMOPSO` are illustrated in Fig. 3, summarised in Table 1, and briefly described below:

- `hydroMOPSO()`: It is the core function of the `hydroMOPSO` R package, which runs the multi-objective Particle Swarm Optimisation algorithm (NMPSO). The default configuration is tuned to achieve good results with a minimum number of iterations (Marinao-Rivas and Zambrano-Bigiarini, 2021).
- `hydromod()`: This function is designed for running R-external models within the multi-objective optimisation algorithm. It writes parameter values into the model input files, runs the external model executable file in the command line, reads the model output files, and computes the user-defined objective functions.
- `read_results()`: This function reads the `hydroMOPSO` results obtained in a previous calibration saved to disk, allowing users to analyse results and generate plots without re-calibrating the model. It returns an object identical to that produced by `hydroMOPSO()`.
- `plot_results()`: Produces high-quality, user-friendly plots to support model calibration, integrating visualisations of hydrological model outputs and parameter values, with particular emphasis on POF solutions. Internally, it calls three subordinate functions (`plot_out()`, `plot_pof()`, `plot_param()`) to generate figures using the model outputs, the objective function values in the POF, and parameter values in the POF, respectively. If no arguments are provided, it internally calls the `read_results()` function to load results from a previous calibration to then generate the plots.

Table 1
Main functions of the `hydroMOPSO` R package[a].

Function	Short description
`hydroMOPSO()`	Performs a model-independent multi-objective optimisation with NMPSO.
`hydromod()`	Runs R-external models.
`read_results()`	Reads results produced by `hydroMOPSO`.
`plot_results()`	Plots the main results of the optimisation.
`plot_pof()`	Plots the Pareto-optimal front.
`plot_out()`	Plots time series from model output variables.
`plot_param()`	Plots boxplots and dotty plots comparing parameter values with OFs.

[a] Reproducibility note: we recommend to always use `set.seed()` and saving `sessionInfo()` with outputs.

Fig. 3 illustrates the interaction between the main functions of the `hydroMOPSO` R package. This flowchart introduces the concept of *wrapper function*, a key element that users must define for each calibration problem. The wrapper function is a short user-defined R function that specifies what are the model parameters to be optimised, what are the observations and the model outputs to be compared, and what are the objective functions to be applied. Detailed information on how to define and use this function is provided in Section 3.2.3.

3.2. Calibrating a model

This section outlines how to (*i*) fine-tune the `hydroMOPSO` parameters, (*ii*) read model outputs, and (*iii*) integrate these steps into a single user-defined wrapper function to handle the entire optimisation process.

3.2.1. Modification of model parameters

When working with R-based hydrological models (i.e., when `fn=='hydromodInR'`), locating and adjusting the model parameters is typically straightforward, since they are passed as arguments in the model's execution function. For example, hydrological models like the `TUWmodel` (Viglione and Parajka, 2020) and `airGR` (Coron et al., 2022) follow this approach. In its simplest form, an R function representing a hydrological model could be structured as `some_Rbased_model(param.values, ...)`, where `param.values` is a vector containing the model's parameter values. This structure is similar to examples detailed in tutorials dedicated to the `hydroPSO` package (Zambrano-Bigiarini and Baez-Villanueva, 2020; Zambrano-Bigiarini, 2020).

However, when working with R-external models (i.e., when `fn='hydromod'`), modifying model parameters requires a more elaborate procedure. In this case, `hydroMOPSO` updates the parameter values directly in the model's input files, using the information specified in the `ParamRanges.txt` and `ParamFiles.txt` files (see Appendix A), before executing the model. This approach is compatible with a wide range of models, including the Soil and Water Assessment Tool (`SWAT`; Arnold et al., 1998), its enhanced version `SWAT+` (Arnold et al., 2024), the `Raven` hydrological modelling framework (Craig et al., 2020), and the U.S. Geological Survey's modular groundwater model (`MODFLOW`; Niswonger et al., 2011). Moreover, it can be used indirectly with models that feature a graphical user interface (GUI), such as the Water Evaluation and Planning System (`WEAP`; Yates et al., 2005). To enable parameter modification via the command line, users must provide two text files. The first, `ParamRanges.txt`, defines the parameter space by specifying the lower and upper bounds for each model parameter. The second, `ParamFiles.txt`, indicates

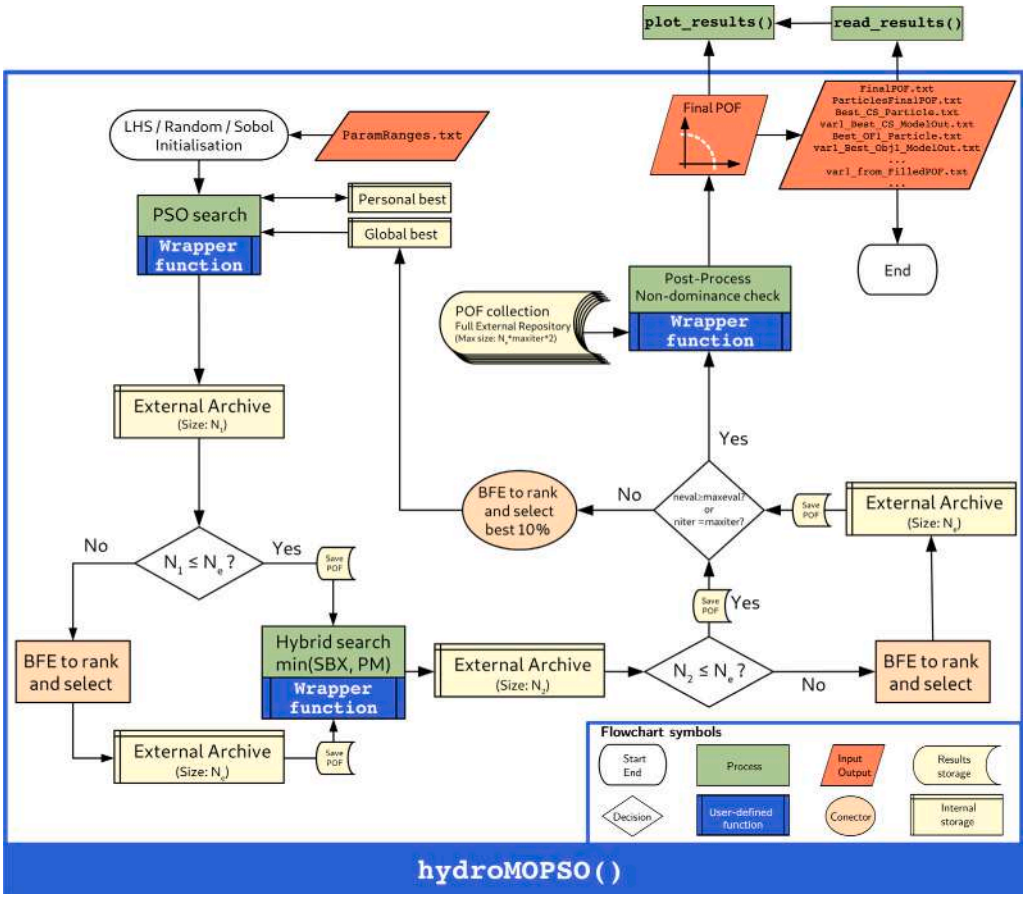

Fig. 3. Flowchart illustrating the interaction between the main `hydroMOPSO` functions, from the initial *PSO and hybrid search* to the update of the global best. The process incorporates a user-defined wrapper function and includes post-processing steps to identify all the non-dominated solutions and a single best compromise solution. The legend located in the bottom-right corner explains the flowchart symbols, which follow Unified Modelling Language (UML) conventions and ensure consistency in the colour scheme used.

the specific locations within the model's input files where the corresponding parameters can be found and modified. A detailed description of the structure and content of these files is provided in Appendix A.

3.2.2. Reading model outputs

Reading outputs after model execution is straightforward in the case of R-based models (when `fn=='hydromodInR'`), since the output variables are typically returned by the model's functions implemented in R. However, for R-external models (when `fn=='hydromod'`), this process is more complex, similar to the situation of input parameters. In these cases, users must create a custom function to read the time series of the relevant output variables. An example of such a function is provided in Appendix B, where the function `swat2zoo()` is designed specifically to read outputs from the `SWAT+` model. The arguments of this function can be customised as needed.

3.2.3. The wrapper function

The flexibility of the `hydroMOPSO` R package is primarily achieved through the wrapper function, which allows users to integrate parameter modification and output reading during model execution. This function is designed to adapt to different calibration approaches and is left to the user to define. The following guidelines can assist in developing the wrapper function:

- The wrapper function must contain the following input arguments: `param.values`, a numeric vector containing the model parameters; `Obs`, a list of observations of the output variables; `var.names`, a character vector with the names of the variables being considered; `var.units`, a character vector with the measurement units for each variable; `OFs.names`, a character vector with the names of the objective functions (OFs); `full.period`, a vector of class `Date`, representing the full calibration period; `warmup.period`, a vector of class `Date` for the warm-up period; and `cal.period`, a vector of class `Date` for the calibration period.
- When dealing with R-external models, this sub-routine must execute the `hydromod()` function. This function writes the parameters provided by `param.values` to the appropriate input files, runs the executable specified by (`exe.name`), and manages the two standard files that detail the parameter changes and ranges (by convention, `ParamFiles.txt` and `ParamRanges.txt`, respectively). Additionally, the `hydromod()` function must include the custom function to read the model output variables, such as the `swat2zoo()` function, as described in Appendix B.
- Using simulations and observations of the variable(s) of interest (for one or more model elements/points), the user can compute the performance metrics (objective functions) as required. The wrapper function must return a list with two specific objects: a numeric vector named `'OFs'` containing all the performance metrics, and a `(sub)list` named `'Sim'` that includes all the simulated output variables used in the performance calculations.

Figs. 4 and 5 illustrate the minimum code structure required for the wrapper function, tailored for R-based and R-external models, respectively. Additionally, Fig. 6 presents a flowchart depicting the

```
WrapperFunction <- function(param.values, Obs, ...){
  # Running the model
  Sim_Rbased <- some_Rbased_model(param.values,
                                  ...)
  # Getting the simulated variables
  sim.var1 <- Sim_Rbased[[1]] #simulated variable 1
  sim.var2 <- Sim_Rbased[[2]] #simulated variable 2
  sim.var3 <- Sim_Rbased[[3]] #simulated variable 3

  # Getting the observations (from user-defined input files)
  obs.var1 <- Obs[[1]] # observed variable 1
  obs.var2 <- Obs[[2]] # observed variable 2
  obs.var3 <- Obs[[3]] # observed variable 3

  # Calculating 3 goodness-of-fit measures:
  gof1 <- KGE(sim = sim.var1, obs = obs.var1) #OF for variable1
  gof2 <- NSE(sim = sim.var2, obs = obs.var2) #OF for variable2
  gof3 <- NSE(sim = sim.var3, obs = obs.var3) #OF for variable3

  # Outputs to be returned to hydroMOPSO
  out <- list(OFs  = c(gof1, gof2, gof3),
              Sims = list(sim.var1, sim.var2, sim.var3))
  return(out)
} # 'WrapperFunction' END
```

Fig. 4. Basic structure of a wrapper function for an R-based model (`fn==''hydromodInR''`).

```
WrapperFunction <- function(param.values, Obs, ...){
  # Running the model
  Sim <- hydromod(param.values,
                  exe.fname="some_Rexternal_model.exe",
                  param.files="MOPSO.in/ParamFiles.txt",
                  param.ranges="MOPSO.in/ParamRanges.txt",
                  out.FUNs=c("swat2zoo", "swat2zoo"),
                  out.FUNs.args=list(
                    list(file.name = "cha_day.txt", id=5, name.col = "fl"),
                    list(file.name = "bas_day.txt", id=1, name.col = "et"),
                    list(file.name = "bas_day.txt", id=1, name.col = "sm")
                    ), ...)
  # Getting the simulated variables
  sim.var1 <- Sim[[1]] #simulated variable 1
  sim.var2 <- Sim[[2]] #simulated variable 2
  sim.var3 <- Sim[[3]] #simulated variable 3

  # Getting the observations (from user-defined input files)
  obs.var1 <- Obs[[1]] # observed variable 1
  obs.var2 <- Obs[[2]] # observed variable 2
  obs.var3 <- Obs[[3]] # observed variable 3

  # Calculating 3 goodness-of-fit measures:
  gof1 <- KGE(sim = sim.var1, obs = obs.var1) #OF for variable1
  gof2 <- NSE(sim = sim.var2, obs = obs.var2) #OF for variable2
  gof3 <- NSE(sim = sim.var3, obs = obs.var3) #OF for variable3

  # Outputs to be returned to hydroMOPSO
  out <- list(OFs  = c(gof1, gof2, gof3),
              Sim = list(sim.var1, sim.var2, sim.var3))
  return(out)
} # 'WrapperFunction' END
```

Fig. 5. Basic structure of a wrapper function for an R-external model (`fn==''hydromod''`).

operation of the wrapper function, with panel (a) dedicated to the R-based model and panel (b) to the R-external model.

4. Case studies

In this work, we evaluate the effectiveness and efficiency of `hydroMOPSO` using two distinct families of case studies: (*i*) benchmark problems, which involve mathematical functions with a known true POF, and (*ii*) real-world model optimisation problems where the true POF is unknown. The real-world case studies are further categorised into two types of optimisation problems, representing a broad range of practical applications: (*a*) optimisation of two R-based hydrological models (`TUWmodel`, `GR4J`), and (*b*) optimisation of a R-external hydrological model (`SWAT+`).

For the benchmark functions and the real-world R-based case studies, we compared the performance of `hydroMOPSO` with `caRamel` (Monteil et al., 2020), the only multi-objective R package publicly available for optimising environmental and hydrological models. `caRamel` uses a hybrid approach that combines the `MEAS` and `NSGA-II` multi-objective optimisation algorithms, which has been shown to outperform these algorithms when applied independently (Monteil et al., 2020).

4.1. hydroMOPSO configuration

For all case studies, we used the default configuration of `hydroMOPSO` as outlined in Marinao-Rivas and Zambrano-Bigiarini (2021). Specifically, the setup included: (*i*) a swarm size of 10 particles (N = 10); (*ii*) a maximum number of particles in the external archive equal to 100 (N_e = 100); and (*iii*) a limit of genetic operations for the external archive set at 50% of N_e (Max_{GO} = 50). Particle initialisation was performed using Latin-hypercube sampling (LHS).

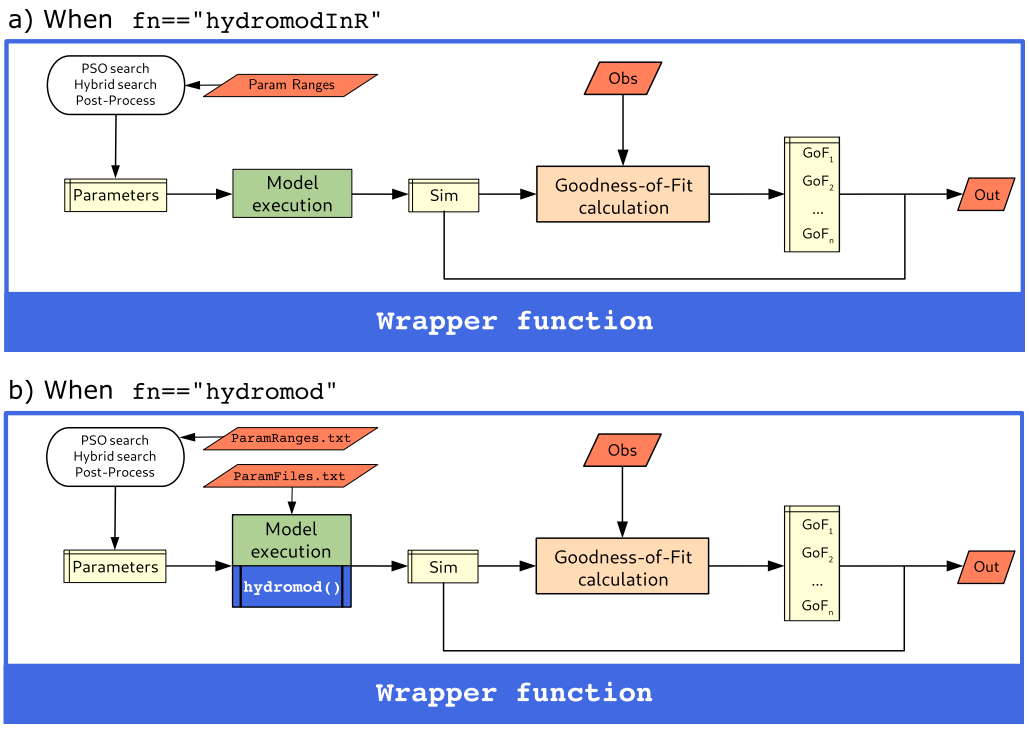

Fig. 6. Flowchart illustrating the wrapper function required to run `hydroMOPSO`. Panel (*a*) details the process for R-based models, whereas panel (*b*) demonstrates the procedure for R-external models, highlighting how the `hydromod()` function manages interactions with the specific R-external model. The meaning of each one of the flowchart symbols is the same as in Fig. 3.

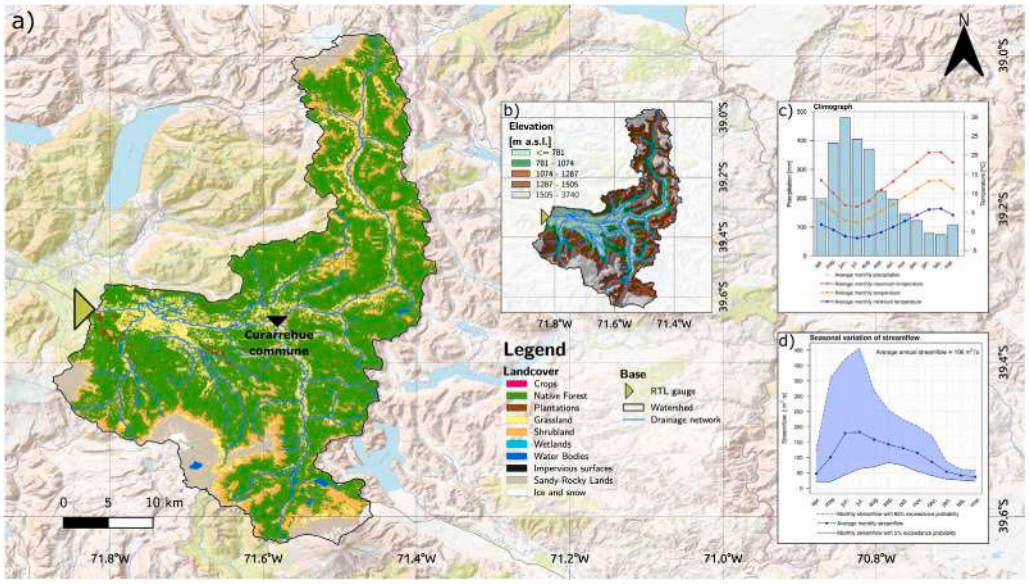

Fig. 7. (*a*) *Location of the Río Trancura antes de Llafenco* (RTL) sub-basin, (*b*) elevations [m a.s.l.], (*c*) mean monthly streamflow [$m^3\ s^{-1}$], (*d*) climograph.

4.2. Benchmark problems

The four benchmark problems used in this study (DTLZ1, DTLZ2, DTLZ3, and Kursawe) are well established in the multi-objective optimisation literature (Deb et al., 2005). These problems were minimised using the `smoof` R package (Bossek, 2017). According to Deb et al. (2005), the number of parameters (D) for each benchmark problem is defined by the relationship $D = N + k - 1$, where N represents the number of objective functions, and k is an integer specific to each problem. For DTLZ1, DTLZ2, and DTLZ3, the values of k were set to 5, 10, and 10, respectively. Therefore, with three objective functions for each benchmark problem ($N = 3$), the number of parameters to be optimised (D) was 7, 12, and 12 for DTLZ1, DTLZ2, and DTLZ3, respectively. For these DTLZ minimisation problems, the HV was computed using the search space defined by the POF and a Nad point of Ref. Deb et al. (2005), i.e., improvements of the POF should result in increases in the corresponding HV values. For the DTLZ1 function, the Nad point was set at $(0.5, 0.5, 0.5)$, while for DTLZ2 and DTLZ3, the reference point was $(1, 1, 1)$. For the Kursawe benchmark function, which has two objectives and is conventionally solved as a minimisation problem, the known Pareto front imposed a Nad point located at $(-13, 0)$ for HV calculations.

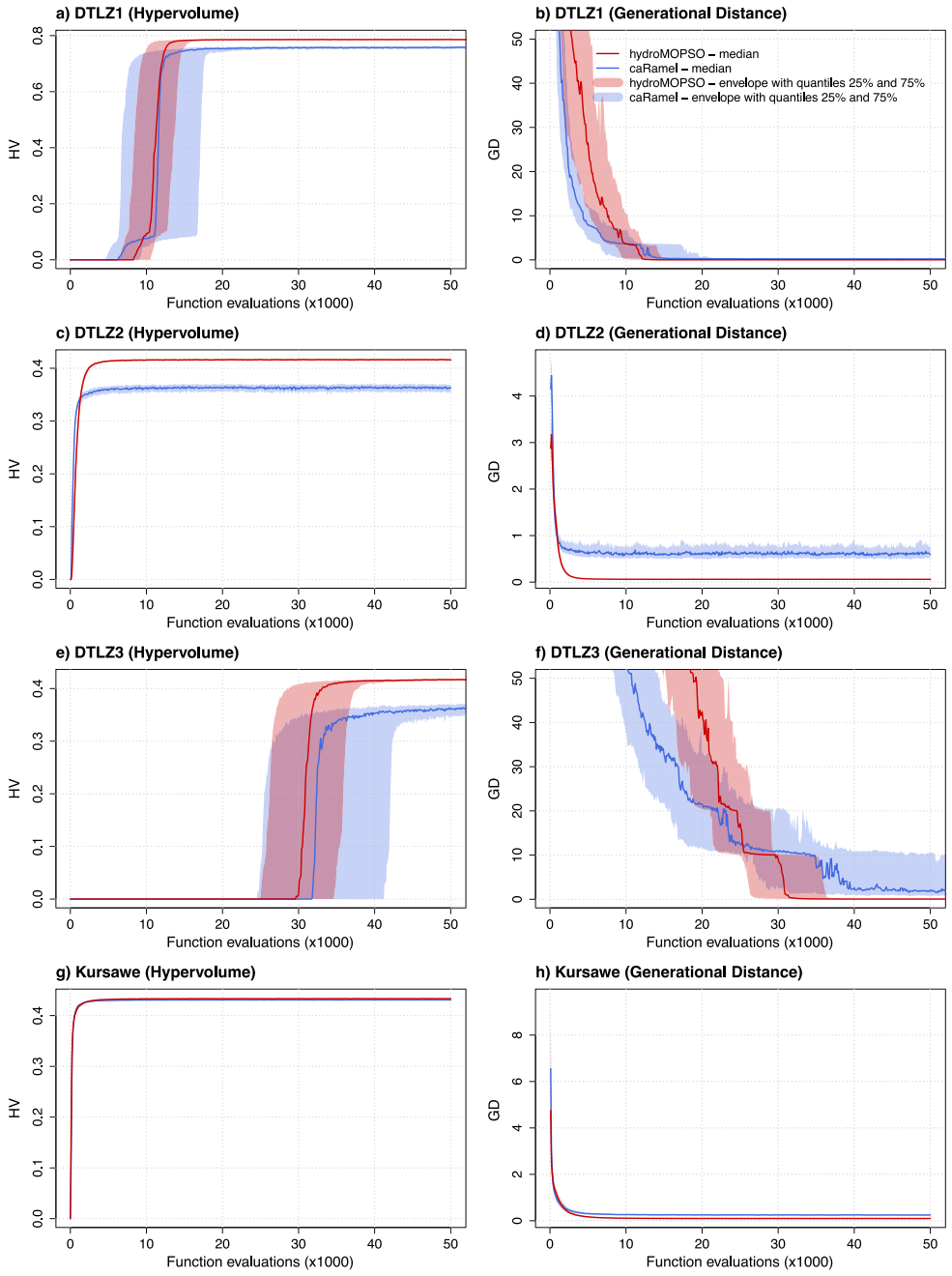

Fig. 8. Performance comparison between `hydroMOPSO` and `caRamel` in terms of HV and GD for benchmark problems. The left column of the graph composition (panels a, c, e, and g) presents the HV results, while the right column (panels b, d, f, and h) depicts the GD results.

To account for the uncertainty inherent in the stochastic nature of the algorithms used, each algorithm was applied to each problem 200 times with random initialisations, using a large number of iterations (up to 100,000 model evaluations). The goal was to visualise the median evolution of each metric (HV and GD) and capture the uncertainty through an envelope, reflecting the variability across different starting points.

4.3. Real-world case studies

As real-world case studies, we applied two R-based lumped hydrological models (`TUWmodel` and `GR4J`) and one semi-distributed, externally coupled hydrological model (`SWAT+`) to an Andean catchment in southern Chile.

4.3.1. Study area

For all the hydrological models applied in this study, we used the *Río Trancura antes de Llafenco* sub-basin (referred to as the RTL sub-basin) as the study area. This sub-basin is a tributary of the *Río Toltén* which flows through the eastern slopes of the La Araucanía region in southern Chile (see Fig. 7). It has an area of 1380 km^2 and elevations ranging from 353 to 3740 m a.s.l., defined by the homonymous streamflow station (ID 9414001). It has a pluvio–nival hydrological regime, with mean annual values of precipitation, air temperature and streamflows of 2790 mm, 7.6°C and 106 $m^3\ s^{-1}$, respectively, for the period 1980–2020. The climograph in Fig. 7c shows the occurrence of precipitation throughout the year, with maximum values from May to August (380–480 mm/month), mean air temperatures varying from 3°C (JJA) to 14°C (DJF), while the mean monthly streamflows shown in

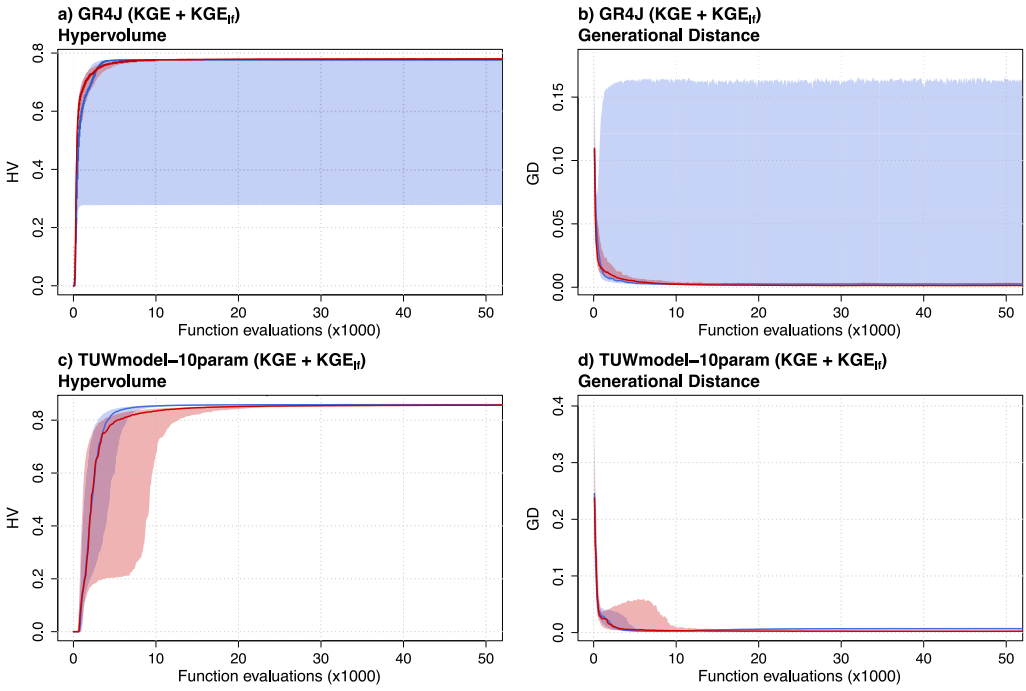

Fig. 9. Performance comparison between `hydroMOPSO` and `caRamel` in terms of HV and GD for real-world case studies developed as R-based models. Panels *a* and *c* present the HV results, while panels *b* and *d* show the GD results.

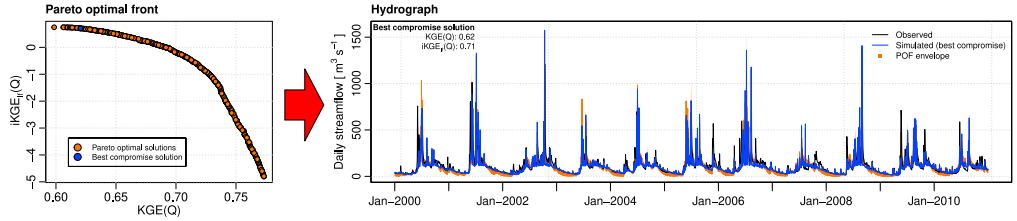

Fig. 10. Results of the R-external model calibration using `SWAT+`. The left panel displays the final Pareto front, highlighting the utopia point and the best compromise solution. The right panel presents the observed and simulated flow hydrographs, with the envelope highlighted in orange and the best compromise solution in blue.

7d reach their maximum value during winter (JJA), decreasing slowly during spring (SON) and summer (DJF).

4.3.2. Input data

For catchment delineation, the `TanDEM-X DLR 2017` and `ALOS PALSAR` (Rosenqvist et al., 2007) datasets were combined into a single digital elevation model with 12.5 m resolution. Land use information for 2018 was obtained from the `CLDynamicLandCover` product (Galleguillos et al., 2024) at 30 m resolution and was considered representative of the study period. Soil type information was collected from `CLSoilMaps` (Dinamarca et al., 2023), a 100 m gridded dataset of physical soil properties and hydraulic parameters for Chile and shared basins with Argentina. This dataset has six standardised soil depths, following the standards of the GlobalSoilMap project, and was created using digital soil mapping techniques and the Rosetta3 pedotransfer function (Zhang and Schaap, 2017).

Daily precipitation and maximum/minimum air temperature data were obtained from the CR2MET v2.5 gridded product (Boisier, 2023), and daily potential evaporation was aggregated from the hourly global `hPET` gridded product (Singer et al., 2021), which is derived from ERA5-Land. Finally, daily streamflow data were obtained from the Chilean General Water Directorate (DGA, https://dga.mop.gob.cl/servicioshidrometeorologicos/Paginas/default.aspx).

4.3.3. R-based case studies

Two R-based lumped hydrological models (`TUWmodel`, `GR4J`) were used as real-world case studies with both `hydroMOPSO` and `caRamel` R packages, to allow for a direct evaluation of the quality of the solutions produced by these two algorithms. The evaluation was performed in terms of the HV and GD metrics (see Section 2.4) obtained from these two multi-objective optimisation R packages. As with the benchmark functions presented in Section 4.2, to account for the uncertainty from the stochastic nature of the algorithms, the evolution of the performance metrics was analysed using 200 random initialisations, each with up to 100,000 iterations. This approach provided a median evolution of each metric and an envelope representing the uncertainty across different initialisations. For each problem, the final results of all runs (using both methods) were used to define the final POF, from which the HV and GD metrics were calculated.

The following paragraphs provide a short description of the two selected R-based lumped hydrological models.

`TUWmodel`. Developed by the Technical University of Vienna, this hydrological model is based on the HBV model (Bergström and Forsman, 1973), operates on daily and hourly scales, and is implemented in R via the `TUWmodel` package (Viglione and Parajka, 2020). It simulates snow accumulation, soil moisture, and surface runoff, and was validated across 320 Austrian basins (Parajka et al., 2007). In this

study, a sensitivity analysis was used to identify the most important parameters to be calibrated out of a total of 14 model parameters, as outlined in Baez-Villanueva et al. (2021) and following guidelines from Parajka et al. (2007), Ceola et al. (2015), Viglione and Parajka (2020) and Széles et al. (2020).

GR4J. The *modèle du Génie Rural à 4 paramètres Journalier* is a daily lumped rainfall-runoff model developed by Perrin et al. (2003) and implemented in R via the `airGR` package (Coron et al., 2017). For calibration we included all four parameters of the model, with ranges based on the 80% confidence intervals outlined by Perrin et al. (2003).

4.3.4. R-external case study

To further demonstrate the versatility of `hydroMOPSO`, the semi-distributed `SWAT+` hydrological model was used as an R-external real-world case study. This example showcases the practical applications of `hydroMOPSO` and its ability to handle models that cannot be executed from within the environment of a programming language, and therefore need to be run from the command line of the operating system.

SWAT+. An open-source, semi-distributed hydrological model designed to simulate both surface and groundwater quantity and quality at various scales. It includes features for analysing the impact of water management practices and predicting the effects of land use and climate change (Bieger et al., 2017). `SWAT+` (Arnold et al., 2024), available since 2015, is an updated version of the Soil Water Assessment Tool (SWAT; Arnold et al., 1998), which has been widely used for over 40 years to model hydrological processes and pollutant transport under diverse environmental conditions, management practices, and scenarios of climate and land use change (Douglas-Mankin et al., 2010; Gassman et al., 2007). In addition to meteorological forcings and observed output variables, `SWAT+` requires all input data mentioned in Section 4.3.2. Further modelling considerations for `SWAT+` are detailed in Appendix E.

4.3.5. Objective functions

For all the hydrological models considered in this work, we focused exclusively on streamflow and evaluated two objective functions. The first is the Kling–Gupta efficiency (KGE) (Gupta et al., 2009) (Eq. (4)), which is particularly sensitive to high flow conditions.

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

where r is the Pearson correlation coefficient between observed and simulated values, α is a measure of the variability error, defined as the ratio of the standard deviation of simulated to observed values, and β is a measure of bias, expressed as the ratio of the means of simulated to observed values.

As a second objective, we use the average between the Kling–Gupta efficiency applied to flows and the same metric applied to inverse flows $KGE_{lf}(Q)$ (Eq. (5)), as proposed by Garcia et al. (2017) for calibrating low flows. The challenge with this metric is that it involves inverted flows, i.e., $1/Q$, which can lead to issues with values very close to zero, or zero. To address this, we incorporated an epsilon value, equivalent to 1% of the average observed flow ($\epsilon = \overline{Q}/100$), as suggested by Pushpalatha et al. (2012). This prevents the inverted flows from approaching infinity in such cases.

$$KGE_{lf}(Q) = \frac{KGE(Q) + KGE\left(\frac{1}{Q+\epsilon}\right)}{2} \tag{5}$$

Both $KGE(Q)$ and $KGE_{lf}(Q)$ were calculated using the `hydroGOF` R package (Zambrano-Bigiarini, 2024).

4.3.6. Sensitivity analysis

For the `TUWmodel` and `SWAT+` case studies, a global sensitivity analysis (GSA) was performed to reduce the number of parameters to be calibrated. This analysis used variance-based GSA using samples from Sobol' sequences (Sobol, 2001; Saltelli et al., 2010). The single objective function used for this analysis was the average of $KGE(Q)$ and $KGE_{lf}(Q)$, as described in Section 4.3.5. Parameters were ranked by their Total Order Index (TOI) values, which measure their impact on the model, including both direct effects and interactions with other parameters. The 10 parameters with the highest TOI values were selected for calibration in both models.

In the case of `TUWmodel`, the analysis included 14 parameters with calibration ranges adapted from previous studies (Parajka et al., 2007; Ceola et al., 2015; Baez-Villanueva et al., 2021). Detailed information about these parameters, including their descriptions, calibration ranges, and the Total Order Index values obtained from the global sensitivity analysis, is provided in Table C.1.

For `SWAT+`, the sensitivity analysis was conducted on 17 parameters, selected based on the considerations outlined in Section 4.3.1. These parameters cover various hydrological processes, including: soil properties (`soil_k`, `soil_awc`), flow in the root zone (`perco`, `latq_co`), flow in the shallow aquifer (`alpha_bf`, `rchg_dp`, `spec_yld`, `flo_min`, `revap_min`, `revap`), runoff (`CN2`), snow accumulation and melting (`fall_tmp`, `melt_tmp`, `melt_max`, `melt_min`), canopy storage (`can_max`), and plant development (`rt_dp_max`). Detailed information about these parameters, including their descriptions, calibration ranges, and the Total Order Index values obtained from the global sensitivity analysis, is provided in Table D.1.

5. Results

The results are organised according to the case studies detailed in Section 4, encompassing both benchmark problems and real-world applications for R-based and R-external models.

5.1. Benchmark problems

Regarding the performance evaluation of the benchmark problems (all R-based), Fig. 8 provides a summary of the evolution of HV and GD relative to the number of model runs. The figures include solid lines that highlight the median results from all random initialisations, along with an envelope representing the range of variability.

Regarding the DTLZ problems, both methods exhibit similar behaviour in HV evolution during the evaluations. However, `hydroMOPSO` achieves higher HV values as the Pareto front stabilises. Specifically, for DTLZ1, `hydroMOPSO` reaches a value of 0.79 before 20,000 evaluations; for DTLZ2 it achieves 0.42 before 10,000 evaluations; and for DTLZ3 obtains a value of 0.42 before 45,000 evaluations. In contrast, `caRamel` only reaches values of 0.76, 0.37, and 0.37 for DTLZ1, DTLZ2, and DTLZ3, respectively, requiring significantly more evaluations for DTLZ1 and DTLZ3. For the Kursawe function, there is no substantial difference between the two algorithms. Both methods stabilise at an HV value of 0.43, with nearly the same number of evaluations.

In terms of GD, `hydroMOPSO` generally performs better, as indicated by the median evolution, showing a superior ability to achieve high accuracy of the Pareto front. For the DTLZ2 and Kursawe functions, `hydroMOPSO` reaches GD values close to zero in less than 5000 model runs. For DTLZ1 and DTLZ3, it achieves similarly low GD values in less than 12,000 and 32,000 model runs, respectively.

5.2. Real-world case studies

Regarding the performance evaluation of the real-world case studies developed as R-based models, Fig. 9 provides a summary of the evolution of HV and GD as a function of the number of model runs. The solid line in each graph highlights the median values derived from all random initialisations, accompanied by an envelope that represents the range of variability.

Regarding the calibration of the GR4J hydrological model, the results indicate that hydroMOPSO shows a slight advantage in convergence to the Pareto-optimal front, as evidenced by the median hypervolume, which reflects both accuracy and diversity. The envelope for hydroMOPSO is narrower compared to that of caRamel, suggesting that hydroMOPSO often avoids sub-optimal solutions even after a large number of model runs (up to 20,000 in Fig. 9). In contrast, for the calibration of the TUWmodel, the median HV evolution is similar for both methods, but caRamel shows a small advantage. Additionally, caRamel also produces a smaller envelope compared to hydroMOPSO, indicating that it delivered more consistent solutions.

In terms of GD, which measures accuracy, the calibration results for the GR4J model show that both methods have similar median GD values up to 5000 model runs, with a slight advantage for caRamel. However, after 15,000 runs, hydroMOPSO surpasses caRamel in terms of accuracy. The envelope of caRamel is relatively wide, suggesting that it may often converge to sub-optimal Pareto fronts.

For TUWmodel, caRamel initially has an advantage with a narrower envelope, indicating more consistent solutions. In contrast, hydroMOPSO requires more evaluations to stabilise the Pareto front but maintains stable accuracy with additional model runs. Notably, while hydroMOPSO maintains its accuracy, caRamel experiences a degradation after 13,000 runs.

5.2.1. R-external case study

Pareto-optimal front and hydrographs. Fig. 10 illustrates the outcomes of the multi-objective calibration of SWAT+. The left panel displays the final Pareto front obtained with hydroMOPSO in the objective space, highlighting the best compromise solution. The final Pareto front reveals the typical concave shape, reflecting the trade-off between the two objectives, but there is a significant difference in the ranges of the two objectives. Specifically, $KGE(Q)$ values are concentrated within a narrow range ($0.60, 0.77$), while $KGE_{lf}(Q)$ presents a wider one ($-4.80, 0.75$). This disparity introduces a distortion in identifying the best compromise solution, making it necessary to adapt the front to the normalised objective space, following the approach explained in Section 2.3. After normalisation, the best compromise solution is $BCS = (0.62, 0.71)$. The right panel of the figure shows the hydrographs of all solutions of the Pareto front, represented as an envelope with a p-factor of 0.71 and an r-factor of 0.6, compared to the observed values.

In Fig. 11, the streamflow output variable (as shown in Fig. 10) is presented as flow duration curves (FDCs) to evaluate the model's performance across different flow magnitudes. These curves highlight the flow characteristics provided by the solutions that maximise $KGE(Q)$, maximise $KGE_{lf}(Q)$, and the one that represents the best compromise solution. In the left panel of Fig. 11, the solution that maximises $KGE(Q)$ proves to be less effective for low flows (above 50% probability of exceedance), substantially underestimating these values. Conversely, the solution maximising $KGE_{lf}(Q)$ accurately adjusts to low flows, while the best compromise solution performs well in this range. The envelope encompasses most low flows, except for those with a probability of exceedance above 90%.

In the right panel of Fig. 11 the flow duration curves illustrate the mean-high flows. The solution that maximises $KGE_{lf}(Q)$ exhibits a poorer representation of high flows compared to the solution that maximises $KGE(Q)$. Specifically, the $KGE(Q)$-maximising solution aligns well with the observed values for exceedance probabilities between 10 and 40%.

Parameters. Regarding the parameters of the optimal solutions, we highlight general observations on their behaviour based on the dispersion shown in the dotty plots of Figs. 12 and 13, for KGE and $KGE_{lf}(Q)$, respectively. For the parameter alpha_bf, the model achieves satisfactory simulations for both objectives when this parameter is set to its maximum value of 1. This indicates that the basin's baseflow responds quickly to changes in recharge.

The perco parameter substantially influences the performance of the metrics across its entire range. The best $KGE(Q)$ performances are achieved when perco is close to 1, indicating that percolation occurs at lower moisture levels, relatively closer to field capacity. In contrast, the best $KGE_{lf}(Q)$ results are obtained with perco values around 0.75, suggesting that percolation is less frequent and requires higher moisture levels.

For the parameter latq_co, optimal solutions are obtained when its value is 0, which indicates that the sub-basin has a negligible contribution of lateral flow, both for low and medium-high flows. Similarly, the parameter rchg_dp also yields optimal solutions when it takes a value 0, suggesting that the catchment has a minimal contribution from recharge to the deep aquifer.

The soil_awc parameter also substantially influences the model's response for both medium-high and low flows but optimal performance is achieved within a more constrained range than the originally specified (1.6–2.4). Higher values of available water capacity, which indicate greater water retention in the soil layers, enhance the simulation of medium-high flows but reduce the accuracy of low flow representation. Conversely, lower available water capacity values improve the simulation of low streamflows but reduce the accuracy of medium and high flows.

The parameters CN, soil_k, flo_min, revap_min, and revap exhibit different behaviour from those discussed previously. Specifically, these parameters show significant dispersion within a narrower range than those used during calibration. This means that for the same parameter value, the performance can vary considerably across different objective functions. Similar performance levels can be achieved with different parameter values. This does not necessarily indicate that these parameters are insensitive within the narrower range; rather, it suggests that within this range of optimal solutions, the model's response is primarily influenced by other parameters, such as perco and soil_awc.

6. Conclusion

In this manuscript, we introduced hydroMOPSO, a new model-independent and multi-platform R package designed for the multi-objective optimisation of environmental and hydrological models. Our package implements the robust NMOPSO MOO algorithm, and its flexibility allows its application to both R-based and R-external types of models with high-quality results. hydroMOPSO offers fine-tuning options and features a default configuration optimised to efficiently achieve a satisfactory Pareto-optimal front with a reduced number of evaluations.

To assess the performance of hydroMOPSO, we compared it with the caRamel R package, which has demonstrated effectiveness in solving multi-objective optimisation problems. Our comparisons demonstrated that hydroMOPSO outperforms caRamel in solving four benchmark problems, as evidenced by the HV and GD metrics. In addition, in the calibration of the R-based TUWmodel hydroMOPSO exhibited greater dispersion but similar median HV values compared to caRamel. Conversely, for the calibration of the R-based GR4J model, hydroMOPSO achieved faster convergence to a higher HV with less dispersion than caRamel.

To further illustrate the flexibility of hydroMOPSO, the semi-distributed SWAT+ hydrological model was used as an R-external real-world case study with two objectives (KGE and $KGE_{lf}(Q)$). Results obtained in all case studies highlight the ease of use of hydroMOPSO, including the automatic generation of highly informative results for the users, such as hydrographs, flow duration curves, and dotty plots based on the parameters obtained from the Pareto-optimal front.

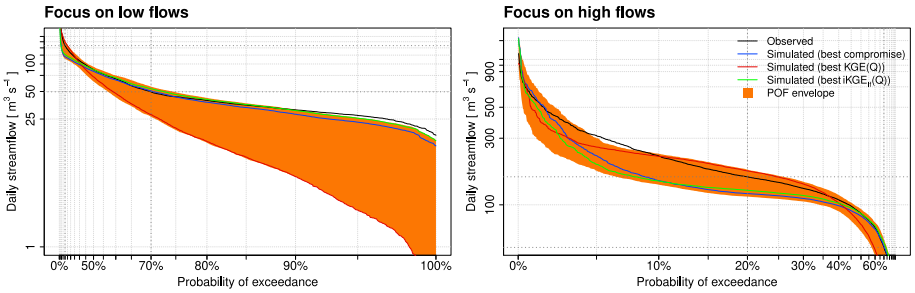

Fig. 11. Flow duration curves from the multi-objective calibration results, indicating the observed values, the envelope of simulations, and particular solutions of interest. In the left panel the focus is on low flows, while in the right panel, the focus is on high flows.

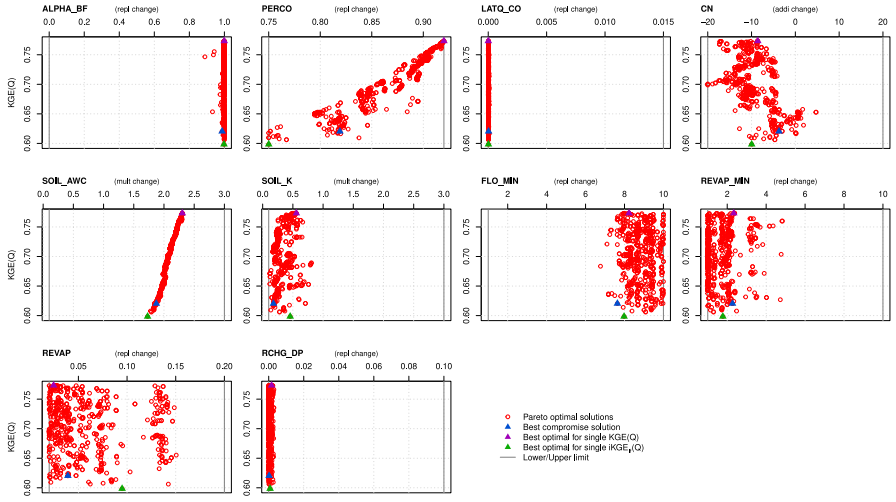

Fig. 12. Dotty plots showing the model performance according to $KGE(Q)$ vs. optimal parameter values of final Pareto-optimal front.

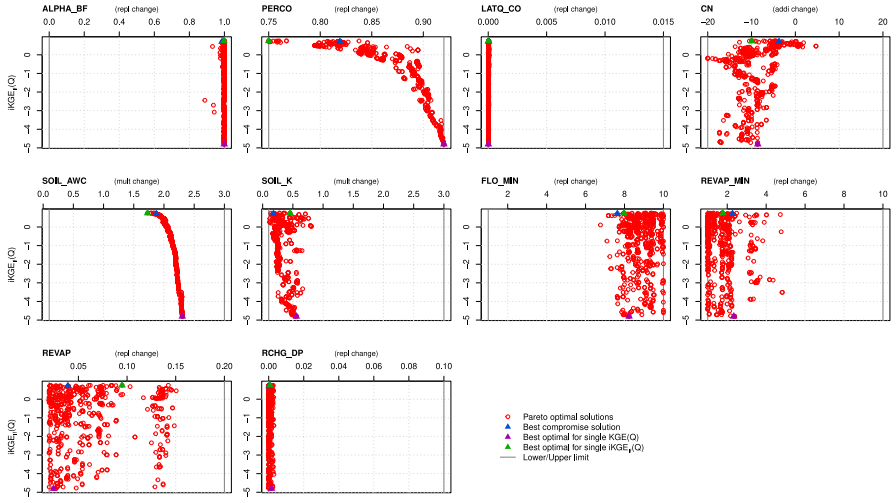

Fig. 13. Dotty plots showing the model performance according to $KGE_{lf}(Q)$ vs. optimal parameter values of final Pareto-optimal front.

`hydroMOPSO` was developed as a robust and user-friendly software for multi-objective calibration, offering extensive options for fine-tuning, flexibility and model independence, taking advantage of multi-core machines and network clusters. It allows the user to easily specify the main arguments of the algorithm, select a set of existing objective functions or define their own, define which model outputs need to be read for the calculation of the objective functions (including raster maps, time series or text files), providing excepcional customisation capabilities and ease of use.

In summary, in this article `hydroMOPSO` proved to be a robust and efficient R package for multi-objective optimisation of environmental and hydrological models. Its superior performance, adaptability, and ease of use, demonstrated by various R-based and R-external case studies, make it an attractive tool for researchers and practitioners trying to

use multi-objective optimisation to obtain robust model parameters and good simulated values for the right reasons. We encourage the scientific community to embrace `hydroMOPSO` and explore its potential for tackling complex multi-objective optimisation problems in the fields of environment, hydrology and beyond.

Software availability

`hydroMOPSO` is an R package publicly available on CRAN at the following URL: https://cran.r-project.org/package=hydroMOPSO.

Package Details:

- **Title:** Multi-Objective Optimisation with Focus on Environmental Models
- **Programming language:** R version 4.5.1 (2025-06-13) – "Great Square Root"
- **Version:** 0.1–14
- **Authors:** Rodrigo Marinao-Rivas (aut, cre, cph) and Mauricio Zambrano-Bigiarini (aut, ctb, cph)
- **Maintainer:** Rodrigo Marinao-Rivas (ra.marinao.rivas@gmail.com)
- **License:** GPL (≥ 2)
- **Source code:** https://gitlab.com/rmarinao/hydroMOPSO
- **Bug reports:** https://gitlab.com/rmarinao/hydroMOPSO/-/issues
- **CRAN Repository:** https://cran.r-project.org/package=hydroMOPSO
- **Date first available**: 2023-Apr-25
- **Packages used in this study**: caramel 1.4, TUWmodel 1.1-1, airGR 1.7.6.

CRediT authorship contribution statement

Rodrigo Marinao: Writing – original draft, Visualization, Software, Methodology, Investigation, Data curation, Conceptualization. **Mauricio Zambrano-Bigiarini:** Writing – review & editing, Supervision, Software, Resources, Project administration, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Oscar M. Baez-Villanueva:** Writing – review & editing, Validation.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used ChatGPT-4o and ChatGPT-5 to improve the readability and language of our manuscript. After using the tool, the authors reviewed and edited the content as needed to take full responsibility for the content of the published article.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was partially funded by ANID, Chile PCI-NSFC 190018 (*Management of global change impacts on hydrological extremes by coupling remote sensing data and an interdisciplinary modelling approach*) for the development of the package for specific objectives and data availability. It was also supported by ANID, Chile Fondecyt 1212071 (*The catchment's memory: understanding how hydrological extremes are modulated by antecedent soil moisture conditions in a warmer climate*), which provided hardware to accelerate the execution of numerous proof-of-concept trials and case studies. This research was also partially supported by the supercomputing infrastructure of the NLHPC, Chile (CCSS210001), and is a contribution to the TanDEM-X, Europe DEM_GEOL0845 and DEM_GEOL0707 projects. Finally, OMBV acknowledges funding from the European Space Agency project CCI Land Evaporation, Europe (contract no. 4000147355/25/I-LR).

Appendix A. `ParamFiles.txt` and `paramRanges.txt` input files

The calibration of R-external models using `hydroMOPSO` requires two input files: `ParamRanges.txt`, which specifies the minimum and maximum values of each parameter included in the calibration; and `ParamFiles.txt`, which defines where parameter values must be modified within the model input files. Together, these files allow the `hydroMOPSO` engine to update parameter values in the user-defined model across iterations

The `ParamRanges.txt` (see Table A.1) file contains as many rows as there are parameters to calibrate (D parameters), plus a header row. In detail:

- `ParameterNmbr` is a unique, consecutive identifier assigned to each parameter.
- `ParameterName` specifies the name of the model parameter. It is recommended to use the same identifier as in the useer-defined model (e.g., `CN2`).
- `MinValue` and `MaxValue` define the lower and upper bounds used for calibrating each parameter. These values may have different interpretations depending on the selected `ChangeType`.
- `ChangeType` defines how the parameter is modified during calibration, and can be specified as replacement (`repl`), multiplicative (`mult`), or additive (`addi`) change.
- `Min4Change` and `Max4Change` specify the absolute minimum and maximum allowable values for each model parameter during calibration. These fields are required only when `ChangeType` is set to multiplicative (`mult`) or additive (`addi`) change.

The `ParamFiles.txt` (see Table A.2) file might be considerably larger than `ParamRanges.txt` because changing the value of a single parameter might require a large number of modifications to the model input files. In detail:

- `ParameterNmbr` is a consecutive (and not necessarily unique) identifier assigned to each parameter. It must match the parameter number specified in the `ParamRanges.txt` file.
- `ParameterName` is a user-defined parameter label. It must correspond to the `ParameterName` used in the `ParamRanges.txt` file.
- `Filename` specifies the name of the model input file in which the parameter is located.
- `Row.Number` indicates the row number in the input file where the parameter value is found.
- `Col.Start` and `Col.End` define the starting and ending column positions in the input file where the parameter value should be written.
- `DecimalPlaces` specifies the number of decimal places used when writing the parameter values.
- `RefValue` defines the reference value used for multiplicative or additive changes. For multiplicative (`mult`) changes, parameter values are scaled relative to this reference value, while for additive (`addi`) changes, increments or decrements are applied around it. This value must be consistent with the parameter ranges specified in the `ParamRanges.txt` file.

Table A.1
ParamRanges.txt. This file contains the detail of the change of n parameters.

ParameterNmbr	ParameterName	MinValue	MaxValue	ChangeType	Min4Change	Max4Change
1	par_1	minval_1	maxval_1	typ_1	min_1	max_1
2	par_2	minval_2	maxval_2	typ_1	min_1	max_1
3	par_3	minval_3	maxval_3	typ_1	min_1	max_1
4	par_4	minval_4	maxval_4	typ_1	min_1	max_1
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
D	par_D	minval_D	maxval_D	typ_D	min_D	max_D

Table A.2
ParamFiles.txt. This file contains the detail of the m modifications needed to change n parameters indicated in ParamRanges.txt.

ParameterNmbr	ParameterName	Filename	Row.Number	Col.Start	Col.End	DecimalPlaces	RefValue
1	par_1	file_1	x_1	y_1	z_1	dec_1	ref_1
2	par_2	file_2	x_2	y_2	z_2	dec_2	ref_2
3	par_3	file_3	x_3	y_3	z_3	dec_3	ref_3
4	par_4	file_4	x_4	y_4	z_4	dec_4	ref_4
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
n	par_m	file_m	x_m	y_m	z_m	dec_m	ref_m

Appendix B. Function used to read an output variable from a model

As an example, the swat2zoo() function listed below is used to read a specific variable (name.col) from one output file (channel_sd_day.txt) of the SWAT+ hydrological model.

```
swat2zoo <- function(file.name = "channel_sd_day.txt",
id, first.date=NULL, last.date=NULL, name.col){

  require(zoo)

  header    <- read.table(file.name, skip=1, nrows=1,
                          sep="")
  header    <- as.character(header)
  indexVar  <- which(header == name.col)
  indexID   <- which(header == "gis_id")

  classes           <- rep("NULL", length(header))
  classes[indexID]  <- "character"
  classes[indexVar] <- "numeric"

  df.read       <- read.table(file.name, skip = 3,
                   colClasses = classes, sep="")
  all.id.out    <- as.character(df.read[,1])
  all.out       <- as.numeric(df.read[,2])
  index.id      <- all.id.out %in% id
  main.out      <- all.out[index.id]
  number.dates  <- length(main.out)
  indexDates    <- c(2,3,4)

  classesDates             <- rep("NULL", length(head))
  classesDates[indexDates] <- "numeric"

  df.date <- read.table(file.name, skip=3,
                        colClasses=classesDates,
                        sep="")[index.id,]

  dates <- as.Date(apply(df.date, FUN=paste,
                   collapse="-", MARGIN=1),
                   format = "%m-%d-%Y")

  out.zoo <- zoo::zoo(main.out, dates)

  if ( !is.null(first.date) )
    out.zoo <- window(out.zoo, start=first.date)

  if ( !is.null(last.date) )
    out.zoo <- window(out.zoo, end=last.date)

  return(out.zoo)
} # 'swat2zoo' END
```

Appendix C. Sensitivity analysis of TUWmodel

See Table C.1.

Appendix D. Sensitivity analysis of the SWAT+ model

See Table D.1.

Appendix E. Considerations on hydrological modelling with SWAT+

Since SWAT+ allows the representation of various hydrological processes, it was necessary to establish a simplified representation of the catchment that was simple enough for the scope of the present work, but robust enough to adequately represent large-scale hydrological processes.

The RTL sub-basin does not have extensive previous studies for the elaboration of a conceptual hydrological model, so we will state some key assumptions from our own field experience, which includes trial pits surveys and the collection of field data, information that can be consulted on the website https://chi2.ufro.cl/Datos/.

- Given the pluvial–nival regime characterised in the basin, parameters related to snow accumulation and melting may have some degree of importance, so they were considered in the sensitivity analysis.
- Recharge to the deep aquifer is not ruled out, as the background studies by GORE Araucanía - DGA (2016) show that at least 100 m depth to basement in the lower part of the basin, however, there is no data for the upper part of the basin. With this in mind, it was decided to keep the value of recharge to the deep aquifer at around the default value of 5%, specifically considering values between 0% and 10% in the sensitivity analysis and calibration.
- Although estimates of saturated hydraulic conductivity (K_{sat}) and available water capacity (AWC) are available from the SoilGrids-based product, only the spatial variability of these data from the discretisations achieved according to soil textures was taken into account. Thus, these can be adjusted with a factor, which in the context of the calibration with hydroMOPSO will be called multiplicative change (mult). This is motivated by the uncertainty of these parameters when direct field measurements are not even available.

Table C.1
`TUWmodel` parameters considered for the Sobol' sensitivity analysis. The first ten parameters were used for the calibration, ranked according to the Sobol' total order index (Sobol' TOI) value.

ID	Short description	Unit	Process	Range	Sobol TOI	Ranking
cperc	Constant percolation rate	mm/day	Infiltration	0.0–8.0	0.619168	1
k2	Storage coefficient for slow response	day	Runoff	30–250	0.281497	2
BETA	Non-linear parameter for runoff production	–	Infiltration	0.0–20	0.102511	3
FC	Field capacity	mm	Infiltration	0.0–600	0.044742	4
k1	Storage coefficient for fast response	day	Runoff	2.0–30	0.033318	5
lsuz	Threshold storage state	mm	Runoff	1.0–100	0.028031	6
k0	Storage coefficient for very fast response	day	Runoff	0.0–2.0	0.026030	7
LPrat	Parameter related to the limit for potential evaporation	–	Evaporation	0.0–1.0	0.012672	8
DDF	Degree-day factor	mm/°C/day	Snow	0.0–5.0	0.008622	9
croute	Free scaling parameter	day^2/mm	Runoff	0.0–50	0.006355	10
Twb	Wet bulb temperature	°C	Snow	−3.0–3.0	0.006137	11
bmax	Maximum base at low flows	day	Runoff	0.0–30	0.005714	12
Tm	Temperature threshold above which melt starts	°C	Snow	−2.0–2.0	0.000394	13
SCF	Snow correction factor	–	Snow	0.9–1.5	0.000328	14

Table D.1
`SWAT+` model parameters considered for the Sobol' sensitivity analysis. The first ten parameters were used for the calibration, ranked according to the Sobol' total order index (Sobol' TOI) value. The **Change** field refers to the type of change used for each model parameter during the MO optimisation, with `repl`, `mult` and `add` representing direct replacement, multiplicative change and additive change, respectively.

ID	Short description	Units	Filename	Change	Range	Sobol TOI	Ranking
soil_awc	Available water capacity	–	soils.sol	mult	0.1–3.0	0.402039	1
soil_k	Saturated hydraulic conductivity	mm/h	soils.sol	mult	0.1–3.0	0.237215	2
latq_co	Lateral soil flow coefficient - linear adjustment	–	hydrology.hyd	repl	0.000–0.015	0.180337	3
perco	Percolation coefficient	–	hydrology.hyd	repl	0.75–0.92	0.118543	4
revap_min	Threshold depth of water for revap to occur	m	aquifer.aqu	repl	1–10	0.086714	5
flo_min	Minimum aquifer storage to allow return flow	m	aquifer.aqu	repl	1–10	0.080658	6
alpha_bf	Baseflow alpha factor	1/day	aquifer.aqu	repl	0–1	0.050953	7
revap	Groundwater revap coefficient	–	aquifer.aqu	repl	0.02–0.2	0.033571	8
rchg_dp	Deep aquifer percolation fraction	–	aquifer.aqu	repl	0.0–0.1	0.006292	9
CN2	Curve number	–	cntable.lum	mult	0.6–1.2	0.004502	10
melt_max	Maximum melt rate for snow during year	mm/°C/day	snow.sno	repl	0–10	0.001314	11
fall_tmp	Snowfall temperature	°C	snow.sno	repl	−5–5	0.001125	12
melt_min	Minimum melt rate for snow during year	mm/°C/day	snow.sno	repl	0–10	0.000887	13
spec_yld	Specific yield of the shallow aquifer	m^3/m^3	aquifer.aqu	repl	0–0.4	0.000522	14
can_max	Maximum canopy storage	mm	hydrology.hyd	repl	1–10	0.000482	15
melt_tmp	Snow melt base temperature	°C	snow.sno	repl	−5–5	0.000349	16
rt_dp_max	Maximum root depth	m	plants.plt	mult	0.75–1.15	0.000053	17

- The soil horizon was set to a depth of 3 m in all soil classes discretised from the SoilGrids data. This is assumed on the basis that deep root growth exceeding 2 m has been observed in native forest areas. This was done by simply extending the last soil layer.
- Rooting depth was identified as a sensitive parameter in preliminary analyses of the catchment (not shown). It was selected for calibration due to (*i*) its strong influence on evaporative processes of tall vegetation (e.g., forests) through its control on vegetation access to groundwater, and (*ii*) its role in regulating soil moisture availability, which in turn affects transpiration rates and the partitioning between evaporation and runoff. Additionally, rooting depth can modulate ecosystem resilience to drought conditions, thereby influencing the hydrological response at both seasonal and interannual scales.

Data availability

• Marinao, R.; Zambrano-Bigiarini, M.; Baez-Villanueva, O. M. (2025). Supplementary material for manuscript ENVSOFT-D-25-01387 "hydroMOPSO: a flexible and model-independent multi-objective optimisation R package for environmental and hydrological models" by R. Marinao, M. Zambrano-Bigiarini, and O. Baez-Villanueva (Version 1). Zenodo. https://doi.org/10.5281/zenodo.15743400.

• Marinao, R.; Zambrano-Bigiarini, M.; Baez-Villanueva, O. M.; Muñoz-Neira, R. (2025). Tutorial for using hydroMOPSO to calibrate SWAT+ (v0.1). Zenodo. https://doi.org/10.5281/zenodo.17273537.

• Zambrano-Bigiarini, M.; Marinao, R.; Baez-Villanueva, O. M.; Muñoz-Neira, R. (2025). Tutorial for using hydroMOPSO to calibrate TUWmodel (v0.1). Zenodo. https://doi.org/10.5281/zenodo.17273949.

The first dataset contains supplementary materials used to generate the results in the manuscript, including R scripts, calibration outputs, and visualisations. The second and third datasets contain detailed tutorials about how to use hydroMOPSO to carry out a multi-objective calibration of SWAT+ and TUWmodel, respectively. Examples of benchmark functions are included in the `hydroMOPSO` package documentation and do not require external data.

References

Arnold, J.G., Čerkasova, N., White, M.J., Bailey, R., Thorp, K., Jeong, J., Zhang, X., Ugraskan, T., van Griensven, A., Rathjens, H., Raj, C., Cai, X., Geter, W.F., David, O., Carlson, J.R., Le, K.N., 2024. Soil and water assessment tool plus (SWAT+). URL: https://github.com/swat-model/swatplus.

Arnold, J.G., Srinivasan, R., Muttiah, R.S., Williams, J.R., 1998. Large area hydrologic modeling and assessment Part I: Model development. JAWRA J. Am. Water Resour. Assoc. 34, 73–89. http://dx.doi.org/10.1111/J.1752-1688.1998.TB05961.X.

Baez-Villanueva, O.M., Zambrano-Bigiarini, M., Mendoza, P.A., McNamara, I., Beck, H.E., Thurner, J., Nauditt, A., Ribbe, L., Thinh, N.X., 2021. On the selection of precipitation products for the regionalisation of hydrological model parameters. Hydrol. Earth Syst. Sci. 25, 5805–5837. http://dx.doi.org/10.5194/hess-25-5805-2021.

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

Barría, P., Chadwick, C., Ocampo-Melgar, A., Galleguillos, M., Garreaud, R., Díaz-Vasconcellos, R., Poblete, D., Rubio-Álvarez, E., Poblete-Caballero, D., 2021. Water management or megadrought: what caused the Chilean aculeo lake drying? Reg. Environ. Chang. 21, 1–15. http://dx.doi.org/10.1007/S10113-021-01750-W/FIGURES/6.

Benitez, F., Pinto-Roa, D.P., 2022. rmoo: Multi-objective optimization in R. URL: https://CRAN.R-project.org/package=rmoo/. R package version 0.2.2.

Benítez-Hidalgo, A., Nebro, A.J., García-Nieto, J., Oregi, I., Del Ser, J., 2019. jMetalPy: A Python framework for multi-objective optimization with metaheuristics. Swarm Evol. Comput. 51, 100598. http://dx.doi.org/10.1016/J.SWEVO.2019.100598.

Bergström, S., Forsman, A., 1973. Development of a conceptual deterministic rainfall-runoff model. Hydrol. Res. 4, 147–170. http://dx.doi.org/10.2166/NH.1973.0012.

Beven, K., 1993. Prophecy, reality and uncertainty in distributed hydrological modelling. Adv. Water Resour. 16 (1), 41–51. http://dx.doi.org/10.1016/0309-1708(93)90028-E.

Beven, K., 2006. A manifesto for the equifinality thesis. J. Hydrol. 320 (1), 18–36. http://dx.doi.org/10.1016/j.jhydrol.2005.07.007.

Beven, K., Binley, A., 1992. The future of distributed models: Model calibration and uncertainty prediction. Hydrol. Process. 6 (3), 279–298. http://dx.doi.org/10.1002/hyp.3360060305.

Beykal, B., Boukouvala, F., Floudas, C., Pistikopoulos, E., 2018. Optimal design of energy systems using constrained grey-box multi-objective optimization. Comput. Chem. Eng. 116, 488–502. http://dx.doi.org/10.1016/j.compchemeng.2018.02.017.

Bieger, K., Arnold, J.G., Rathjens, H., White, M.J., Bosch, D.D., Allen, P.M., Volk, M., Srinivasan, R., 2017. Introduction to SWAT+, A completely restructured version of the soil and water assessment tool. JAWRA J. Am. Water Resour. Assoc. 53, 115–130. http://dx.doi.org/10.1111/1752-1688.12482.

Binois, M., Picheny, V., 2019. GPareto: An R package for Gaussian-process-based multi-objective optimization and analysis. J. Stat. Softw. 89 (8), 1–30. http://dx.doi.org/10.18637/jss.v089.i08.

Bisselink, B., Zambrano-Bigiarini, M., Burek, P., de Roo, A., 2016. Assessing the role of uncertain precipitation estimates on the robustness of hydrological model parameters under highly variable climate conditions. J. Hydrol.: Reg. Stud. 8, 112–129. http://dx.doi.org/10.1016/j.ejrh.2016.09.003.

Blank, J., Deb, K., 2020. Pymoo: Multi-objective optimization in Python. IEEE Access 8, 89497–89509. http://dx.doi.org/10.1109/ACCESS.2020.2990567.

Boisier, J.P., 2023. CR2MET: A high-resolution precipitation and temperature dataset for the period 1960–2021 in continental Chile. http://dx.doi.org/10.5281/zenodo.7529682, URL: https://zenodo.org/records/7529682.

Bossek, J., 2017. Smoof: Single- and multi-objective optimization test functions. R J. URL: https://journal.r-project.org/archive/2017/RJ-2017-004/index.html.

Cazzaniga, P., Nobile, M.S., Besozzi, D., 2015. The impact of particles initialization in PSO: Parameter estimation as a case in point. In: 2015 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology. CIBCB, IEEE, Niagara Falls, ON, Canada, pp. 1–8. http://dx.doi.org/10.1109/CIBCB.2015.7300288, URL: http://ieeexplore.ieee.org/document/7300288/.

Ceola, S., Arheimer, B., Baratti, E., Blöschl, G., Capell, R., Castellarin, A., Freer, J., Han, D., Hrachowitz, M., Hundecha, Y., Hutton, C., Lindström, G., Montanari, A., Nijzink, R., Parajka, J., Toth, E., Viglione, A., Wagener, T., 2015. Virtual laboratories: New opportunities for collaborative water science. Hydrol. Earth Syst. Sci. 19, 2101–2117. http://dx.doi.org/10.5194/HESS-19-2101-2015.

Coello, C.A., 1999. A comprehensive survey of evolutionary-based multiobjective optimization techniques. Knowl. Inf. Syst. 1 (3), 269–308. http://dx.doi.org/10.1007/BF03325101.

Coello, C.A., 2005. Recent trends in evolutionary multiobjective optimization. Evol. Multiobjective Optim. 7–32. http://dx.doi.org/10.1007/1-84628-137-7_2.

Coello, C.A., Lamont, G.B., 2004. Applications of Multi-Objective Evolutionary Algorithms. World Scientific, http://dx.doi.org/10.1142/5712.

Coello, C.A., Lamont, G.B., Van-Veldhuizen, D.A., 2007. Evolutionary Algorithms for Solving Multi-Objective Problems. Springer US, Boston, MA, http://dx.doi.org/10.1007/978-0-387-36797-2_1.

Coello, C.A., Lechuga, M.S., 2002. MOPSO: A proposal for multiple objective particle swarm optimization. In: Proceedings of the 2002 Congress on Evolutionary Computation. CEC 2002, Vol. 2, IEEE Computer Society, pp. 1051–1056. http://dx.doi.org/10.1109/CEC.2002.1004388.

Coron, L., Andréassian, V., Perrin, C., Bourqui, M., Hendrickx, F., 2014. On the lack of robustness of hydrologic models regarding water balance simulation: a diagnostic approach applied to three models of increasing complexity on 20 mountainous catchments. Hydrol. Earth Syst. Sci. 18 (2), 727–746. http://dx.doi.org/10.5194/hess-18-727-2014.

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

Coron, L., Delaigue, O., Thirel, G., Dorchies, D., Perrin, C., Michel, C., 2022. airGR: Suite of GR hydrological models for precipitation-runoff modelling. http://dx.doi.org/10.15454/EX11NA, URL: https://CRAN.R-project.org/package=airGR. R package version 1.7.0.

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

Craig, J.R., Brown, G., Chlumsky, R., Jenkinson, R.W., Jost, G., Lee, K., Mai, J., Serrer, M., Sgro, N., Shafii, M., Snowdon, A.P., Tolson, B.A., 2020. Flexible watershed simulation with the raven hydrological modelling framework. Environ. Model. Softw. 129, 104728. http://dx.doi.org/10.1016/J.ENVSOFT.2020.104728.

Dasgupta, A., Arnal, L., Emerton, R., Harrigan, S., Matthews, G., Muhammad, A., O'Regan, K., Pérez-Ciria, T., Valdez, E., van Osnabrugge, B., Werner, M., Buontempo, C., Cloke, H., Pappenberger, F., Pechlivanidis, I.G., Prudhomme, C., Ramos, M.H., Salamon, P., 2023. Connecting hydrological modelling and forecasting from global to local scales: Perspectives from an international joint virtual workshop. J. Flood Risk Manag. http://dx.doi.org/10.1111/JFR3.12880.

Deb, K., 2014. Multi-objective optimization. Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques, second ed. Springer US, pp. 403–450. http://dx.doi.org/10.1007/978-1-4614-6940-7_15.

Deb, K., 2015. Multi-Objective Evolutionary Algorithms. Springer, Berlin, Heidelberg, pp. 995–1015. http://dx.doi.org/10.1007/978-3-662-43505-2_49, Chapter 49.

Deb, K., Jain, H., 2014. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints. IEEE Trans. Evol. Comput. 18, 577–601. http://dx.doi.org/10.1109/TEVC.2013.2281535.

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. 6, 182–197. http://dx.doi.org/10.1109/4235.996017.

Deb, K., Thiele, L., Laumanns, M., Zitzler, E., 2005. Scalable test problems for evolutionary multiobjective optimization. In: Evolutionary Multiobjective Optimization: Theoretical Advances and Applications. Springer London, London, pp. 105–145. http://dx.doi.org/10.1007/1-84628-137-7_6, Chapter 6.

Demirel, M.C., Mai, J., Mendiguren, G., Koch, J., Samaniego, L., Stisen, S., 2018. Combining satellite data and appropriate objective functions for improved spatial pattern performance of a distributed hydrologic model. Hydrol. Earth Syst. Sci. 22 (2), 1299–1315. http://dx.doi.org/10.5194/hess-22-1299-2018.

Dinamarca, D.I., Galleguillos, M., Seguel, O., Faúndez Urbina, C., 2023. CLSoilMaps: A national soil gridded database of physical and hydraulic soil properties for Chile. Sci. Data 10 (1), 630. http://dx.doi.org/10.1038/s41597-023-02536-x, URL: https://www.nature.com/articles/s41597-023-02536-x. Publisher: Nature Publishing Group.

Douglas-Mankin, K.R., Srinivasan, R., Arnold, J.G., Raghavan Srinivasan, K.., Member, A., Douglas-Mankin, K.R., 2010. Soil and water assessment tool (SWAT) model: Current developments and applications. Trans. ASABE 53 (5), 1423–1431. http://dx.doi.org/10.13031/2013.34915.

Duan, Q., Sorooshian, S., Gupta, V., 1992. Effective and efficient global optimization for conceptual rainfall-runoff models. Water Resour. Res. 28, 1015–1031. http://dx.doi.org/10.1029/91WR02985.

Efstratiadis, A., Koutsoyiannis, D., 2008. Fitting hydrological models on multiple responses using the multiobjective evolutionary annealing-simplex approach. In: Practical Hydroinformatics: Computational Intelligence and Technological Developments in Water Applications. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 259–273. http://dx.doi.org/10.1007/978-3-540-79881-1_19, Chapter 19.

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

Ehrgott, M., 2005. Efficiency and nondominance. In: Multicriteria Optimization. Springer Berlin Heidelberg, Berlin - Heidelberg, pp. 23–64. http://dx.doi.org/10.1007/3-540-27659-9_2, Chapter 2.

Fonseca, C., Fleming, P., 1993. Genetic algorithms for multi-objective optimization: Formulation, discussion and generalization. In: Proceedings of the Fifth International Conference on Genetic Algorithms. Morgan Kauffman Publishers, San Mateo, California, pp. 416–423.

Galleguillos, M., Ceballos-Comisso, A., Gimeno, F., Zambrano-Bigiarini, M., 2024. CLDynamicLandCover. http://dx.doi.org/10.5281/zenodo.13153631, URL: https://zenodo.org/records/13153631.

Garcia, F., Folton, N., Oudin, L., 2017. Which objective function to calibrate rainfall–runoff models for low-flow index simulations? Hydrol. Sci. J. 62, 1149–1166. http://dx.doi.org/10.1080/02626667.2017.1308511.

Gassman, P.W., Reyes, M.R., Green, C.H., Arnold, J.G., 2007. The soil and water assessment tool: Historical development, applications, and future research directions. Trans. ASABE 50 (4), 1211–1250. http://dx.doi.org/10.13031/2013.23637.

Gharari, S., Hrachowitz, M., Fenicia, F., Savenije, H.H., 2013. An approach to identify time consistent model parameters: Sub-period calibration. Hydrol. Earth Syst. Sci. 17, 149–161. http://dx.doi.org/10.5194/HESS-17-149-2013.

GORE Araucanía - DGA, 2016. Estudio Hidrogeológico, Región de La Araucanía - Realizado por Arcadis Chile S.A.. Technical Report, Dirección General de Aguas, Santiago, URL: https://snia.mop.gob.cl/repositoriodga/handle/20.500.13000/6930.

Gupta, H.V., Kling, H., Yilmaz, K.K., Martinez, G.F., 2009. Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling. J. Hydrol. 377, 80–91. http://dx.doi.org/10.1016/J.JHYDROL.2009.08.003.

Gupta, H.V., Sorooshian, S., Yapo, P.O., 1998. Toward improved calibration of hydrologic models: Multiple and noncommensurable measures of information. Water Resour. Res. 34, 751–763. http://dx.doi.org/10.1029/97WR03495.

Guswa, A.J., Brauman, K.A., Brown, C., Hamel, P., Keeler, B.L., Stratton Sayre, S., 2014. Ecosystem services: Challenges and opportunities for hydrologic modeling to support decision making. Water Resour. Res. 50, 4535–4544. http://dx.doi.org/10.1002/2014WR015497.

Hartmann, A., Goldscheider, N., Wagener, T., Lange, J., Weiler, M., 2014. Karst water resources in a changing world: Review of hydrological modeling approaches. Rev. Geophys. 52, 218–242. http://dx.doi.org/10.1002/2013RG000443.

Her, Y., Seong, C., 2018. Responses of hydrological model equifinality, uncertainty, and performance to multi-objective parameter calibration. J. Hydroinformatics 20 (4), 864–885. http://dx.doi.org/10.2166/hydro.2018.108.

Herrera, P.A., Marazuela, M.A., Hofmann, T., 2022. Parameter estimation and uncertainty analysis in hydrological modeling. WIREs Water 9 (1), e1569. http://dx.doi.org/10.1002/wat2.1569.

Hu, W., Yen, G.G., 2015. Adaptive multiobjective particle swarm optimization based on parallel cell coordinate system. IEEE Trans. Evol. Comput. 19 (1), 1–18. http://dx.doi.org/10.1109/TEVC.2013.2296151.

Jayakrishnan, R., Srinivasan, R., Santhi, C., Arnold, J.G., 2005. Advances in the application of the SWAT model for water resources management. Hydrol. Process. 19, 749–762. http://dx.doi.org/10.1002/HYP.5624.

Jung, D., Choi, Y., Kim, J., 2017. Multiobjective automatic parameter calibration of a hydrological model. Water 9 (3), 187. http://dx.doi.org/10.3390/w9030187, URL: https://www.mdpi.com/2073-4441/9/3/187.

Kauffeldt, A., Wetterhall, F., Pappenberger, F., Salamon, P., Thielen, J., 2016. Technical review of large-scale hydrological models for implementation in operational flood forecasting schemes on continental level. Environ. Model. Softw. 75, 68–76. http://dx.doi.org/10.1016/J.ENVSOFT.2015.09.009.

Kavetski, D., Fenicia, F., 2011. Elements of a flexible approach for conceptual hydrological modeling: 2. Application and experimental insights. Water Resour. Res. 47 (11), W11511. http://dx.doi.org/10.1029/2011wr010748.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proceedings of ICNN'95 - International Conference on Neural Networks. Vol. 4, IEEE, pp. 1942–1948. http://dx.doi.org/10.1109/ICNN.1995.488968.

Khu, S.T., Madsen, H., di Pierro, F., 2008. Incorporating multiple observations for distributed hydrologic model calibration: An approach using a multi-objective evolutionary algorithm and clustering. Adv. Water Resour. 31, 1387–1398. http://dx.doi.org/10.1016/J.ADVWATRES.2008.07.011.

Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. Optimization by simulated annealing. Science 220, 671–680. http://dx.doi.org/10.1126/SCIENCE.220.4598.671.

Klemeš, V., 1986a. Dilettantism in hydrology: Transition or destiny? Water Resour. Res. 22 (9S), 177S–188S. http://dx.doi.org/10.1029/WR022i09Sp0177S.

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

Kogiso, N., Kodama, R., Toyoda, M., 2014. Reliability-based multiobjective optimization using the satisficing trade-off method. Mech. Eng. J. 1, DSM0063. http://dx.doi.org/10.1299/mej.2014dsm0063.

Kollat, J.B., Reed, P.M., 2005. The value of online adaptive search: A performance comparison of NSGAII, ϵ-NSGAII and ϵMOEA. Lecture Notes in Comput. Sci. 3410, 386–398. http://dx.doi.org/10.1007/978-3-540-31880-4_27.

Lin, Q., Li, J., Du, Z., Chen, J., Ming, Z., 2015. A novel multi-objective particle swarm optimization with multiple search strategies. European J. Oper. Res. 247, 732–744. http://dx.doi.org/10.1016/J.EJOR.2015.06.071.

Lin, Q., Liu, S., Zhu, Q., Tang, C., Song, R., Chen, J., Coello, C.A.C., Wong, K.-C., Zhang, J., 2016. Particle swarm optimization with a balanceable fitness estimation for many-objective optimization problems. IEEE Trans. Evol. Comput. 22 (1), 32–46. http://dx.doi.org/10.1109/TEVC.2016.2631279.

Luo, J., Chen, C., Xie, J., 2014. Multi-objective immune algorithm with preference-based selection for reservoir flood control operation. Water Resour. Manag. 29, 1447–1466. http://dx.doi.org/10.1007/s11269-014-0886-6.

Madsen, H., 2003. Parameter estimation in distributed hydrological catchment modelling using automatic calibration with multiple objectives. Adv. Water Resour. 26, 205–216. http://dx.doi.org/10.1016/S0309-1708(02)00092-1.

Marinao-Rivas, R., Zambrano-Bigiarini, M., 2021. Towards best default configuration settings for NMPSO in multi-objective optimization. In: 2021 IEEE Latin American Conference on Computational Intelligence. LA-CCI 2021, Institute of Electrical and Electronics Engineers Inc, http://dx.doi.org/10.1109/LA-CCI48322.2021.9769844.

McColl, C., Aggett, G., 2007. Land-use forecasting and hydrologic model integration for improved land-use decision support. J. Environ. Manag. 84, 494–512. http://dx.doi.org/10.1016/J.JENVMAN.2006.06.023.

McKay, M.D., Beckman, R.J., Conover, W.J., 1979. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21 (2), 239–245. http://dx.doi.org/10.2307/1268522, URL: https://www.jstor.org/stable/1268522. Publisher: [Taylor & Francis, Ltd., American Statistical Association, American Society for Quality].

Miettinen, K., 1998. Concepts. In: Nonlinear Multiobjective Optimization. Springer US, Boston, MA, pp. 23–25. http://dx.doi.org/10.1007/978-1-4615-5563-6_1, Chapter 2.

Minville, M., Cartier, D., Guay, C., Leclaire, L.-A., Audet, C., Le Digabel, S., Merleau, J., 2014. Improving process representation in conceptual hydrological model calibration using climate simulations. Water Resour. Res. 50 (6), 5044–5073. http://dx.doi.org/10.1002/2013wr013857.

Monteil, C., Zaoui, F., Le Moine, N., Hendrickx, F., 2020. Multi-objective calibration by combination of stochastic and gradient-like parameter generation rules - The caRamel algorithm. Hydrol. Earth Syst. Sci. 24, 3189–3209. http://dx.doi.org/10.5194/HESS-24-3189-2020.

Moore, J., Chapman, R., Dozier, G., 2000. Multiobjective particle swarm optimization. Proc. Annu. Southeast Conf. 2000-April, 56–57. http://dx.doi.org/10.1145/1127716.1127729.

Mostafaie, A., Forootan, E., Safari, A., Schumacher, M., 2018. Comparing multi-objective optimization techniques to calibrate a conceptual hydrological model using in situ runoff and daily GRACE data. Comput. Geosci. 22, 789–814. http://dx.doi.org/10.1007/S10596-018-9726-8, 2018 22:3.

Naghdi, S., Bozorg-Haddad, O., Khorsandi, M., Chu, X., 2021. Multi-objective optimization for allocation of surface water and groundwater resources. Sci. Total Environ. 776, 146026. http://dx.doi.org/10.1016/J.SCITOTENV.2021.146026.

Nebro, A.J., Durillo, J.J., Nieto, G., Coello, C.A., Luna, F., Alba, E., 2009. SMPSO: A new pso-based metaheuristic for multi-objective optimization. In: 2009 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making, MCDM 2009 - Proceedings. pp. 66–73. http://dx.doi.org/10.1109/MCDM.2009.4938830.

Niswonger, R.G., Panday, S., Ibaraki, M., 2011. MODFLOW-NWT, a Newton formulation for MODFLOW-2005. Tech. Methods http://dx.doi.org/10.3133/tm6A37.

Parajka, J., Merz, R., Blöschl, G., 2007. Uncertainty and multiple objective calibration in regional water balance modelling: case study in 320 Austrian catchments. Hydrol. Process. 21, 435–446. http://dx.doi.org/10.1002/HYP.6253.

Paul, M., Rajib, A., Negahban-Azar, M., Shirmohammadi, A., Srivastava, P., 2021. Improved agricultural water management in data-scarce semi-arid watersheds: Value of integrating remotely sensed leaf area index in hydrological modeling. Sci. Total Environ. 791, 148177. http://dx.doi.org/10.1016/J.SCITOTENV.2021.148177.

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

Perrin, C., Oudin, L., Andreassian, V., Rojas-Serna, C., Michel, C., Mathevet, T., 2007. Impact of limited streamflow data on the efficiency and the parameters of rainfall-runoff models. Hydrol. Sci. J. 52 (1), 131–151. http://dx.doi.org/10.1623/hysj.52.1.131.

Pushpalatha, R., Perrin, C., Le Moine, N., Andréassian, V., 2012. A review of efficiency criteria suitable for evaluating low-flow simulations. J. Hydrol. 420–421, 171–182. http://dx.doi.org/10.1016/J.JHYDROL.2011.11.055.

Qi, W., Zhang, C., Fu, G., Sweetapple, C., Liu, Y., 2019. Impact of robustness of hydrological model parameters on flood prediction uncertainty. J. Flood Risk Manag. 12 (S1), e12488. http://dx.doi.org/10.1111/jfr3.12488.

Rafiei, V., Nejadhashemi, A.P., Mushtaq, S., Bailey, R.T., An-Vo, D.A., 2022. An improved calibration technique to address high dimensionality and non-linearity in integrated groundwater and surface water models. Environ. Model. Softw. 149, 105312. http://dx.doi.org/10.1016/J.ENVSOFT.2022.105312.

Reed, P., Minsker, B.S., Goldberg, D.E., 2003. Simplifying multiobjective optimization: An automated design methodology for the nondominated sorted genetic algorithm-II. Water Resour. Res. 39, 1196. http://dx.doi.org/10.1029/2002WR001483.

Refsgaard, J.C., Henriksen, H.J., 2004. Modelling guidelines—terminology and guiding principles. Adv. Water Resour. 27, 71–82. http://dx.doi.org/10.1016/J.ADVWATRES.2003.08.006.

Ringuest, J.L., 1992. Multiobjective Optimization: Behavioral and Computational Considerations. Springer US, Boston, MA, pp. 51–59. http://dx.doi.org/10.1007/978-1-4615-3612-3_4, Chapter 4.

Riquelme, N., Von Lucken, C., Baran, B., 2015. Performance metrics in multi-objective optimization. In: 2015 Latin American Computing Conference. CLEI, p. 1. http://dx.doi.org/10.1109/CLEI.2015.7360024.

Rosenqvist, A., Shimada, M., Ito, N., Watanabe, M., 2007. ALOS PALSAR: A pathfinder mission for global-scale monitoring of the environment. IEEE Trans. Geosci. Remote Sens. 45, 3307–3316. http://dx.doi.org/10.1109/TGRS.2007.901027.

Royer-Gaspard, P., Andréassian, V., Thirel, G., 2021. Technical note: PMR - a proxy metric to assess hydrological model robustness in a changing climate. Hydrol. Earth Syst. Sci. 25 (11), 5703–5716. http://dx.doi.org/10.5194/hess-25-5703-2021.

Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, M., Tarantola, S., 2010. Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index. Comput. Phys. Comm. 181, 259–270. http://dx.doi.org/10.1016/J.CPC.2009.09.018.

Savenije, H.H.G., 2009. Hess opinions "The art of hydrology". Hydrol. Earth Syst. Sci. 13 (2), 157–161. http://dx.doi.org/10.5194/hess-13-157-2009.

Schaffer, J.D., 1985. Multiple objective optimization with vector evaluated genetic algorithms. In: Proceedings of the First International Conference on Genetic Algorithms and their Applications. Psychology Press, pp. 93–100. http://dx.doi.org/10.4324/9781315799674.

Schuol, J., Abbaspour, K.C., Srinivasan, R., Yang, H., 2008. Estimation of freshwater availability in the west African sub-continent using the SWAT hydrologic model. J. Hydrol. 352, 30–49. http://dx.doi.org/10.1016/J.JHYDROL.2007.12.025.

Shafii, M., De Smedt, F., 2009. Multi-objective calibration of a distributed hydrological model (WetSpa) using a genetic algorithm. Hydrol. Earth Syst. Sci. 13 (11), 2137–2149. http://dx.doi.org/10.5194/hess-13-2137-2009, URL: https://hess.copernicus.org/articles/13/2137/2009/. Publisher: Copernicus GmbH.

Shafii, M., Tolson, B.A., 2015. Optimizing hydrological consistency by incorporating hydrological signatures into model calibration objectives. Water Resour. Res. 51, 3796–3814. http://dx.doi.org/10.1002/2014WR016520.

Sierra, M., Coello, C.A., 2005. Improving PSO-based multi-objective optimization using crowding, mutation and e-dominance. Lecture Notes in Comput. Sci. 3410, 505–519. http://dx.doi.org/10.1007/978-3-540-31880-4_35.

Singer, M.B., Asfaw, D.T., Rosolem, R., Cuthbert, M.O., Miralles, D.G., MacLeod, D., Quichimbo, E.A., Michaelides, K., 2021. Hourly potential evapotranspiration at 0.1° resolution for the global land surface from 1981-present. Sci. Data 8, 1–13. http://dx.doi.org/10.1038/s41597-021-01003-9, 2021 8:1.

Sobol, I.M., 1967. On the distribution of points in a cube and the approximate evaluation of integrals. USSR Comput. Math. Math. Phys. 7 (4), 86–112. http://dx.doi.org/10.1016/0041-5553(67)90144-9.

Sobol, I.M., 2001. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Math. Comput. Simulation 55, 271–280. http://dx.doi.org/10.1016/S0378-4754(00)00270-6.

Srinivas, N., Deb, K., 1994. Muiltiobjective optimization using nondominated sorting in genetic algorithms. Evol. Comput. 2, 221–248. http://dx.doi.org/10.1162/EVCO.1994.2.3.221.

Storn, R., Price, K., 1997. Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. 11, 341–359. http://dx.doi.org/10.1023/A:1008202821328, 1997 11:4.

Sun, X., Shan, R., Liu, F., 2020. Spatio-temporal quantification of patterns, trade-offs and synergies among multiple hydrological ecosystem services in different topographic basins. J. Clean. Prod. 268, 122338. http://dx.doi.org/10.1016/J.JCLEPRO.2020.122338.

Széles, B., Parajka, J., Hogan, P., Silasari, R., Pavlin, L., Strauss, P., Blöschl, G., 2020. The added value of different data types for calibrating and testing a hydrologic model in a small catchment. Water Resour. Res. 56, e2019WR026153. http://dx.doi.org/10.1029/2019WR026153.

Tang, Y., Reed, P., Wagener, T., 2006. How effective and efficient are multiobjective evolutionary algorithms at hydrologic model calibration? Hydrol. Earth Syst. Sci. 10 (2), 289–307. http://dx.doi.org/10.5194/hess-10-289-2006.

Tolson, B.A., Shoemaker, C.A., 2007. Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. Water Resour. Res. 43, 1413. http://dx.doi.org/10.1029/2005WR004723.

Van Veldhuizen, D.A., 1999. Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations. Air Force Institute of Technology, URL: https://scholar.afit.edu/etd/5128/.

Velea, M., Lache, S., 2015. Decision making process on multi-objective optimization results. Int. J. Mater. Mech. Manuf. 4, 213–217. http://dx.doi.org/10.7763/ijmmm.2016.v4.259.

Viglione, A., Parajka, J., 2020. TUWmodel: Lumped/semi-distributed hydrological model for education purposes. URL: https://CRAN.R-project.org/package=TUWmodel. R package version 1.1-1.

Vrugt, J.A., Gupta, H.V., Bastidas, L.A., Bouten, W., Sorooshian, S., 2003. Effective and efficient algorithm for multiobjective optimization of hydrologic models. Water Resour. Res. 39, 1214. http://dx.doi.org/10.1029/2002WR001746.

Wagener, T., 2003. Evaluation of catchment models. Hydrol. Process. 17, 3375–3378. http://dx.doi.org/10.1002/HYP.5158.

Wagener, T., Gupta, H.V., 2005. Model identification for hydrological forecasting under uncertainty. Stoch. Environ. Res. Risk Assess. 19 (6), 378–387. http://dx.doi.org/10.1007/s00477-005-0006-5.

Wallner, M., Haberlandt, U., Dietrich, J., 2012. Evaluation of different calibration strategies for large scale continuous hydrological modelling. Adv. Geosci. 31, 67–74. http://dx.doi.org/10.5194/adgeo-31-67-2012.

Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L.B., Bourne, P.E., et al., 2016. The FAIR guiding principles for scientific data management and stewardship. Sci. Data 3 (1), 1–9. http://dx.doi.org/10.1038/sdata.2016.18.

Yapo, P.O., Gupta, H.V., Sorooshian, S., 1996. Automatic calibration of conceptual rainfall-runoff models: sensitivity to calibration data. J. Hydrol. 181 (1), 23–48. http://dx.doi.org/10.1016/0022-1694(95)02918-4.

Yapo, P.O., Gupta, H.V., Sorooshian, S., 1998. Multi-objective global optimization for hydrologic models. J. Hydrol. 204, 83–97. http://dx.doi.org/10.1016/S0022-1694(97)00107-8.

Yates, D., Sieber, J., Purkey, D., Huber-Lee, A., 2005. WEAP21—A demand-, priority-, and preference-driven water planning model. Water Int. 30, 487–500. http://dx.doi.org/10.1080/02508060508691893.

Zambrano-Bigiarini, M., 2020. Tutorial for using hydroPSO to calibrate the GR4J model. http://dx.doi.org/10.5281/zenodo.3774533, Publisher: Zenodo.

Zambrano-Bigiarini, M., 2024. hydroGOF: Goodness-of-fit functions for comparison of simulated and observed hydrological time series. http://dx.doi.org/10.5281/zenodo.839854, URL: https://cran.r-project.org/package=hydroGOF. R package version 0.6-0. doi:10.5281/zenodo.839854.

Zambrano-Bigiarini, M., Baez-Villanueva, O., 2020. Tutorial for using hydroPSO to calibrate TUWmodel. http://dx.doi.org/10.5281/zenodo.3772176, Publisher: Zenodo.

Zambrano-Bigiarini, M., Rojas, R., 2013. A model-independent particle swarm optimisation software for model calibration. Environ. Model. Softw. 43, 5–25. http://dx.doi.org/10.1016/j.envsoft.2013.01.004.

Zhan, Z.H., Li, J., Cao, J., Zhang, J., Chung, H.S.H., Shi, Y.H., 2013. Multiple populations for multiple objectives: A coevolutionary technique for solving multiobjective optimization problems. IEEE Trans. Cybern. 43 (2), 445–463. http://dx.doi.org/10.1109/TSMCB.2012.2209115.

Zhang, Q., Li, H., 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Trans. Evol. Comput. 11 (6), 712–731. http://dx.doi.org/10.1109/TEVC.2007.892759.

Zhang, Y., Schaap, M.G., 2017. Weighted recalibration of the rosetta pedotransfer model with improved estimates of hydraulic parameter distributions and summary statistics (Rosetta3). J. Hydrol. 547, 39–53. http://dx.doi.org/10.1016/j.jhydrol.2017.01.004.

Zitzler, E., Laumanns, M., Bleuler, S., 2004. A tutorial on evolutionary multiobjective optimization. In: Gandibleux, X., Sevaux, M., Sörensen, K., T'kindt, V. (Eds.), Metaheuristics for Multiobjective Optimisation. Springer, Berlin, Heidelberg, pp. 3–37. http://dx.doi.org/10.1007/978-3-642-17144-4_1.

Zitzler, E., Laumanns, M., Thiele, L., 2001. SPEA2: Improving the strength pareto evolutionary algorithm. Report, ETH Zurich, Computer Engineering and Networks Laboratory, http://dx.doi.org/10.3929/ethz-a-004284029.

Zitzler, E., Thiele, L., 1999. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. IEEE Trans. Evol. Comput. 3, 257–271. http://dx.doi.org/10.1109/4235.797969.