Taylor & Francis
Taylor & Francis Group

Hydrological Sciences Journal

ISSN: 0262-6667 (Print) 2150-3435 (Online) Journal homepage: https://www.tandfonline.com/loi/thsj20

Are modern metaheuristics successful in calibrating simple conceptual rainfall–runoff models?

Adam P. Piotrowski, Maciej J. Napiorkowski, Jaroslaw J. Napiorkowski, Marzena Osuch & Zbigniew W. Kundzewicz

To cite this article: Adam P. Piotrowski, Maciej J. Napiorkowski, Jaroslaw J. Napiorkowski, Marzena Osuch & Zbigniew W. Kundzewicz (2017) Are modern metaheuristics successful in calibrating simple conceptual rainfall–runoff models?, Hydrological Sciences Journal, 62:4, 606-625, DOI: 10.1080/02626667.2016.1234712

To link to this article: https://doi.org/10.1080/02626667.2016.1234712

View supplementary material

Accepted author version posted online: 11 Oct 2016.
Published online: 05 Dec 2016.

Submit your article to this journal

Article views: 459

View related articles

View Crossmark data

Citing articles: 11 View citing articles

HYDROLOGICAL SCIENCES JOURNAL – JOURNAL DES SCIENCES HYDROLOGIQUES, 2017
VOL. 62, NO. 4, 606–625
http://dx.doi.org/10.1080/02626667.2016.1234712

Are modern metaheuristics successful in calibrating simple conceptual rainfall–runoff models?

Adam P. Piotrowski[a], Maciej J. Napiorkowski[b], Jaroslaw J. Napiorkowski[a], Marzena Osuch[a] and Zbigniew W. Kundzewicz[c,d]

[a]Institute of Geophysics, Polish Academy of Sciences, Warsaw, Poland; [b]Environmental Engineering Faculty, Warsaw University of Technology, Warsaw, Poland; [c]Institute for Agricultural and Forest Environment, Polish Academy of Sciences, Poznań, Poland; [d]Potsdam Institute for Climate Impact Research, Potsdam, Germany

ABSTRACT

In recent years sampling approaches have been used more widely than optimization algorithms to find parameters of conceptual rainfall–runoff models, but the difficulty of calibration of such models remains in dispute. The problem of finding a set of optimal parameters for conceptual rainfall–runoff models is interpreted differently in various studies, ranging from simple to relatively complex and difficult. In many papers, it is claimed that novel calibration approaches, so-called metaheuristics, outperform the older ones when applied to this task, but contradictory opinions are also plentiful. The present study aims at calibration of two simple lumped conceptual hydrological models, HBV and GR4J, by means of a large number of metaheuristic algorithms. The tests are performed on four catchments located in regions with relatively similar climatic conditions, but on different continents. The comparison shows that, although parameters found may somehow differ, the performance criteria achieved with simple lumped models calibrated by various metaheuristics are very similar and differences are insignificant from the hydrological point of view. However, occasionally some algorithms find slightly better solutions than those found by the vast majority of methods. This means that the problem of calibration of simple lumped HBV or GR4J models may be deceptive from the optimization perspective, as the vast majority of algorithms that follow a common evolutionary principle of survival of the fittest lead to sub-optimal solutions.

ARTICLE HISTORY
Received 21 January 2016
Accepted 14 July 2016

EDITOR
D. Koutsoyiannis

ASSOCIATE EDITOR
C. Perrin

KEYWORDS
conceptual rainfall–runoff models; HBV; GR4J; evolutionary algorithms; differential evolution; particle swarm optimization

1 Introduction

The problem of calibration of conceptual rainfall–runoff models has been discussed in hydrology for at least 50 years. Initially, manual calibration prevailed, with the objective of fitting the model parameters to minimize the simulation or prediction error for the observed data (Pechlivanidis *et al*. 2011). Although such manual calibration is sometimes still in use (Kim *et al*. 2007, Vansteenkiste *et al*. 2014, Willems 2014), the so-called automatic optimization procedures have gained popularity in line with the dramatic increase in computational resources. A number of automatic calibration procedures were proposed during 1960s to 1980s, many of them (e.g. Rosenbrock 1960, Nelder and Mead 1965, Kirkpatrick *et al*. 1983) did not require the objective function to be differentiable, as the gradient optimization methods did. This allowed their application to a wide range of conceptual rainfall–runoff models. However, problems with finding a good and unique set of parameters were soon reported (Ibbitt and O'Donnell 1971, Johnston and Pilgrim 1976, Pickup 1977) and became widely acknowledged following the work of Duan *et al*. (1992). In recent decades, the attention of hydrologists has shifted to uncertainty of data, models and their parameters, rather than being focused on minimization of the prediction error alone. As a result, instead of optimization algorithms aimed at a search for the (possibly global) optimum, sampling methods, especially Markov chain Monte Carlo (MCMC) ones, such as the Metropolis-Hastings algorithm (Metropolis *et al*. 1953, Hastings 1970), SCEM-UA (Vrugt *et al*. 2003) or DREAM (Vrugt *et al*. 2009a, 2012, Vrugt 2016), have become widely used. Somehow in parallel the automatic multi-objective optimization of rainfall–runoff models has become popular. One may mention the works by Yapo *et al*. (1998), Madsen (2000), Madsen *et al*. (2002) and Tang *et al*. (2006). For detailed reviews of the more recent studies the reader is referred to Efstratiadis and Koutsoyiannis (2010) and Reed *et al*. (2013).

Although among the hydrological community popularity has shifted to multi-objective optimization and MCMC sampling, optimization methods that search for the global optimum are still used and compared in various studies aimed at calibration of rainfall–runoff models (Blasone *et al*. 2007, Goswami and O'Connor 2007, Kim *et al*. 2007, Tolson and Shoemaker 2007, Wang *et al*. 2010, Romanowicz *et al*. 2013, Xu *et al*. 2013, Willems *et al*. 2014, Tigkas *et al*. 2015). The ability to find near-optimum solutions of real-world problems in a reasonable length of time was considered to be one of the main challenges in Maier *et al*. (2014). Unfortunately, in most

CONTACT Adam P. Piotrowski ✉ adampp@igf.edu.pl
The supplementary data for this article can be accessed here.

papers the number of optimization methods compared is very small (usually between two and six) and the conclusions differ significantly. In some studies the superiority of some approaches over others is claimed, while in other studies very similar performance is shown for all tested algorithms. For example, Wang *et al.* (2010) found that shuffled complex evolution (SCE) and two different variants of genetic algorithms (GA) perform very similarly when applied to calibration of a distributed rainfall–runoff model for a small catchment located in Taiwan. Goswami and O'Connor (2007) found very slight superiority of simulated annealing over particle swarm optimization (PSO), GA, shuffled complex evolution–University of Arizona (SCE-UA), Nelder-Mead simplex (NMA) and Rosenbrock algorithm (RA) in calibration of a soil moisture accounting and routing model for rainfall–runoff simulation at two catchments of very different size, located in Ireland and China. Minor differences between various tested optimization methods were also reported by Gan and Biftu (1996), Blasone *et al.* (2007) and Kavetski and Clark (2010). On the other hand, superiority of some optimization methods over others was claimed, for example, in Tolson and Shoemaker (2007), Xu *et al.* (2013), Tigkas *et al.* (2015) and a few older studies (Cooper *et al.* 1997, Kuczera 1997, Franchini *et al.* 1998). Note that similarly contradictory conclusions are given by different authors when comparing MCMC approaches (Laloy and Vrugt 2012, Chu *et al.* 2014, Vrugt and Laloy 2014). There may be several reasons for such differences. For example, recently it was shown that improper choice of control parameters of the optimization method itself affects the quality of solutions found by a particular algorithm in the case of rainfall–runoff modelling (Qi *et al.* 2016). Such contradictory results may have been achieved because particular studies used various implementations of different models tested on various rivers with various amounts and quality of available data, and so on. This may be of great importance, as discussed by Kavetski and Clark (2011). The discussion of such a litany of reasons is beyond the scope of this paper. Nonetheless, irrespective of the reasons, this variety of opinions shows the need for a wider comparison of a larger number of optimization algorithms. Indeed, over the past two decades one could observe an influx of such methods, especially so-called metaheuristics, i.e. heuristics that are applicable to versatile kinds of problems (Glover 1986).

In the present paper, over 20 optimization algorithms are tested on calibration of two simple lumped conceptual rainfall–runoff models—HBV (Bergström 1976, Lindström 1997) and GR4J (Perrin *et al.* 2003)—applied to daily runoff forecasting at four catchments located in temperate climate zones. As shown in Vansteenkiste *et al.* (2014), simple lumped conceptual models are still a reasonable alternative to distributed models; tests on higher-parameterized, distributed models are left for the future. Between the two chosen models, GR4J was initially proposed without any snow routine (Perrin *et al.* 2003), which allowed the number of parameters to be kept as low as four, but led to unsatisfactory performance in some applications (Pokhrel *et al.* 2014). Only recently (Valery *et al.* 2014a, 2014b) have snow modules been added to the GR4J model. As snow accumulation and melting play an important role in three out of the four considered catchments, in this paper GR4J is also implemented with a very simple snow routine, which extends the number of its parameters to seven. However, the original name GR4J is retained throughout the paper. The variant of the second model tested in this study (HBV) requires 13 parameters to be optimized. As both models are frequently used in forecasting mode, their performance is improved here by using classical linear regression with exogenous inputs as the data assimilation procedure for error correction, as suggested by Refsgaard (1997) and Madsen and Skotner (2005). In this study, the updating procedure is performed for the final solutions only, after termination of the calibration procedures (Refsgaard 1997).

2 Conceptual rainfall–runoff models

Models developed to characterize the rainfall–runoff process in catchments are usually classified as physically-based, conceptual or empirical. This paper considers two lumped conceptual models, HBV and GR4J, that involve a configuration of interconnected stores with mathematical transfer functions used to direct the movement of water between stores and into the stream. In both models, elevation correction is not taken into account.

2.1 HBV model

The HBV model, introduced by Bergström and Forsman (1973), is a standard tool for runoff simulations and flood forecasting in Scandinavia, and has been applied in over 50 countries worldwide. A large majority of these applications make use of various modified versions of the original HBV model (Bergström 1995, Bergström and Lindström 2015); therefore, a detailed description of HBV components, including subroutines for snow accumulation and melting, soil moisture accounting and response generation of runoff, for the version adopted in this paper is given in the Appendix. The 13 parameters to be calibrated are denoted by capital letters.

The input variables to the HBV model are daily precipitation totals (Precip), mean air temperature (Temp) and estimated potential evapotranspiration (Pet) calculated by the Thornthwaite method (Thornthwaite 1948). The model has five state variables representing storage of snow pack (ssp), snowmelt water (ssw), soil moisture (ssm), fast runoff (sfr) and base flow (sbf).

2.2 GR4J model

The GR4J conceptual model was introduced by Perrin (2000) as an extension of the GR3J approach proposed by Edijatno *et al.* (1999). The detailed mathematical description of the GR4J model may be found in Perrin *et al.* (2003), so readers are referred to that paper. The model performs well even for data collected within short time intervals (Ficchi *et al.* 2016) or almost ungauged catchments (Rojas-Serna *et al.* in press). Since our study is concerned with catchments located in temperate climatic conditions, the original model is extended by adding a snow module (as suggested in Valery *et al.* 2014a, 2014b). The snow module used in this study is a simplified version of that used in the HBV model (see Appendix), but with water holding capacity of snow set to zero. The inputs to the GR4J model include daily precipitation (Precip), mean air temperature (Temp) and potential evapotranspiration (Pet). Although this extended version of GR4J has seven parameters instead of four, i.e. three parameters in the snow routine (TT, TTI, CFMAX) and four original parameters representing maximum capacity of production store (x1, mm), groundwater exchange coefficient (x2, mm), one-day-ahead maximum capacity of routing store (x3, mm) and time base of unit hydrograph UH1 (x4, days), the original name GR4J is retained in this paper.

2.3 Updating procedure

The forecasting performance of both conceptual models may be significantly improved by means of data assimilation procedures (Refsgaard 1997, Madsen *et al.* 2000, Madsen and Skotner 2005). In the present paper, after termination of the calibration procedure, the results from the HBV and GR4J models are updated by means of linear regression with exogenous inputs, as in Piotrowski and Napiorkowski (2012). The past forecasts from "classical" HBV and GR4J models are added as exogenous inputs to the linear regression error model $\varepsilon^p_{t+1} = L(\varepsilon^p_t, \varepsilon^p_{t-1}, \ldots, \varepsilon^p_{t-\delta+1}, y^{Model}_{t+1}, y^{Model}_t, \ldots, y^{Model}_{t-\delta})$ where *Model* denotes HBV or GR4J and $\varepsilon^p_t = y_t - y^{Model}_t$ is the prediction error. The forecast flow for both considered models is calculated as $y^p_{t+1} = y^{Model}_{t+1} + \varepsilon^p_{t+1}$. The number of required previous observations δ used in HBV and GR4J has been set to three.

3 Study catchments and data

The present study is based on data collected from four catchments that, although located in different countries and clearly differing in topography, have roughly similar climatic conditions and size. Although testing a large number of algorithms on a larger database, such as the one available within the MOPEX project (Duan *et al.* 2006), would strengthen the results (Gupta *et al.* 2014), some trade-off between the number of catchments, the number of algorithms and runs performed by each algorithm has to be coined. The main information on the four chosen catchments is given in Table 1 and briefly discussed below.

In the Annapolis River catchment (Nova Scotia, Canada), snowfall occurs from November to April and peak rainfalls are observed between September and November. A detailed, even if not the most recent, description of the catchment may be found in Trescott (1968). The daily runoff data for the gauge station situated in Wilmot settlement are available from the Water Survey of Canada and Canada's National Climate Data and Information Archive. The daily air temperature and precipitation data used in this study were measured at a single site, the meteorological station located at the Greenwood Airfield, 9 km to the east of Wilmot.

The Biala Tarnowska catchment shares climatic conditions during winter months with the similar Annapolis catchment, but the highest rainfalls are observed in summer. One lead-day runoff forecasting in Koszyce Wielkie is based on air temperature, precipitation and runoff measurements. Runoff measurements were recorded at Koszyce Wielkie village, while precipitation was measured at 12 locations within, or close to, the catchment. The catchment average daily rainfall time series was created by means of the Thiessen polygons method.

The Allier River enjoys a mild oceanic climate. Rainfall is noted throughout the year, but although snow is not uncommon at higher elevations in winter, this is the only studied catchment for which it plays a very limited role. Highest runoff is observed in late spring and in autumn. A detailed description of the catchment may be found in Thirel *et al.* (2015). Daily runoff forecasts performed in the present study for the Allier catchment are based on river discharge, air temperature, potential evapotranspiration and precipitation data available from www.hydro.eaufrance.fr (Vidal *et al.* 2010).

In the Nysa Klodzka catchment, snow plays an important role in the flow regime during winter and spring and, due to the specific orographic and climatic conditions of the area, flooding is frequent. Precipitation is available for five locations within the catchment and the time series of the average precipitation has been formed by means of the Thiessen polygons method.

For each catchment, the first 365 days of the training sets were used as a warm-up period and did not have an impact on the objective function.

4 Metaheuristics used and comparison criteria

Although mathematical programming and direct search methods (Kolda *et al.* 2003) have been known for many years, today the popularity of so-called metaheuristics is also soaring in hydrology (Maier *et al.* 2014). Metaheuristics usually draws from biological inspiration. A large number of such methods (a review may be found in Boussaid *et al.* 2013), including genetic algorithms (GA) (Holland 1975), evolution strategies (ES) (Bäck and Schwefel 1993), genetic programming (GP) (Koza 1992), differential evolution (DE) (Storn and Price 1995), particle swarm optimization (PSO) (Eberhart and Kennedy 1995) or ant colony optimization (Dorigo *et al.* 1996), are well established in the literature and have turned out to be successful in many real-world applications in different fields of science. However, in recent

Table 1. Main catchment data.

	Catchment	Location	Size (km^2)/Closing station	Orography/Highest altitude	Calibration period	Validation period
1	Annapolis	Nova Scotia, Canada	546/Wilmot (NS)	Hilly/275	01/01/1980–31/12/1999	01/01/2000–31/12/2009
2	Biala Tarnowska	Poland	956/Koszyce Wielkie (near Tarnow)	Mountainous/997	01/01/1971–31/12/1989	01/01/1990–31/10/2000
3	Allier	France	2269/Veille-Brioude	Mountainous/1565	01/08/1978–31/07/1998	01/08/1999–31/07/2008
4	Nysa Klodzka	Poland	1061/Klodzko	Mountainous/1425	01/01/1971–31/12/1995	01/01/1996–31/10/2010

years, many approaches with very "exotic" motivations and names have been proposed (see for example a list in Xing and Gao 2014, Biswas *et al.* 2013), resulting in critical papers showing that at least some such methods mimic the older ones, lack any true novelty except for a spectacular name, or are developed without scientific rigour (Weyland 2010, Crepinsek *et al.* 2012, Piotrowski *et al.* 2014, Sorensen 2015).

Due to the observed abundance of emerging metaheuristics, a general comparison among them is, in fact, infeasible. As a result, some, usually subjective, initial selection of methods is needed. Due to the reasons mentioned, the algorithms with "novel" inspirations are not considered in this paper, as the choice among variants of widely accepted methods is wide enough. A list of the 26 algorithms tested in this study, with brief descriptions, is presented in Table 2.

From Table 2 one can see that most attention is drawn to variants of DE algorithms that have already been used in various hydrological applications (e.g. Kisi 2004, Zheng *et al.* 2011, Piotrowski and Napiorkowski 2012, Dokou and Karatzas 2013, Elci and Ayvaz 2014, Ren *et al.* 2016, Piotrowski *et al.* in press) and become the basis of MCMC approaches within the DREAM family of methods (ter Braak and Vrugt 2008, Vrugt *et al.* 2009a, Vrugt 2016). In-depth discussion of DE algorithms may be found in review papers by Neri and Tirronen (2010), Das and Suganthan (2011) and Das *et al.* (2016). The crucial point in application of DE algorithms is the proper choice of population size (Piotrowski in press). In this study, the population size of the majority of applied DE variants is set to 5D, where D is the dimensionality of the problem. However, some DE variants require different population sizes: such cases are clearly described in the above list of applied algorithms. Apart from DE algorithms, seven approaches from among other kinds of evolutionary algorithms, swarm intelligence and direct search methods are also tested in this study.

The population size of non-DE algorithms depends on the specific characteristics of each method. The maximum number of function evaluations is set to 30 000 for both HBV and GR4J models. To get a large enough sample to justify conclusions from the tests performed, each algorithm is run independently 30 times for every model and catchment, starting from different, randomly-generated, initial solutions.

In addition to classical optimizers, for comparison purpose, one MCMC method is used, namely DE-based DREAM_ZS (Laloy and Vrugt 2012, Vrugt 2016) with parameter settings suggested in Vrugt *et al.* (2008) for HYMOD model calibration. In the case of DREAM_ZS, in this study only the performance of the best solution (in terms of calibration criterion defined in Equation (1) given below, determined for the training period) in each run is used for comparison with solutions determined by optimization metaheuristics. Note that 30 runs are performed by DREAM_ZS, as in the case of standard optimization algorithms. Comparison of classical optimizers with the MCMC method was motivated by Laloy and Vrugt (2012), who, using a similar approach, tested DREAM_ZS against two optimization algorithms, namely SP-UCI (a variant discussed in Chu *et al.* 2010) and PEST (Doherty 2009) and found DREAM_ZS to outperform SP-UCI and be comparable with PEST. Hence we wish to verify the applicability of DREAM_ZS for calibration purposes against a large number of modern optimization algorithms.

The techniques for handling bounds may have some impact on the results achieved. In the case of NMA, RA, PMS and DE-based approaches (including DCMA, but not CLPSO-DEGL) the classical rebounding (or reflection) approach is used (as in Helwig *et al.* 2013, Piotrowski 2013). For the hybrid CLPSO-DEGL algorithm, two different bound-handling approaches are implemented: for CLPSO the one suggested in Liang *et al.* (2006), while for DEGL, the rebounding method. In the case of other metaheuristics, techniques suggested in the source papers are implemented.

For all metaheuristics except AMALGAM, which uses its own initialization procedure (see Vrugt *et al.* 2009b), and DREAM_ZS, which uses latin hypercube sampling, the initial values of HBV and GR4J parameters are drawn randomly from the uniform distributions within the defined upper and lower parameter bounds (Tables 3 and 4). The parameter ranges are based on experience and literature review (e.g. Bergström 1976, Perrin *et al.* 2003) and are kept fixed for all four catchments in the GR4J model. However, when the HBV model was used on the Allier catchment, slightly wider parameter ranges were required than those used for the other three catchments.

All metaheuristics considered in this study are used for calibration of both HBV and GR4J models for 1 lead-day runoff forecasting in the Annapolis, Biala Tarnowska, Allier and Nysa Klodzka catchments. Denoting the number of data in each set (training or validation, note that in the case of training data the 365-day warm-up period is excluded) by N, the lead time (equal to 1 day) by LT, and the forecast and observed runoff as ${y_n}^P$ and y_n, respectively, the mean square error (MSE) is defined as:

$$\mathrm{MSE} = \frac{1}{N}\sum_{n=1}^{N}\left(y_n^P - y_n\right)^2 \tag{1}$$

MSE is also used as the objective function (to be minimized) during model calibration. The quality of the results is also checked using the persistence index (PI) (Kitanidis and Bras 1980):

$$\mathrm{PI} = 1 - \frac{\sum\limits_{n=\mathrm{LT}+1}^{N}\left(y_n^P - y_n\right)^2}{\sum\limits_{n=\mathrm{LT}+1}^{N}\left(y_n - y_{n-\mathrm{LT}}\right)^2} \tag{2}$$

A value of PI equal to 1 means a perfect fit, while negative values suggest that it is better to accept the last measured flow as a forecast (i.e. the so-called conservative forecast) rather than using the tested model.

5 Results

The statistics obtained by the HBV and GR4J models calibrated by means of each of 26 metaheuristics are, due to space restrictions, given in the Supplementary material (Tables S1–S8). They include a 30-run averaged MSE and PI for training and validation data, accompanied by appropriate standard deviations, the lowest MSE found during 30 runs according to training sets and the lowest MSE found during 30 runs according to validation sets

Table 2. Optimization algorithms used. DE: differential evolution; PSO: particle swarm optimization; NP: population size.

	Abbreviation	Full name	Reference	Comments
1	NMA	Nelder-Mead algorithm with re-initialization	Nelder and Mead 1965; Lagarias *et al.* 1998	The classical NMA with added possibility of re-initialization of all points but the best one. Re-initialized solutions are generated randomly from uniform distribution within parameter bounds. By definition NP equals D + 1. Algorithm is re-initialized when either the maximum difference between coordinates of the best and the worst point is less than 10^{-4}, or the difference in fitness between the best and the worst point is less than 10^{-4}.
2	RA	Rosenbrocck's algorithm with re-initialization	Rosenbrock 1960	This is a non-"population based" algorithm (one may say that population size is equal to 1) and the only local search procedure tested. It is used in this study with re-initialization. After every 100D (where D is the dimensionality of the problem) function calls it is verified if during the last 100D function calls the solution was improved by more than 10^{-4}. If not, the location of the RA point is randomly re-initialized, the value of the step length is re-set to 0.1 (see Rosenbrock 1960) and the coordinates are re-set to the initial system. The following parameter settings are used: $\alpha = 3$, $\beta = -0.5$, initial estep = 0.1 (variable during search).
3	SADE	Self-adaptive DE	Qin *et al.* 2009	Probably the most popular adaptive DE variant.
4	RB-SADE	Ranking-based SADE	Gong and Cai 2013	RB-SADE is a modified SADE variant, in which better vectors are more frequently used as a base and terminal points in DE mutation schemes.
5	DEGL	DE with global and local neighbourhood mutation operators	Das *et al.* 2009	DEGL variant with self-adaptive weight values is used, as suggested in Das *et al.* 2009.
6	AM-DEGL	Adaptive memetic DEGL	Piotrowski 2013	Adaptive memetic DE variant, based on DEGL, SADE and NMA.
7	CLPSO	Comprehensive learning PSO	Liang *et al.* 2006	State-of-the-art PSO variant. The velocity of each particle is restricted within 20% of every parameter range, and initialized within this range. The population size is set to 30 particles in this paper.
8	CLPSO-DEGL	Hybrid CLPSO and DEGL approach	Epitropakis *et al.* 2012	Algorithm merges the benefits of both classical PSO and DE variants. More specifically, at each generation the algorithm initially performs the search by means of CLPSO, then the best positions of each particle form the DE population and such a population is managed by means of DEGL. The CLPSO and DEGL moves are implemented alternately. The specific control parameter settings of both CLPSO and DEGL algorithms are adopted (but population size is set to 30, following CLPSO).
9	DE-SG	DE with separated groups	Piotrowski *et al.* 2012a, 2012b	Distributed DE variant, which is an updated version of grouping DE (Piotrowski and Napiorkowski 2010). The population size is set to the closest number lower than or equal to 5D that, when divided by 10, produces a quotient without remainder. As in previous papers aimed at artificial neural network training (Piotrowski and Napiorkowski 2012, Piotrowski *et al.* in press), to speed up convergence of DE-SG, the parameter named pre-defined number of iterations (PNI) is reduced to 10 and the parameter that defines migration probability (MigProb) is set to 1/PNI;
10	SspDE	Self-adaptive DE	Pan *et al.* 2011	Self adaptive DE variant.
11	JADE	JADE	Zhang and Sanderson 2009	Variant with archive is used, as suggested by Zhang and Sanderson 2009.
12	AdapSS-JADE	JADE with adaptive strategy selection	Gong *et al.* 2011	The variant with normalized average reward is used, as the best among four proposed in Gong *et al.* 2011.
13	DECLS	DE with chaotic local search	Jia *et al.* 2011	Memetic DE variant based on chaotic local search.
14	DEahcSPX	DE with adaptive crossover-based local search	Noman and Iba 2008	One of the earliest memetic DE algorithms.
15	CoDE	Composite DE	Wang *et al.* 2011	Unusual DE variant that creates three offspring for each parent.
16	EPSDE	DE with ensemble of mutation strategies and control parameters	Mallipeddi *et al.* 2011	DE variant based on a novel concept of self-adaptation.
17	SFMDE	Super-fit Memetic DE	Caponio *et al.* 2009	This DE variant hybridizes DE, PSO, RA and NMA algorithms.
18	CDE	Clustering-based DE	Cai *et al.* 2011	Probably the first DE approach based on the concept of clustering.
19	IMDE	DE with intersect mutation operator	Zhou *et al.* 2013	IMDE introduces novel DE mutation and crossover schemes. The variant defined as "1st process" is used.
20	MDE_pBX	Modified DE with p-best crossover	Islam *et al.* 2012	MDE_pBX introduces another novel DE crossover and mutation operators.
21	DCMA	Differential covariance matrix adaptation evolution strategy	Ghosh *et al.* 2012	A hybrid of DE and CMA-ES (Hansen and Ostermeier 1996). As suggested by Ghosh *et al.* (2012), DCMA is applied with population size equal to 50 (independent of dimensionality of the problem). The initial value of the control parameter σ is set to 50% of the maximum range among all model parameters.
22	PMS	Parallel memetic structures	Caraffini *et al.* 2013	A kind of memetic computing approach that is a modified version of a simple non-population-based heuristic algorithm designed following the philosophy of Ockham's razor.
23	jDElscop	Self-adaptive differential evolution algorithm using population size reduction and three strategies	Brest and Maucec 2011	Self-adaptive DE variant with variable population size, which is periodically diminished during the search. jDElscop starts from NP = 10D, and finishes with NP = ceil(1.25D).
24	AMALGAM	A multi-algorithm genetically adaptive method for single objective optimization	Vrugt *et al.* 2009b	AMALGAM variant that merges CMA-ES, GA and PSO, as suggested in Vrugt et al 2009. AMALGAM makes a number of sub-runs within the time budget. The population size starts from 15 in the first sub-run and in each consecutive sub-run, within the time budget, is increased by a factor of 2 (but not to a value larger than 480).

(*Continued*)

Table 2. (Continued).

	Abbreviation	Full name	Reference	Comments
25	SP-UCI	Shuffled complex evolution with principal components analysis – University of California at Irvine	Chu *et al.* 2011	A modified version of shuffled complex evolution (Duan *et al.* 1992) that uses a number of NMA simplexes to move through the search space. SP-UCI with four simplexes is used in this study.
26	DREAM_TS	Differential evolution adaptive metropolis	Laloy and Vrugt 2012	The only MCMC method tested. A modified version of DREAM (Vrugt *et al.* 2008) with only three chains (see ter Braak and Vrugt 2008 and Laloy and Vrugt 2012) and density function defined as the sum of square errors (Vrugt *et al.* 2008). Note that only the performance of the best solution found by DREAM in each run is used for comparison with classical optimizers.

Table 3. Parameter ranges of the HBV model. L and U refer to lower and upper bounds, respectively.

	FC	BETA	LP	ALPHA	KF	KS	PERC	CFLUX	TT	TTI	CFMAX	CFR	WHC
	(mm)	(–)	(–)	(–)	(1/d)	(1/d)	(mm/d)	(mm/d)	(°C)	(°C)	(mm/°C d)	(–)	(–)
Annapolis, Biala Tarnowska, Nysa Klodzka catchments													
L	10.0	0.3	0.1	−0.5	0.01	0.01	0.01	0.01	−3.0	0.01	0.5	0.001	0.001
U	500.0	4.0	0.95	2.0	0.5	0.08	10.0	10.0	3.0	8.0	5.0	0.300	0.300
Allier catchment													
L	10.0	0.01	0.1	−0.5	0.0001	0.01	0.01	0.0001	−3.0	0.01	0.5	0.0001	0.0001
U	500.0	4.0	0.999	2.0	0.5	0.3	10.0	10.0	3.0	8.0	5.0	0.300	0.300

Table 4. Parameter ranges of the GR4J model (the same for each catchment). L and U refer to lower and upper bounds, respectively.

	x1	x2	x3	x4	TT	TTI	CFMAX
	(mm)	(mm)	(mm)	(d)	(°C)	(°C)	(mm/°C d)
L	0.1	−7.0	0.1	0.5	−3.0	0.01	0.5
U	1500.0	5.0	500.0	5.0	3.0	8.0	5.0

(note that the best solution for training data is not necessarily the best for validation). Selected results, which include only averaged MSE and PI values for each river, are presented here as Tables 5 and 6. As for other optimizers, only the best solution found by DREAM_ZS during each run (there are 30 runs, hence 30 best solutions) is used to calculate the mentioned statistics.

5.1 Comparison of GR4J and HBV performance

One may note that GR4J outperforms HBV for data collected at the Annapolis River catchment according to the validation but not for the training set. On the contrary, for the Biala Tarnowska River catchment, GR4J is better than HBV on training, but not on validation data. For the Allier and Nysa Klodzka catchments, HBV outperforms GR4J on both training and validation sets. Such differences in results are not

Table 5. Mean results achieved by every algorithm used to calibrate the HBV and GR4J models for rivers Annapolis and Biala Tarnowska. The lowest mean square error results are in bold. MSE: mean square error ($(\mathrm{m}^3/\mathrm{s})^2$); PI: persistence index; t: training; v: validation.

	Annapolis River – HBV				Annapolis River – GR4J				Biala Tarnowska River – HBV				Biala Tarnowska River – GR4J			
	MSE(t)	MSE(v)	PI(t)	PI(v)	MSE(t)	MSE(v)	PI(t)	PI(v)	MSE(t)	MSE(v)	PI(t)	PI(v)	MSE(t)	MSE(v)	PI(t)	PI(v)
DEGL	11.275	16.534	0.574	0.650	11.379	12.889	0.570	0.727	41.264	37.388	0.815	0.777	40.227	42.458	0.819	0.747
JADE	11.284	16.546	0.573	0.650	11.380	12.896	0.570	0.727	41.036	36.947	0.816	0.780	40.167	42.482	0.819	0.747
AdapSS-JADE	11.284	16.544	0.573	0.650	11.380	12.895	0.570	0.727	40.981	36.950	0.816	0.780	40.167	42.482	0.819	0.747
AM-DEGL	11.281	16.538	0.573	0.650	11.380	12.895	0.570	0.727	40.935	36.855	0.816	0.780	40.167	42.482	0.819	0.747
CDE	11.274	16.521	0.574	0.651	11.380	12.893	0.570	0.727	41.153	37.156	0.815	0.778	40.182	42.476	0.819	0.747
CoDE	11.285	16.788	0.573	0.645	11.381	12.903	0.569	0.727	42.062	38.104	0.811	0.773	40.167	42.482	0.819	0.747
DCMA	11.283	16.541	0.573	0.650	11.382	12.903	0.569	0.727	41.136	37.283	0.815	0.778	40.167	42.482	0.819	0.747
DEahcSPX	11.294	16.790	0.573	0.645	11.381	12.902	0.569	0.727	42.358	38.240	0.810	0.772	40.167	42.482	0.819	0.747
DECLS	11.275	16.576	0.574	0.650	11.381	12.898	0.570	0.727	41.462	38.133	0.814	0.773	40.167	42.482	0.819	0.747
DESG	11.280	16.564	0.573	0.650	11.381	12.902	0.569	0.727	41.441	37.855	0.814	0.774	40.167	42.482	0.819	0.747
EPSDE	11.277	16.526	0.573	0.651	11.380	12.896	0.570	0.727	40.961	36.786	0.816	0.781	40.167	42.482	0.819	0.747
IMDE	11.284	16.545	0.573	0.650	11.380	12.893	0.570	0.727	41.179	37.350	0.815	0.777	40.167	42.482	0.819	0.747
jDElscop	11.283	16.542	0.573	0.650	11.381	12.898	0.570	0.727	40.937	36.780	0.816	0.781	40.167	42.482	0.819	0.747
MDE_pBX	11.267	**16.500**	0.574	0.651	11.379	12.880	0.570	0.728	40.916	36.736	0.816	0.781	40.227	42.458	0.819	0.747
RB-SADE	11.270	16.505	0.574	0.651	11.379	12.890	0.570	0.727	40.960	36.783	0.816	0.781	40.197	42.470	0.819	0.747
SADE	11.266	**16.500**	0.574	0.651	11.380	12.893	0.570	0.727	41.025	36.924	0.816	0.780	40.167	42.482	0.819	0.747
SFMDE	11.373	16.974	0.570	0.641	11.646	13.719	0.559	0.710	48.069	44.339	0.784	0.736	40.202	42.471	0.819	0.747
SspDE	**11.261**	16.515	0.574	0.651	11.380	12.892	0.570	0.727	41.081	37.065	0.815	0.779	40.167	42.482	0.819	0.747
SP-UCI	11.280	16.533	0.573	0.650	11.380	12.891	0.570	0.727	40.861	**36.680**	0.816	0.781	40.167	42.482	0.819	0.747
NMA	11.319	16.580	0.572	0.649	11.401	12.881	0.569	0.728	41.708	38.140	0.813	0.773	**38.654**	42.669	0.818	0.745
RA	11.414	17.763	0.568	0.624	11.377	12.877	0.570	0.728	44.041	40.664	0.802	0.757	40.174	42.507	0.819	0.746
PMS	11.266	17.100	0.574	0.638	**11.298**	**12.629**	0.573	0.733	42.657	39.308	0.808	0.766	40.486	42.599	0.818	0.746
CLPSO	11.311	16.843	0.572	0.644	11.374	12.864	0.570	0.728	**40.191**	42.456	0.819	0.747	40.191	**42.456**	0.819	0.747
CLPSO-DEGL	11.353	17.395	0.571	0.632	11.375	12.867	0.570	0.728	43.256	39.507	0.806	0.764	40.193	42.494	0.819	0.747
AMALGAM	11.297	16.635	0.573	0.648	11.374	12.874	0.570	0.728	41.371	37.743	0.814	0.775	40.183	42.483	0.819	0.747
DREAM_ZS	11.271	16.561	0.574	0.650	11.382	12.894	0.569	0.727	41.099	37.380	0.815	0.777	40.165	42.518	0.819	0.746

Table 6. Mean results achieved by every algorithm used to calibrate the HBV and GR4J models for rivers Allier and Nysa Klodzka. The lowest mean square error results are in bold. MSE: mean square error ($(m^3/s)^2$); PI: persistence index; t: training; v: validation.

	Allier River – HBV				Allier River – GR4J				Nysa Klodzka River – HBV				Nysa Klodzka River – GR4J			
	MSE(t)	MSE(v)	PI(t)	PI(v)	MSE(t)	MSE(v)	PI(t)	PI(v)	MSE(t)	MSE(v)	PI(t)	PI(v)	MSE(t)	MSE(v)	PI(t)	PI(v)
DEGL	101.705	57.173	0.718	0.723	108.230	62.661	0.700	0.696	21.426	61.372	0.708	0.728	23.947	65.526	0.673	0.709
JADE	101.469	56.856	0.719	0.724	108.235	62.665	0.700	0.696	21.460	59.936	0.707	0.734	23.824	64.835	0.675	0.712
AdapSS-JADE	101.565	56.811	0.718	0.724	108.235	62.665	0.700	0.696	21.462	59.992	0.707	0.734	23.823	64.817	0.675	0.712
AM-DEGL	101.637	56.973	0.718	0.724	108.235	62.665	0.700	0.696	21.453	60.221	0.707	0.733	23.823	64.833	0.675	0.712
CDE	101.423	56.997	0.719	0.723	108.235	62.665	0.700	0.696	21.468	60.002	0.707	0.734	23.831	64.868	0.675	0.712
CoDE	104.613	58.219	0.710	0.717	108.235	62.665	0.700	0.696	21.488	62.824	0.707	0.721	23.823	64.834	0.675	0.712
DCMA	101.804	**56.495**	0.718	0.726	108.235	62.665	0.700	0.696	21.467	59.747	0.707	0.735	23.824	64.834	0.675	0.712
DEahcSPX	104.973	59.121	0.709	0.713	108.235	62.665	0.700	0.696	21.463	60.767	0.707	0.730	23.824	64.834	0.675	0.712
DECLS	102.993	57.450	0.714	0.721	108.235	62.665	0.700	0.696	21.476	59.891	0.707	0.734	23.822	64.801	0.675	0.712
DESG	101.750	57.441	0.718	0.721	108.235	62.665	0.700	0.696	21.474	59.948	0.707	0.734	23.823	64.833	0.675	0.712
EPSDE	**101.350**	56.887	0.719	0.724	108.235	62.665	0.700	0.696	21.468	59.727	0.707	0.735	23.824	64.835	0.675	0.712
IMDE	101.500	56.781	0.719	0.724	108.235	62.665	0.700	0.696	21.466	59.982	0.707	0.734	23.822	64.827	0.675	0.712
jDElscop	101.464	56.867	0.719	0.724	108.235	62.665	0.700	0.696	21.468	59.908	0.707	0.734	23.824	64.834	0.675	0.712
MDE_pBX	101.426	56.875	0.719	0.724	108.191	62.630	0.700	0.696	21.449	59.909	0.707	0.734	23.825	64.841	0.675	0.712
RB-SADE	101.586	57.137	0.718	0.723	108.230	62.661	0.700	0.696	21.444	60.555	0.707	0.731	23.829	64.859	0.675	0.712
SADE	101.469	57.077	0.719	0.723	108.235	62.665	0.700	0.696	21.458	60.003	0.707	0.734	23.824	64.835	0.675	0.712
SFMDE	111.376	61.840	0.691	0.700	108.238	62.930	0.700	0.695	21.616	62.692	0.705	0.722	23.860	64.935	0.674	0.712
SspDE	101.818	57.525	0.718	0.721	108.235	62.665	0.700	0.696	21.453	60.162	0.707	0.733	23.820	64.820	0.675	0.712
SP-UCI	101.859	57.094	0.718	0.723	108.235	62.665	0.700	0.696	**21.380**	62.095	0.708	0.724	23.813	64.777	0.675	0.713
NMA	102.896	57.855	0.715	0.719	109.002	63.488	0.698	0.692	21.399	63.347	0.708	0.719	23.898	64.930	0.674	0.712
RA	106.715	58.848	0.704	0.714	108.229	62.671	0.700	0.696	21.621	62.929	0.705	0.721	23.821	**64.637**	0.675	0.713
PMS	106.696	58.594	0.704	0.716	**108.177**	**62.538**	0.700	0.697	21.570	70.329	0.706	0.688	24.289	67.024	0.669	0.703
CLPSO	104.160	59.417	0.711	0.712	108.179	62.837	0.700	0.695	21.511	**57.468**	0.706	0.745	23.858	65.097	0.674	0.711
CLPSO-DEGL	109.483	60.263	0.696	0.708	108.178	63.284	0.700	0.693	22.448	64.063	0.694	0.716	23.907	65.076	0.674	0.711
AMALGAM	101.510	56.937	0.719	0.724	108.235	62.658	0.700	0.696	21.540	59.496	0.706	0.736	23.871	65.005	0.674	0.712
DREAM_ZS	102.081	57.483	0.717	0.721	108.239	62.795	0.700	0.695	21.485	60.903	0.707	0.730	**23.809**	64.771	0.675	0.713

surprising, as over decade ago Perrin *et al.* (2001) showed that various conceptual rainfall–runoff models may perform very differently for different catchments, even those located in relatively similar climatic conditions. It is also known that the modelling performance may differ noticeably for training and independent validation periods (Amoussou *et al.* 2014). Such differences could also result from differences in climatic conditions in the calibration and validation periods (Osuch *et al.* 2015).

5.2 Comparison of optimization algorithms

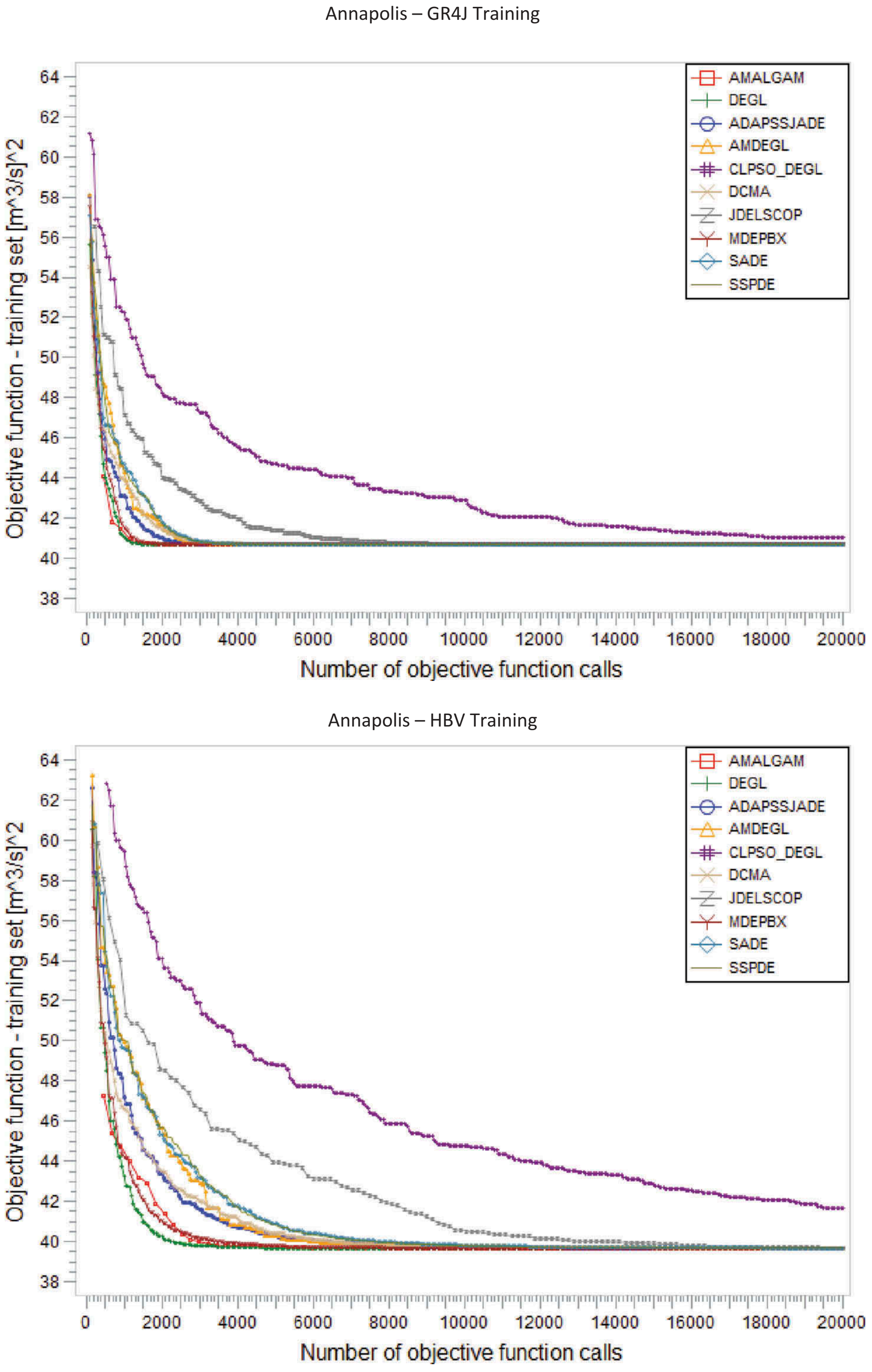
Considering various optimization methods, the results obtained show that the average performance of almost all optimizers tested is similar and none of the methods could be regarded as superior to the others. This is especially clear when tests (with both HBV and GR4J models) are done on data for the Annapolis and Nysa Klodzka catchments, or when GR4J is applied to Allier catchment data. When the Biala Tarnowska catchment is considered, or when the HBV model is applied to Allier River data, some small differences may be found, but they are still meaningless from the hydrological point of view, and in each case they point to another optimization algorithm as the slightly better one. Hence, the experiments on simple conceptual models show that the vast majority of optimization algorithms perform similarly well. This is also true for the two oldest methods, NMA and local search RA, which are considered rather “historical” approaches and are rarely compared with modern metaheuristics. As a large number of optimization methods are tested in this study, including some widely praised approaches (such as SADE, JADE, CLPSO, AMALGAM), such a result is probably not due to inadequate selection of optimization algorithms. As similar conclusions were drawn by Gan and Biftu (1996), who tested just a few methods that were available 20 years ago (including NMA), this means that the use of modern metaheuristics adds little to the practical problem of calibration of simple lumped rainfall–runoff models. The search for the best set of such model parameters turns out to be relatively simple, as suggested by Perrin *et al.* (2003) or Kavetski and Clark (2010).

However, according to 30-run averaged performance, a few methods may be termed slightly poorer than the others, at least for some catchments and models. For example, SFMDE turns out to be slightly poorer than all other algorithms on GR4J calibration for the Annapolis River, and is more clearly inferior to others on HBV calibration for the Biala Tarnowska and Allier rivers. CLPSO-DEGL performs poorly when used to calibrate HBV for the Allier and Nysa Klodzka rivers. The PMS is the only approach showing very uneven performance—it cannot be recommended for calibration of either model for Nysa Klodzka, but leads to marginally the best results when GR4J is applied to Allier and Annapolis catchment data.

It must also be noted here that MCMC DREAM_ZS does indeed find solutions of similar quality to those found by classical optimization algorithms, and in one case (calibration of the GR4J model on Nysa Klodzka data) it even turns out to be marginally the best optimizer according to training data. Such a result confirms that, at least for relatively simple conceptual rainfall–runoff models, DREAM_ZS may indeed be used not only for sampling, but also for calibration purposes, as suggested by Laloy and Vrugt (2012).

5.3 Convergence speed

As almost all algorithms converge to solutions of similar quality, one may ask whether they are similarly quick. In other words, one may wonder how the convergence speed of various algorithms varies. It is difficult to show convergence of so many methods graphically; however, convergence plots of a sample composed of 10 representative algorithms for each model and catchment are given in Figs 1–4. Note that in such figures convergence plots for training data in simulation mode are illustrated, without an using updating procedure (which is applied to the final solutions only in this study), hence the differences between values found in Figs 1–4 and Tables 5 and 6 (accordingly, Tables S1–S8). One may note that for almost every catchment and model, 8 out of 10 methods converge with roughly similar speed, one algorithm (jDElscop) is slightly slower and one (CLPS-DEGL) is much slower than the majority of the methods. The slower


Annapolis – GR4J Training

Annapolis – HBV Training

Figure 1. Convergence plots of 10 selected optimization methods (Annapolis River).

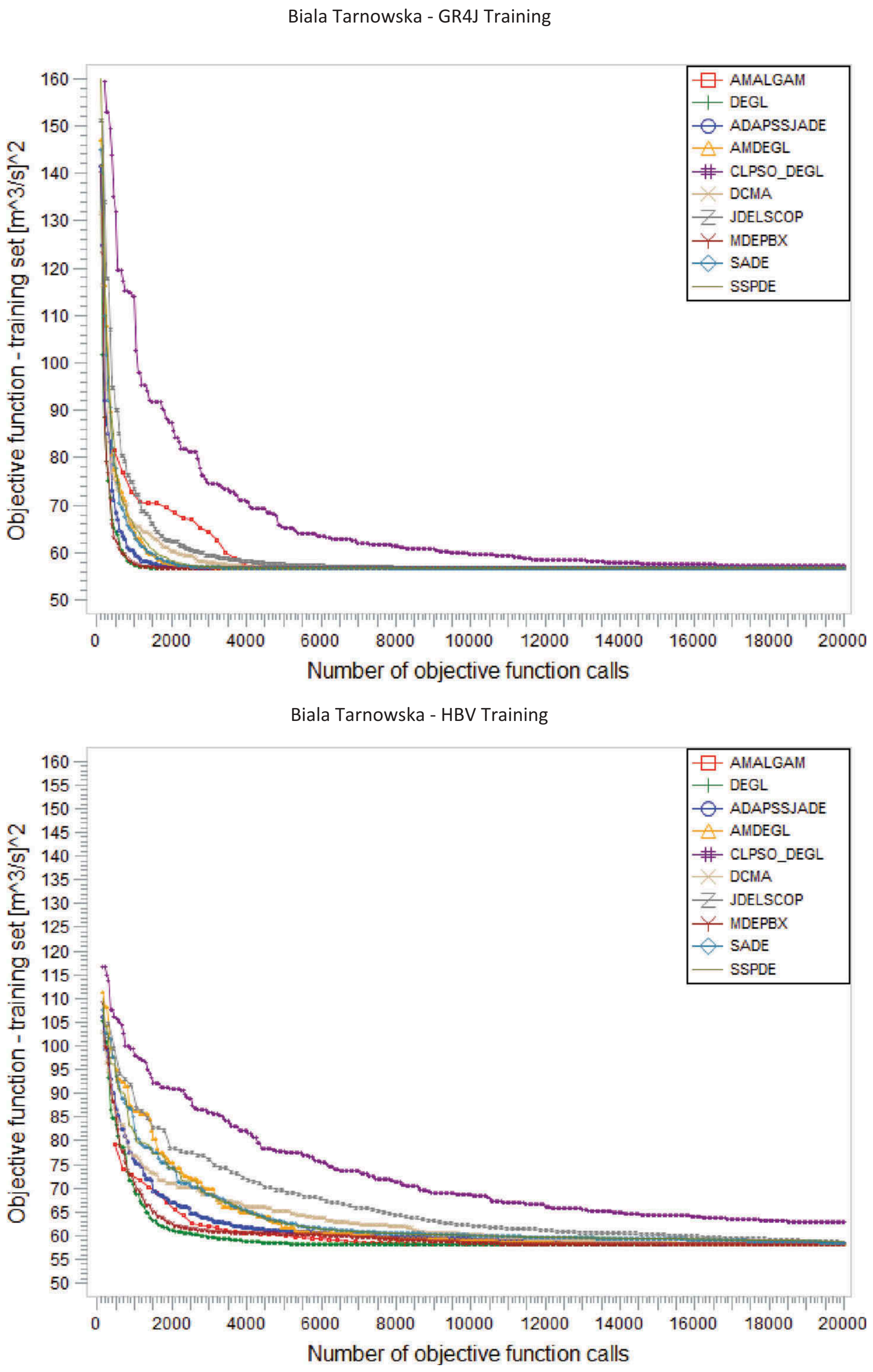


Figure 2. Convergence plots of 10 selected optimization methods (Biala Tarnowska River).

convergence of jDElscop may be easily explained by its method of adapting population size (for detailed procedures, see Brest and Maucec 2011), which depends on the number of allowed function calls. In other words, jDElscop focuses attention on exploration of the search space at earlier stages of the search, hence converges slowly, and at the later stages of the search focuses on exploitation, speeding up convergence to the best local optima found so far. This allows reasonable management of all function calls allowed, but prevents quick convergence of the algorithm, which is seen in almost all plots included in Figs 1–4. We are unable to explain the slow convergence of CLPS-DEGL. However, the main conclusion from Figs 1–4 is that the majority of methods converge with roughly similar speed. Hence, comparing the convergence speed also does not allow identification of the best approach.

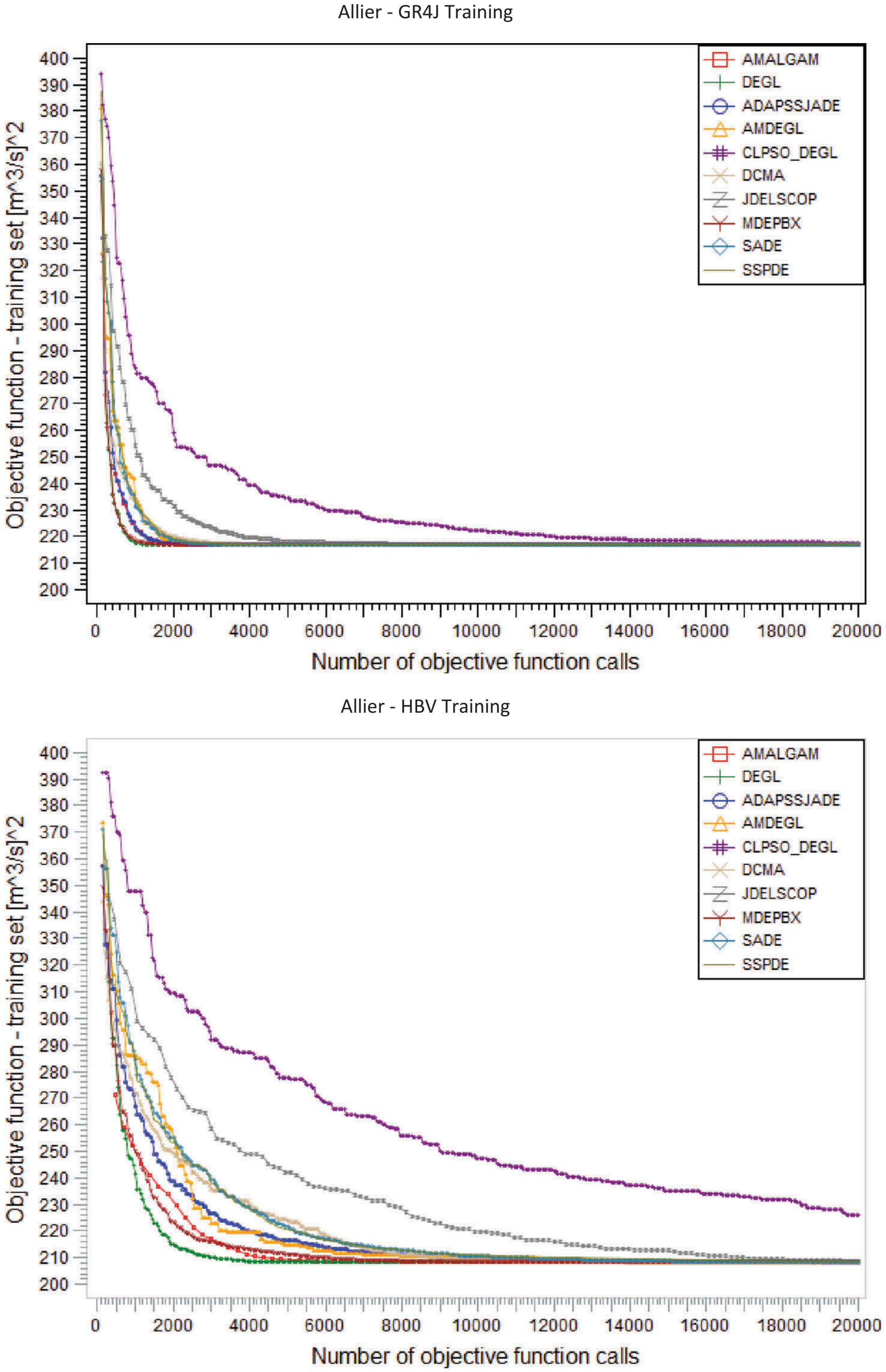


Figure 3. Convergence plots of 10 selected optimization methods (Allier River).

5.4 *Methodological and hydrological perspective: a discussion*

One may look at the results from another perspective. Detailed inspection of the results shown in Tables 5 and 6 and Tables S1–S8 (Supplementary material) reveals that the best solution for training data found during 30 runs by one among the tested methods may be slightly better than both 30-run averaged performance and the best solution found by any other approach. The most evident, but not the sole example of such a case is provided by the GR4J results for the Annapolis catchment data and the PMS approach. This means that, although all algorithms reach similar average performance, they are unable to converge to the global optimum, hence calibration of HBV or GR4J models may be considered as a so-called deceptive problem (Goldberg 1989, Weise *et al*. 2012), in which a common evolutionary principle of survival of the fittest leads to sub-optimal solutions, driving

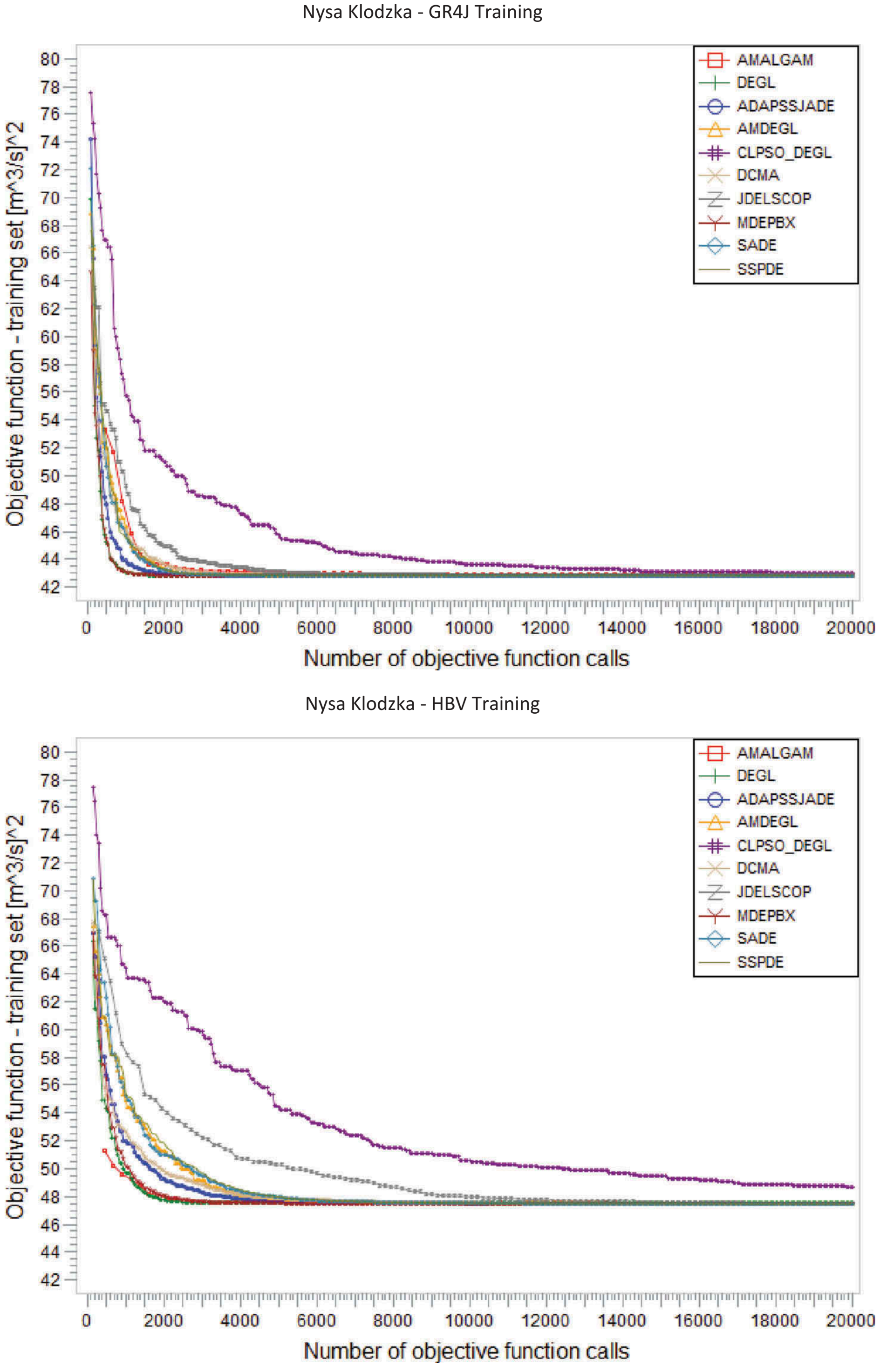


Figure 4. Convergence plots of 10 selected optimization methods (Nysa Klodzka River).

the algorithm away from the global optimum. In fact, in some papers it was empirically shown that for deceptive problems a random search may outperform popular metaheuristics (Oltean 2004, Piotrowski 2013), as should be expected according to the no-free-lunch theorems for optimization (Wolpert and Macready 1997). Interestingly, in the example discussed above, PMS used for GR4J calibration on the Annapolis River found not only the best solution according to training, but also for validation data. Similar deceptive problems, for which the vast majority of methods find solutions of almost equal quality but are unable to determine the global one, are widely used within common problems defined for benchmarking metaheuristics. Probably the best known examples are two artificial multimodal benchmark problems, F8 and F24, from the IEEE Competition on Evolutionary Computation 2005 (Suganthan *et al.* 2005), for which the global optimum is very hard to find by any method, and almost all algorithms end up in the same local optimum.

This suggests that the problem of HBV or GR4J calibration may turn out to be useful for showing the deficiencies of various metaheuristics. However, from a practical hydrological point of view, such marginal differences have to be considered meaningless.

As the results shown in Tables 5 and 6 indicate that the average performance of each optimization method is approximately the same, the question arises whether a good fit of the output from the models to the observed data implies convergence of HBV and GR4J model parameters to one set of “best

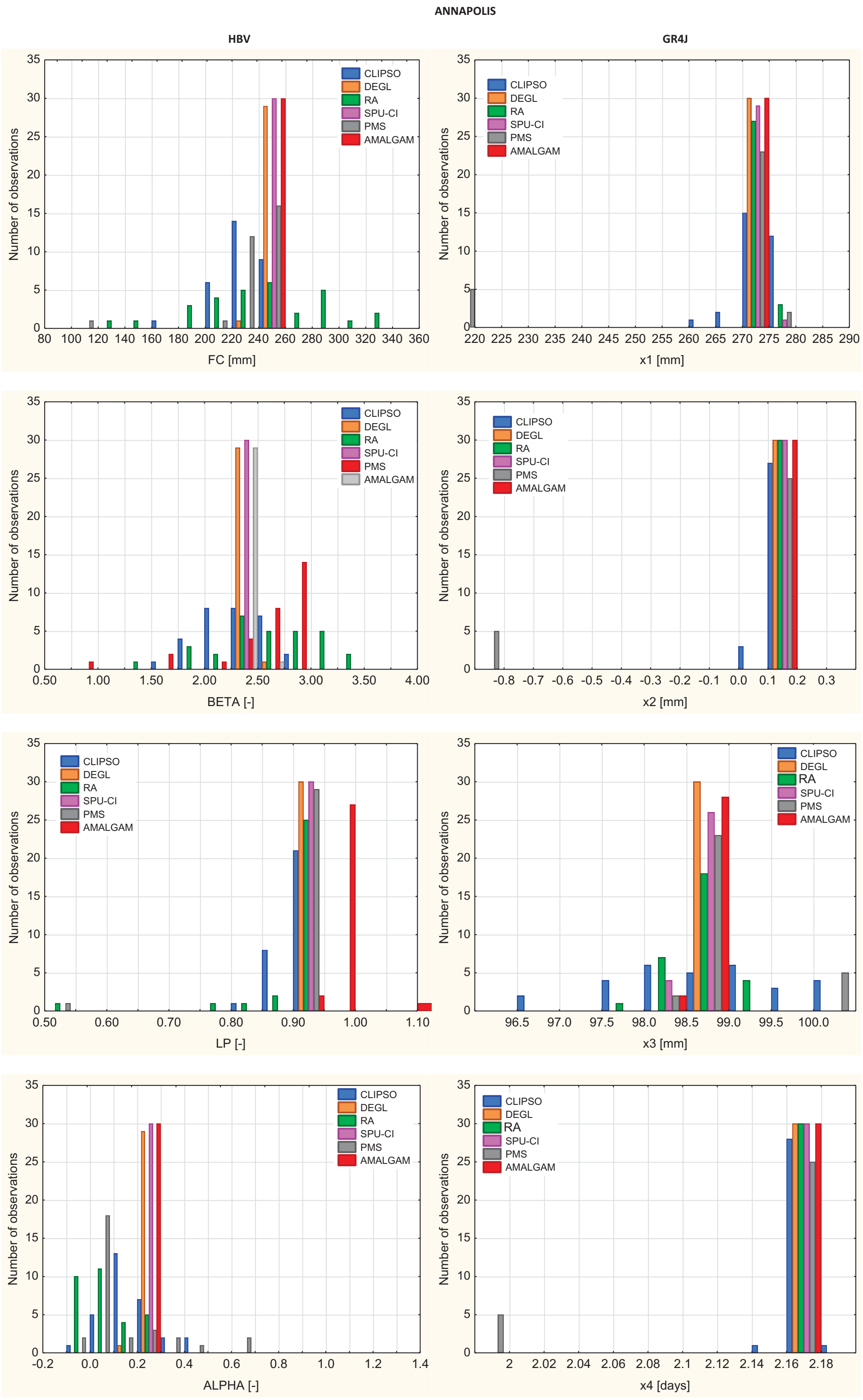


Figure 5. Histogram showing the distribution of GR4J and HBV model parameters obtained during 30 runs by means of six selected optimization algorithms (Annapolis River).

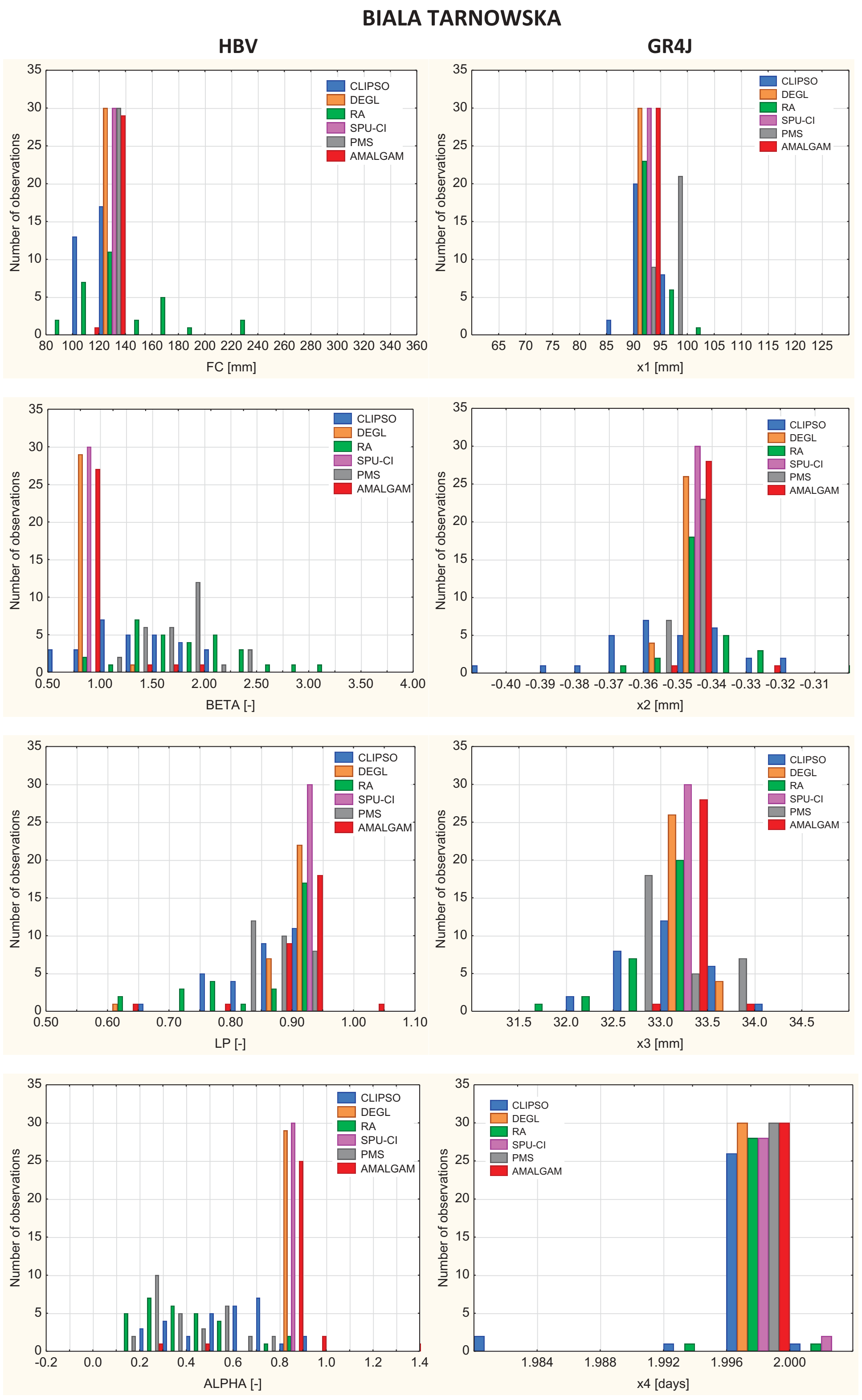


Figure 6. Histogram showing the distribution of GR4J and HBV model parameters obtained during 30 runs by means of six selected optimization algorithms (Biala Tarnowska River).

values". Because the identification of the parameters of both catchment runoff models is a typical example of an ill-posed problem (Napiorkowski 1986), there are different parameter sets within chosen HBV or GR4J model structures that may be acceptable as far as reproducing the observed rainfall–runoff system is concerned.

In Figs 5–8, the histograms of model parameters obtained during 30 runs by six chosen algorithms (including one DE (DEGL), one PSO (CLPSO), one multi-algorithm (AMALGAM), a novel simplex approach (SP-UCI) and two non-population-based approaches—historical RA and novel PMS) are illustrated. As we put most attention on

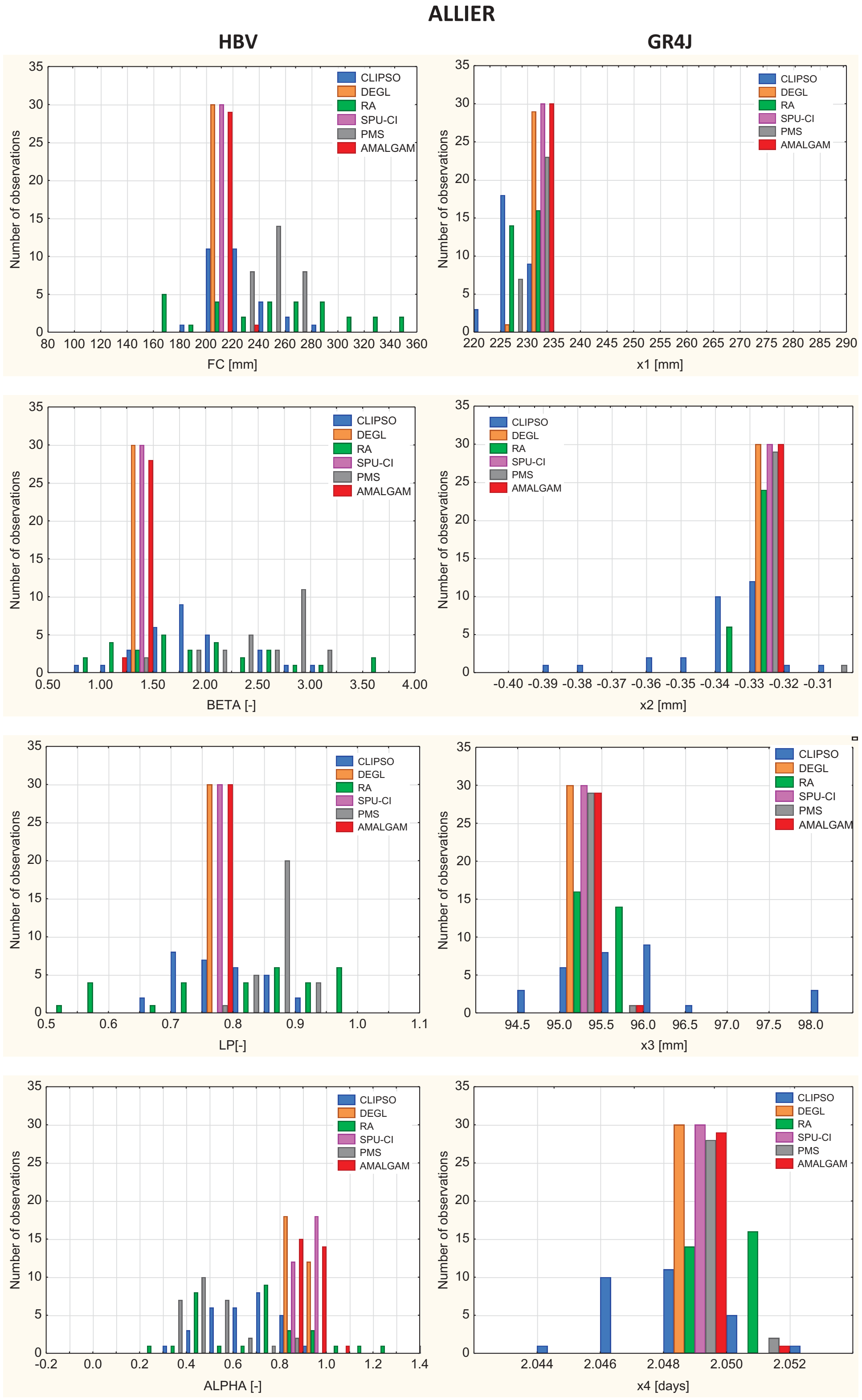


Figure 7. Histogram showing the distribution of GR4J and HBV model parameters obtained during 30 runs by means of six selected optimization algorithms (Allier River).

optimizers in this study, DREAM_ZS results are skipped here. One may note that the spread of parameter values depends on the algorithm: DEGL, AMALGAM and SP-UCI show the most consistent results, while non-population-based approaches and CLPSO show the wider spread ones. Hence, although the final results are of similar quality, some algorithms (often those population based) almost always terminate in the same local optimum, while others (often non-population based, or based on PSO concepts) lead to more scattered solutions.

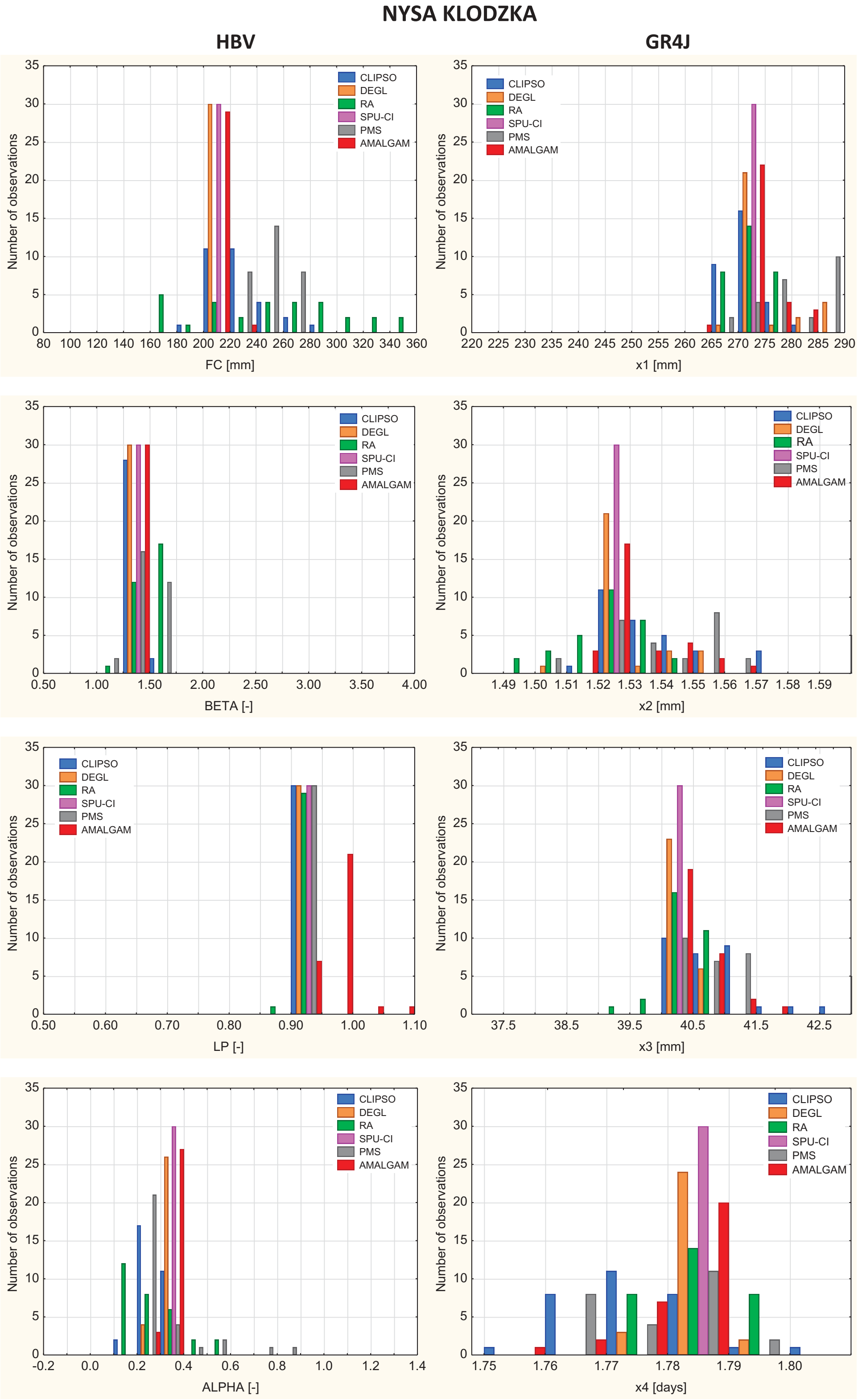


Figure 8. Histogram showing the distribution of GR4J and HBV model parameters obtained during 30 runs by means of six selected optimization algorithms (Nysa Klodzka River).

6 Conclusions

In this study, 26 optimization algorithms, including a number of modern evolutionary or swarm intelligence methods, two historical direct search heuristics and one MCMC sampling approach, have been tested on calibration of simple lumped HBV and GR4J models for four catchments located in roughly similar temperate climatic conditions on two continents.

It has been shown that, with very few exceptions, almost all algorithms perform similarly on each calibration problem and no method may be called superior to the others in terms of the final performance. Although a few methods reach satisfactory solutions slower than the others, the difference in convergence speed among the majority of algorithms is small enough to be of no practical importance. It has also been shown that both historical direct search methods (algorithms proposed by Nelder and Mead 1965, Rosenbrock 1960) and the MCMC sampling approach DREAM_ZS (Laloy and Vrugt 2012) perform equally well in finding the best solutions as modern optimizers, at least where simple lumped conceptual rainfall–runoff models are concerned. Hence there is little room to search for better optimization methods for such models.

On the other hand, it was found that the vast majority of methods, although finding solutions of almost equal quality, do not converge to the global optimum. This is irrelevant from the hydrological perspective, but may be of interest to the optimization community and allows calibration of HBV and GR4J to be termed a deceptive problem (Goldberg 1989); i.e. a problem that lures almost all optimizers to specific local minima, preventing them from finding the global one.

Finally, it was confirmed that neither the HBV nor the GR4J model may be termed superior for all four catchments tested, as should be expected after the tests performed by Perrin *et al.* (2001).

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

Marzena Osuch http://orcid.org/0000-0002-1887-3569

References

Amoussou, E., *et al.*, 2014. Dynamics and modelling of floods in the river basin of Mono in Nangbeto, Togo/Benin. *Hydrological Sciences Journal*, 59 (11), 2060–2071. doi:10.1080/02626667.2013.871015

Bäck, T. and Schwefel, H.P., 1993. An overview of evolutionary algorithms for parameter optimization. *Evolutionary Computation*, 1 (1), 1–23. doi:10.1162/evco.1993.1.1.1

Bergström, S. and Forsman, A., 1973. Development of a conceptual deterministic rainfall–runoff model. *Nordic Hydrology*, 4, 147–170.

Bergström, S., 1976. *Development and application of a conceptual runoff model for Scandinavian catchments*. Norrköping: Svergies Meteorologiska och Hydrologiska Institut, SMHI Report RHO 7, 134 pp.

Bergström, S., 1995. The HBV model. *In*: V.P. Singh, ed. *Computer models of watershed hydrology*. Chapter 13. Highlands Ranch, CO: Water Resources Publications, 443–476.

Bergström, S. and Lindström, G., 2015. Interpretation of runoff processes in hydrological modelling—experience from the HBV approach. *Hydrological Processes*, 29 (16), 3535–3545. doi:10.1002/hyp.10510

Biswas, A., *et al.*, 2013. Physics-inspired optimization algorithms: a survey. *Journal of Optimization*, 2013, 1–16. Article ID 438152.

Blasone, R.S., Madsen, H., and Rosbjerg, D., 2007. Parameter estimation in distributed hydrological modelling: comparison of global and local optimisation techniques. *Nordic Hydrology*, 38 (4–5), 451–476. doi:10.2166/nh.2007.024

Boussaid, I., Lepagnot, J., and Siarry, P., 2013. A survey on optimization metaheuristics. *Information Sciences*, 237, 82–117. doi:10.1016/j.ins.2013.02.041

Brest, J. and Maucec, M.S., 2011. Self-adaptive differential evolution algorithm using population size reduction and three strategies. *Soft Computing*, 15, 2157–2174. doi:10.1007/s00500-010-0644-5

Cai, Z.H., *et al.*, 2011. A clustering-based differential evolution for global optimization. *Applied Soft Computing*, 11 (1), 1363–1379. doi:10.1016/j.asoc.2010.04.008

Caponio, A., Neri, F., and Tirronen, V., 2009. Super-fit control adaptation in memetic differential evolution frameworks. *Soft Computing*, 13 (8–9), 811–831. doi:10.1007/s00500-008-0357-1

Caraffini, F., *et al.*, 2013. Parallel memetic structures. *Information Sciences*, 227 (1), 60–82. doi:10.1016/j.ins.2012.11.017

Chu, W., Gao, X.G., and Sorooshian, S., 2010. Improving the shuffled complex evolution scheme for optimization of complex nonlinear hydrological systems: application to the calibration of the Sacramento soil-moisture accounting model. *Water Resources Research*, 46. Art. No. W09530. doi:10.1029/2010WR009224

Chu, W., Gao, X., and Sorooshian, S., 2011. A new evolutionary search strategy for global optimization of high-dimensional problems. *Information Sciences*, 181 (22), 4909–4927. doi:10.1016/j.ins.2011.06.024

Chu, W., Yang, T.T., and Gao, X.G., 2014. Comment on “High-dimensional posterior exploration of hydrologic models using multipletry DREAM (ZS) and high-performance computing” by Eric Laloy and Jasper A. Vrugt. *Water Resources Research*, 50 (3), 2775–2780. doi:10.1002/2012WR013341

Cooper, V.A., Nguyen, V.T.V., and Nicell, J.A., 1997. Evaluation of global optimization methods for conceptual rainfall–runoff model calibration. *Water Science and Technology*, 36 (5), 53–60. doi:10.1016/S0273-1223(97)00461-7

Crepinsek, M., Liu, S.H., and Mernik, L., 2012. A note on teaching-learning-based optimization algorithm. *Information Sciences*, 212, 79–93. doi:10.1016/j.ins.2012.05.009

Das, S., *et al.*, 2009. Differential evolution using a neighborhood-based mutation operator. *IEEE Transactions on Evolutionary Computation*, 13 (3), 526–553. doi:10.1109/TEVC.2008.2009457

Das, S. and Suganthan, P.N., 2011. Differential evolution: a survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation*, 15 (1), 27–54. doi:10.1109/TEVC.2010.2059031

Das, S., Mullick, S.S., and Suganthan, P.N., 2016. Recent advances in differential evolution – An updated survey. *Swarm and Evolutionary Computation*, 27, 1–30. doi:10.1016/j.swevo.2016.01.004

Doherty, J., 2009. *PEST: model independent parameter estimation, software*. Corinda, Queensland: Watermark Numer. Comput. Available from: www.pesthomepage.org

Dokou, Z. and Karatzas, G.P., 2013. Multi-objective optimization for free-phase LNAPL recovery using evolutionary computation algorithms. *Hydrological Sciences Journal*, 58 (3), 671–685. doi:10.1080/02626667.2012.754103

Dorigo, M., Maniezzo, V., and Colorni, A., 1996. Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems Man and Cybernetics Part B – Cybernetics*, 26 (1), 29–41. doi:10.1109/3477.484436

Duan, Q.Y., Sorooshian, S., and Gupta, V.K., 1992. Effective and efficient global optimization for conceptual rainfall–runoff models. *Water Resources Research*, 28 (4), 1015–1031. doi:10.1029/91WR02985

Duan, Q., *et al.*, 2006. Model parameter estimation experiment (MOPEX): an overview of science strategy and major results from the second and third workshops. *Journal of Hydrology*, 320, 3–17. doi:10.1016/j.jhydrol.2005.07.031

Eberhart, R.C. and Kennedy, J., 1995. A new optimizer using particle swarm theory. *In*: *Proc. 6th Int. Symp. Micromachine human Sci.*, Nagoya, Japan. Piscataway, NJ: Institute of Electrical and Electronics Engineers, Inc., 39–43.

Edijatno, N.N.O., *et al.*, 1999. GR3J: a daily watershed model with three free parameters. *Hydrological Sciences Journal*, 44 (2), 263–277. doi:10.1080/02626669909492221

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

Elci, A. and Ayvaz, M.T., 2014. Differential evolution algorithm based optimization for the site selection of groundwater production wells with the consideration of the vulnerability concept. *Journal of Hydrology*, 511, 736–749. doi:10.1016/j.jhydrol.2014.01.071

Epitropakis, M.G., Plagianakos, V.P., and Vrahatis, M.N., 2012. Evolving cognitive and social experience in particle swarm optimization through differential evolution: a hybrid approach. *Information Sciences*, 216, 50–92. doi:10.1016/j.ins.2012.05.017

Ficchi, A., Perrin, C., and Andreassian, V., 2016. Impact of temporal resolution of inputs on hydrological model performance: an analysis based on 2400 flood events. *Journal of Hydrology*, 538, 454–470. doi:10.1016/j.jhydrol.2016.04.016

Franchini, M., Galeati, G., and Berra, S., 1998. Global optimization techniques for the calibration of conceptual rainfall–runoff models. *Hydrological Sciences Journal*, 43 (3), 443–458. doi:10.1080/02626669809492137

Gan, T.Y. and Biftu, G.F., 1996. Automatic calibration of conceptual rainfall–runoff models: optimization algorithms, catchment conditions, and model structure. *Water Resources Research*, 32 (12), 3513–3524. doi:10.1029/95WR02195

Ghosh, S., *et al*., 2012. A differential covariance matrix adaptation evolutionary algorithm for real parameter optimization. *Information Sciences*, 181 (1), 199–219. doi:10.1016/j.ins.2011.08.014

Glover, F., 1986. Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13 (5), 533–549. doi:10.1016/0305-0548(86)90048-1

Goldberg, D.E., 1989. *Genetic algorithms in search, optimization, and machine learning*. Boston, MA: Addison-Wesley Longman Publishing.

Gong, W.Y., *et al*., 2011. Adaptive strategy selection in differential evolution for numerical optimization: an empirical study. *Information Sciences*, 181 (24), 5364–5386. doi:10.1016/j.ins.2011.07.049

Gong, W.Y. and Cai, Z.H., 2013. Differential evolution with ranking-based mutation operators. *IEEE Transactions on Cybernetics*, 43 (6), 2066–2081. doi:10.1109/TCYB.2013.2239988

Goswami, M. and O'Connor, K.M., 2007. Comparative assessment of six automatic optimization techniques for calibration of a conceptual rainfall–runoff model. *Hydrological Sciences Journal*, 52 (3), 432–449. doi:10.1623/hysj.52.3.432

Gupta, H.V., *et al*., 2014. Large-sample hydrology: a need to balance depth with breadth. *Hydrology and Earth System Sciences*, 18, 463–477. doi:10.5194/hess-18-463-2014

Hansen, N. and Ostermeier, A., 1996. Adapting arbitrary normal mutation distributions in evolution strategies: the covariance matrix adaptation. *In*: *Proc. IEEE Int. Conf. Evol. Comput*., Nagoya, Japan. Piscataway, NJ: Institute of Electrical and Electronics Engineers, Inc., 312–317.

Hastings, W.K., 1970. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57, 97–109. doi:10.1093/biomet/57.1.97

Helwig, S., Branke, J., and Mostaghim, S., 2013. Experimental analysis of bound handling techniques in particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 17 (2), 259–271. doi:10.1109/TEVC.2012.2189404

Holland, I.H., 1975. *Adaptation in natural and artificial systems*. Ann Arbor: University of Michigan. Press.

Ibbitt, R.P. and O'Donnell, T., 1971. Fitting methods for conceptual catchment models. *Journal of Hydraulics Division ASCE*, 97 (9), 1331–1342.

Islam, S.M., *et al*., 2012. An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization. *IEEE Transactions on Systems, Man and Cybernetics, Part B – Cybernetics*, 42 (2), 482–500. doi:10.1109/TSMCB.2011.2167966

Jia, D.L., Zheng, G.X., and Khan, M.K., 2011. An effective memetic differential evolution algorithm based on chaotic local search. *Information Sciences*, 181 (15), 3175–3187. doi:10.1016/j.ins.2011.03.018

Johnston, P.R. and Pilgrim, D., 1976. Parameter optimization for watershed models. *Water Resources Research*, 12 (3), 477–486. doi:10.1029/WR012i003p00477

Kavetski, D. and Clark, M.P., 2010. Ancient numerical daemons of conceptual hydrological modeling: 2. Impact of time stepping schemes on model analysis and prediction. *Water Resources Research*, 46, Art. No. W10511. doi:10.1029/2009WR008896

Kavetski, D. and Clark, M.P., 2011. Numerical troubles in conceptual hydrology: approximations, absurdities and impact on hypothesis testing. *Hydrological Processes*, 25, 661–667. doi:10.1002/hyp.7899

Kim, S.M., *et al*., 2007. Comparison of hydrologic calibration of HSPF using automatic and manual methods. *Water Resources Research*, 43, Art. No. W01402. doi:10.1029/2006WR004883

Kisi, O., 2004. Daily suspended sediment modelling using a fuzzy differential evolution approach. *Hydrological Sciences Journal*, 49 (1), 183–197. doi:10.1623/hysj.49.1.183.54001

Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P., 1983. Optimization by simulated annealing. *Science*, 220, 671–680. doi:10.1126/science.220.4598.671

Kitanidis, P.K. and Bras, R.L., 1980. Real-time forecasting with a conceptual hydrologic model. 2: application and results. *Water Resources Research*, 16 (6), 1034–1044. doi:10.1029/WR016i006p01034

Kolda, T.G., Lewis, R.M., and Torczon, V., 2003. Optimization by direct search: new perspectives on some classical and modern methods. *SIAM Review*, 45 (3), 385–482. doi:10.1137/S003614450242889

Koza, J.R., 1992. *Genetic programming: on the programming of computers by means of natural selection*. Cambridge, MA: MIT Press.

Kuczera, G., 1997. Efficient subspace probabilistic parameter optimization for catchment models. *Water Resources Research*, 33 (1), 177–185. doi:10.1029/96WR02671

Lagarias, J.C., *et al*., 1998. Convergence properties of the Nelder–Mead simplex method in low dimensions. *SIAM Journal on Optimization*, 9 (1), 112–147. doi:10.1137/S1052623496303470

Laloy, E. and Vrugt, J.A., 2012. High-dimensional posterior exploration of hydrologic models using multiple-try DREAM(ZS) and high-performance computing. *Water Resources Research*, 48, Art. No. W01526. doi:10.1029/2011WR010608

Liang, J.J., *et al*., 2006. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Transactions on Evolutionary Computation*, 10 (3), 281–295. doi:10.1109/TEVC.2005.857610

Lindström, G., 1997. A simple automatic calibration routine for the HBV model. *Nordic Hydrology*, 28 (3), 153–168.

Madsen, H., 2000. Automatic calibration of a conceptual rainfall–runoff model using multiple objectives. *Journal of Hydrology*, 235, 276–288. doi:10.1016/S0022-1694(00)00279-1

Madsen, H., *et al*., 2000. Data assimilation in rainfall–runoff forecasting. *In*: *Fourth international conference on hydroinformatics*, 23–27 July 2000, Cedar Rapids, IA. Iowa City, IA: University of Iowa, College of Engineering.

Madsen, H., Wilson, G., and Ammentorp, H.C., 2002. Comparison of different automated strategies for calibration of rainfall–runoff models. *Journal of Hydrology*, 261, 48–59. doi:10.1016/S0022-1694(01)00619-9

Madsen, H. and Skotner, C., 2005. Adaptive state updating in real-time river flow forecasting – a combined filtering and error forecasting procedure. *Journal of Hydrology*, 308, 302–312. doi:10.1016/j.jhydrol.2004.10.030

Maier, H.R., *et al*., 2014. Evolutionary algorithms and other metaheuristics in water resources: current status, research challenges and future directions. *Environmental Modelling & Software*, 62, 271–299. doi:10.1016/j.envsoft.2014.09.013

Mallipeddi, R., *et al*., 2011. Differential Evolution algorithm with ensemble of parameters and mutation strategies. *Applied Soft Computing*, 11 (2), 1679–1696. doi:10.1016/j.asoc.2010.04.024

Metropolis, N., *et al.*, 1953. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21, 1087–1091. doi:10.1063/1.1699114

Napiorkowski, J.J., 1986. Application of Volterra series to modelling of rainfall–runoff systems and flow in open channels. *Hydrological Sciences Journal*, 31 (2), 187–203. doi:10.1080/02626668609491039

Nelder, A. and Mead, R., 1965. A simplex-method for function minimization. *The Computer Journal*, 7 (4), 308–313. doi:10.1093/comjnl/7.4.308

Neri, F. and Tirronen, V., 2010. Recent advances in differential evolution: a survey and experimental analysis. *Artificial Intelligence Review*, 33 (1–2), 61–106. doi:10.1007/s10462-009-9137-2

Noman, N. and Iba, H., 2008. Accelerating differential evolution using an adaptive local search. *IEEE Transactions on Evolutionary Computation*, 12 (1), 107–125. doi:10.1109/TEVC.2007.895272

Oltean, M., 2004. Searching for a practical evidence for the no free lunch theorems. *In*: A.J. Ljspeert, M. Murata, and N. Wakamiya, eds. *Bioinspired approaches to advanced information technology*. Lausanne, Switzerland: Springer.

Osuch, M., Romanowicz, R.J., and Booij, M., 2015. The influence of parametric uncertainty on the relationships between HBV model parameters and climatic characteristics. *Hydrological Sciences Journal*, 60 (7–8), 1299–1316. doi:10.1080/02626667.2014.967694

Pan, Q.K., *et al.*, 2011. A differential evolution algorithm with self-adapting strategy and control parameters. *Computers & Operations Research*, 38, 394–408. doi:10.1016/j.cor.2010.06.007

Pechlivanidis, I.G., *et al.*, 2011. Catchment scale hydrological modeling: a review of model types, calibration approaches and uncertainty analysis methods in the context of recent developments in technology and applications. *Global NEST Journal*, 13 (3), 193–214.

Perrin, C., 2000. Vers une amélioration d'un modèle global pluie-débit au travers d'une approche comparative. PhD Thesis, INPG (Grenoble) and Cemagref (Antony), France, 530 pp, (in French).

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

Perrin, C., Michel, C., and Andreassian, V., 2003. Improvement of a parsimonious model for streamflow simulation. *Journal of Hydrology*, 279, 275–289. doi:10.1016/S0022-1694(03)00225-7

Pickup, G., 1977. Testing the efficiency of algorithms and strategies for automatic calibration of rainfall–runoff models. *Hydrological Sciences Bulletin*, 22 (2), 257–274. doi:10.1080/02626667709491716

Piotrowski, A.P. and Napiorkowski, J.J., 2010. Grouping differential evolution algorithm for multi-dimensional optimization problems. *Control and Cybernetics*, 39 (2), 527–550.

Piotrowski, A.P., Napiorkowski, J.J., and Kiczko, A., 2012a. Differential evolution algorithm with separated groups for multi-dimensional optimization problems. *European Journal of Operational Research*, 216, 33–46. doi:10.1016/j.ejor.2011.07.038

Piotrowski, A.P., Napiorkowski, J.J., and Kiczko, A., 2012b. Corrigendum to: "Differential evolution algorithm with separated groups for multi-dimensional optimization problems" (Eur. J. Oper. Res. 216 (2012) 33–46). *European Journal of Operational Research*, 219, 488. doi:10.1016/j.ejor.2011.12.043

Piotrowski, A.P. and Napiorkowski, J.J., 2012. Product-units neural networks for catchment runoff forecasting. *Advances in Water Resources*, 49, 97–113. doi:10.1016/j.advwatres.2012.05.016

Piotrowski, A.P., 2013. Adaptive memetic differential evolution with global and local neighborhood-based mutation operators. *Information Sciences*, 241, 164–194. doi:10.1016/j.ins.2013.03.060

Piotrowski, A.P., Napiorkowski, J.J., and Rowinski, P.M., 2014. How novel is the "novel" black hole optimization approach? *Information Sciences*, 267, 191–200. doi:10.1016/j.ins.2014.01.026

Piotrowski, A.P., *et al.*, in press. On the importance of training methods and ensemble aggregation for runoff prediction by means of artificial neural networks. *Hydrological Sciences Journal*. doi:10.1080/02626667.2015.1085650

Piotrowski, A.P., in press. Review of differential evolution population size. *Swarm and Evolutionary Computation*. doi:10.1016/j.swevo.2016.05.003

Pokhrel, B.K., *et al.*, 2014. Comparison of two snowmelt modelling approaches in the Dudh Koshi basin (eastern Himalayas, Nepal). *Hydrological Sciences Journal*, 59 (8), 1507–1518. doi:10.1080/02626667.2013.842282

Qi, W., *et al.*, 2016. Quantifying dynamic sensitivity of optimization algorithm parameters to improve hydrological model calibration. *Journal of Hydrology*, 533, 213–223. doi:10.1016/j.jhydrol.2015.11.052

Qin, A.K., Huang, V.L., and Suganthan, P.N., 2009. Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Transactions on Evolutionary Computation*, 13 (2), 398–417. doi:10.1109/TEVC.2008.927706

Reed, P.M., *et al.*, 2013. Evolutionary multiobjective optimization in water resources: the past, present, and future. *Advances in Water Resources*, 51, 438–456. doi:10.1016/j.advwatres.2012.01.005

Refsgaard, J.C., 1997. Validation and intercomparison of different updating procedures for real-time forecasting. *Nordic Hydrology*, 28, 65–84.

Ren, H.Y., *et al.*, 2016. Classification of hydrological parameter sensitivity and evaluation of parameter transferability across 431 US MOPEX basins. *Journal of Hydrology*, 536, 92–108. doi:10.1016/j.jhydrol.2016.02.042

Rojas-Serna, C., *et al.*, in press. How should a rainfall–runoff model be parameterized in an almost ungauged catchment? A methodology tested on 609 catchments. *Water Resources Research*. doi:10.1002/2015WR018549

Romanowicz, R.J., Osuch, M., and Grabowiecka, M., 2013, On the choice of calibration periods and objective functions: a practical guide to model parameter identification. *Acta Geophysica*, 61 (6), 1477–1503. doi:10.2478/s11600-013-0157-6

Rosenbrock, H.H., 1960. An automated method for finding the greatest or least value of a function. *Computer Journal*, 3 (3), 175–184. doi:10.1093/comjnl/3.3.175

Sorensen, K., 2015, Metaheuristics—the metaphor exposed. *International Transactions in Operational Research*, 22 (1), 3–18. doi:10.1111/itor.12001

Storn, R. and Price, K.V., 1995. *Differential Evolution – a simple and efficient adaptive scheme for global optimization over continuous spaces*. Berkeley, CA: International Computer Sciences Institute, Tech. Report TR-95-012.

Suganthan, P.N., *et al.*, 2005. Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. Nanyang Technol. Univ., Singapore, Tech. Rep. KanGAL #2005005. India: IIT Kanpur.

Tang, Y., Reed, P., and Wagener, T., 2006. How effective and efficient are multiobjective evolutionary algorithms at hydrologic model calibration? *Hydrology and Earth System Sciences*, 10, 289–307. doi:10.5194/hess-10-289-2006

ter Braak, C.J.F. and Vrugt, J.A., 2008. Differential evolution Markov Chain with snooker updater and fewer chains. *Soft Computing*, 18, 435–446.

Thirel, G., *et al.*, 2015. Hydrology under change: an evaluation protocol to investigate how hydrological models deal with changing catchments. *Hydrological Sciences Journal*, 60 (7–8), 1184–1199. doi:10.1080/02626667.2014.967248

Thornthwaite, C.W., 1948. An approach toward a rational classification of climate. *Geographical Review*, 38 (1), 55–94. doi:10.2307/210739

Tigkas, D., Christelis, V., and Tsakiris, G., 2015. The global optimisation approach for calibrating hydrological models: the case of Medbasin - D model. *In*: *Proceedings of the 9th EWRA world Congress "Water Resources Management in a Changing World: challenges and Opportunities"*, Istanbul, Turkey. Athens: European Water Resources Association.

Tolson, B.A. and Shoemaker, C.A., 2007. Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. *Water Resources Research*, 43, Art. No. W01413. doi:10.1029/2005WR004723

Trescott, P.C., 1968. *Groundwater resources and hydrogeology of the Annapolis-Cronwallis Valley, Nova Scotia*. Nova Scotia: Nova Scotia Department of Mines, Memoir 6, Halifax, 159p.

Valery, A., Andreassian, V., and Perrin, C., 2014a. 'As simple as possible but not simpler': What is useful in a temperature-based snow-accounting routine? Part 1 - Comparison of six snow accounting routines on 380 catchments. *Journal of Hydrology*, 517, 1166–1175. doi:10.1016/j.jhydrol.2014.04.059

Valery, A., Andreassian, V., and Perrin, C., 2014b. 'As simple as possible but not simpler': what is useful in a temperature-based snow-accounting routine? Part 2 - Sensitivity analysis of the Cemaneige snow accounting routine on 380 catchments. *Journal of Hydrology*, 517, 1176–1187. doi:10.1016/j.jhydrol.2014.04.058

Vansteenkiste, T., *et al.*, 2014. Intercomparison of five lumped and distributed models for catchment runoff and extreme flow simulation. *Journal of Hydrology*, 511, 335–349. doi:10.1016/j.jhydrol.2014.01.050

Vidal, J.P., *et al.*, 2010. A 50-year high-resolution atmospheric reanalysis over France with the Safran system. *International Journal of Climatology*, 30, 1627–1644. doi:10.1002/joc.2003

Vrugt, J.A., *et al.*, 2003. A shuffled complex metropolis algorithm for optimization and uncertainty assessment of hydrologic model parameters. *Water Resources Research*, 39 (8), Art. No. 1201. doi:10.1029/2002WR001642

Vrugt, J.A., *et al.*, 2008. Treatment of input uncertainty in hydrologic modeling: doing hydrology backward with Markov chain Monte Carlo simulation. *Water Resources Research*, 44, Art. No. W00B09. doi:10.1029/2007WR006720

Vrugt, J.A., *et al.*, 2009a. Accelerating Markov chain Monte Carlo simulation by differential evolution with self-adaptive randomized subspace sampling. *International Journal of Nonlinear Sciences and Numerical Simulation*, 10 (3), 273–290. doi:10.1515/IJNSNS.2009.10.3.273

Vrugt, J.A., Robinson, B.A., and Hyman, J.M., 2009b. Self-adaptive multimethod search for global optimization in real-parameter spaces. *IEEE Transactions on Evolutionary Computation*, 13 (2), 243–259. doi:10.1109/TEVC.2008.924428

Vrugt, J.A., *et al.*, 2012. Hydrologic data assimilation using particle Markov chain Monte Carlo simulation: theory, concepts and applications. *Advances in Water Resources*, 51, 457–478. doi:10.1016/j.advwatres.2012.04.002

Vrugt, J.A. and Laloy, J., 2014. Reply to comment by Chu et al. on "High-dimensional posterior exploration of hydrologic models using multiple-try DREAM(ZS) and high-performance computing". *Water Resources Research*, 50 (3), 2781–2786. doi:10.1002/2013WR014425

Vrugt, J.A., 2016. Markov chain Monte Carlo simulation using the DREAM software package: theory, concepts, and MATLAB implementation. *Environmental Modelling & Software*, 75, 273–316. doi:10.1016/j.envsoft.2015.08.013

Wang, Y.C., Yu, P.S., and Yang, T.C., 2010. Comparison of genetic algorithms and shuffled complex evolution approach for calibrating distributed rainfall–runoff model. *Hydrological Processes*, 24, 1015–1026. doi:10.1002/hyp.7543

Wang, Y., Cai, Z.X., and Zhang, Q.F., 2011. Differential evolution with composite trial vector generation strategies and control parameters. *IEEE Transactions on Evolutionary Computation*, 15 (1), 55–66. doi:10.1109/TEVC.2010.2087271

Weise, T., Chiong, R., and Tang, K., 2012. Evolutionary optimization: pitfalls and booby traps. *Journal of Computer Science and Technology*, 27 (5), 907–936. doi:10.1007/s11390-012-1274-4

Weyland, D., 2010. A rigorous analysis of the harmony search algorithm – How the research community can be misled by a "novel" methodology. *International Journal of Applied Metaheuristic Computing*, (1–2), 50–60. doi:10.4018/jamc.2010040104

Willems, P., 2014. Parsimonious rainfall–runoff model construction supported by time series processing and validation of hydrological extremes – Part 1: step-wise model-structure identification and calibration approach. *Journal of Hydrology*, 510, 578–590. doi:10.1016/j.jhydrol.2014.01.017

Willems, P., *et al.*, 2014. Parsimonious rainfall–runoff model construction supported by time series processing and validation of hydrological extremes – Part 2: intercomparison of models and calibration approaches. *Journal of Hydrology*, 510, 591–609. doi:10.1016/j.jhydrol.2014.01.028

Wolpert, D.H. and Macready, W.G., 1997. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1 (1), 67–82. doi:10.1109/4235.585893

Xing, B. and Gao, W.J., 2014. *Innovative computational intelligence: a rough guide to 134 clever algorithms*. Intelligent Systems Reference Library series, vol. 62. Germany: Springer, 451 pp.

Xu, D.M., *et al.*, 2013. Comparison of three global optimization algorithms for calibration of the Xinanjiang model parameters. *Journal of Hydroinformatics*, 15 (1), 174–193. doi:10.2166/hydro.2012.053

Yapo, P.O., Gupta, H.V., and Sorooshian, S., 1998. Multi-objective global optimization for hydrologic models. *Journal of Hydrology*, 204, 83–97. doi:10.1016/S0022-1694(97)00107-8

Zhang, J. and Sanderson, A.C., 2009. JADE: adaptive differential evolution with optional external archive. *IEEE Transactions on Evolutionary Computation*, 13 (5), 945–958. doi:10.1109/TEVC.2009.2014613

Zheng, F.F., Simpson, A.R., and Zecchin, A.C., 2011. A combined NLP-differential evolution algorithm approach for the optimization of looped water distribution systems. *Water Resources Research*, 47, Art. No. W08531. doi:10.1029/2011WR010394

Zhou, Y., Li, X., and Gao, L., 2013, A differential evolution algorithm with intersect mutation operator. *Applied Soft Computing*, 13 (1), 390–401. doi:10.1016/j.asoc.2012.08.014

APPENDIX

Applied HBV model version

The input variables to the HBV version used in this study include: daily precipitation (Precip), mean air temperature (Temp) and estimated potential evapotranspiration (Pet) calculated by the Thornthwaite method (Thornthwaite 1948). HBV has five state variables representing storages of snow pack (ssp), snowmelt water (ssw), soil moisture (ssm), fast runoff (sfr) and base flow (sbf). Precipitation may occur in the form of rainfall (r), snowfall (s) or a mixture of snowfall and rainfall. In the model, the threshold temperature (TT, °C) is used to distinguish rainfall from snowfall. It is assumed that at the TT half of the precipitation consists of snow. The TT is extended to an interval TTI (°C) and within this interval precipitation is assumed to be a mix of rain and snow, decreasing linearly from 100% snow at the lower bound to 0% at the upper bound, i.e.:

$$\begin{aligned} s(t) &= \mathrm{Precip}(t)\cdot(\mathrm{TT}+0.5\cdot\mathrm{TTI}-\mathrm{Temp}(t))/\mathrm{TTI} \\ r(t) &= \mathrm{Precip}(t)\cdot(\mathrm{Temp}(t)-(\mathrm{TT}-0.5\cdot\mathrm{TTI}))/\mathrm{TTI} \end{aligned} \tag{A1}$$

Precipitation is assumed to be in the form of snowfall if Temp(t) remains below the interval. If Temp(t) is above the interval only rain occurs, otherwise precipitation is considered to be a mix of snow and rain. Snowfall is added to the snow reservoir and rainfall is added to the free water reservoir, which represents the liquid water content of the snow pack.

Daily snowmelt water (sw) is computed by means of the degree-day method:

$$\mathrm{sw}(t) = \min(\mathrm{CFMAX}\cdot(\mathrm{Temp}(t)-\mathrm{TT}),\mathrm{ssp}(t)) \tag{A2}$$

where CFMAX is the degree-day factor (mm/°C d).

The snowpack retains meltwater as long as the amount of water does not exceed a certain fraction of the snow (WHC, mm/mm). When the temperature decreases below TT, this water refreezes gradually according to the refreezing factor (CFR, dimensionless), which reflects the fraction of water that will freeze after being released from the melting snow (sr):

$$\mathrm{sr}(t) = \min(\mathrm{CFR}\cdot\mathrm{CFMAX}\cdot(\mathrm{TT}-\mathrm{Temp}(t)),\mathrm{ssw}(t)) \tag{A3}$$

The precipitation routine generates inflow (in) to the soil moisture routine:

$$\mathrm{in}(t) = \max(\mathrm{ssw}(t) + \mathrm{sw}(t) + r(t) - \mathrm{sr}(t) - \mathrm{WHC} \cdot \mathrm{ssp}(t), 0) \qquad (A4)$$

where WHC is the water holding capacity of snow.

The main part of the HBV model is the soil moisture routine. This module receives inflow (in) calculated by means of Equation (A4) and computes the state of soil moisture (ssm) based on the direct runoff (qd), the ground water recharge (qin) and the actual evapotranspiration (ea). In addition, water can be drawn up from the groundwater zone to the soil moisture zone. This routine is based on the three parameters, BETA (–), LP (–) and FC (mm), being, respectively: the shape coefficient that describes the discharge from the unsaturated zone to the fast runoff reservoir; the soil moisture value above which evapotranspiration reaches its potential value; and the maximum soil moisture storage in the model. The parameter LP is given as a fraction of FC.

If inflow generated from the precipitation routine is greater than the empty part of the soil moisture reservoir, i.e. in(t) > FC – ssm(t), then the direct runoff (qd) is transferred directly to the fast runoff reservoir:

$$\mathrm{qd}(t) = \max((\mathrm{in}(t) + \mathrm{ssm}(t) - \mathrm{FC}), 0) \qquad (A5)$$

and the groundwater recharge from the soil moisture routine is calculated as:

$$\mathrm{qin}(t) = \left(\frac{\mathrm{ssm}(t)}{\mathrm{FC}}\right)^{\mathrm{BETA}} \cdot (\mathrm{in}(t) - \mathrm{qd}(t)) \qquad (A6)$$

Potential evapotranspiration (Pet) is reduced to the actual values (ea) according to the simple function of the total computed soil moisture conditions:

$$\mathrm{ea}(t) = \begin{cases} \frac{\mathrm{Pet}(t) \cdot \mathrm{ssm}(t)}{\mathrm{FC}} & \text{if } \mathrm{ssm}(t)/\mathrm{FC} < \mathrm{LP} \\ \mathrm{Pet}(t) & \text{if } \mathrm{ssm}(t)/\mathrm{FC} \geq \mathrm{LP} \end{cases} \qquad (A7)$$

Capillary flow from the upper reservoir to the soil moisture zone is given by:

$$\mathrm{qc}(t) = \mathrm{CFLUX} \cdot \frac{(\mathrm{FC} - \mathrm{ssm}(t))}{\mathrm{FC}} \qquad (A8)$$

where CFLUX is the maximum capillary flow.

The version of the HBV model applied in this paper has a response function represented by means of two reservoirs. Excess water enters the upper zone and then leaves as runoff through its outlet or percolate, at a constant rate (PERC) down to the lower zone.

The upper reservoir (a part of the fast runoff routine) is nonlinear and its outflow is given by

$$\mathrm{qf}(t) = \mathrm{KF} \cdot \mathrm{ssw}(t)^{(1+\mathrm{ALPHA})} \qquad (A9)$$

where ALPHA (–) can be considered as a measure of nonlinearity and KF (1/d) is the recession coefficient.

The other reservoir is used to simulate the baseflow and its outflow depends linearly on retention (sgw):

$$\mathrm{qs}(t) = \mathrm{KS} \cdot \mathrm{sgw}(t) \qquad (A10)$$

where KS (1/d) is the recession coefficient for the slow runoff reservoir. The outflow from the HBV model is formed by a sum of runoff components qf and qs.

To summarize, the adapted version of the HBV model used in this paper has 13 parameters to calibrate: five in the snow routine (TT, TTI, CFMAX, CFR, WHC), three in the soil moisture routine (FC, LP, BETA) and five in the response function (PERC, KF, KS, ALPHA, CFLUX).