EFFICIENT METHODS OF AUTOMATIC CALIBRATION FOR RAINFALL-RUNOFF MODELLING IN THE FLOREON⁺ SYSTEM

M. Theuer, *R. Vavřík,* *V. Vondrák,* *Š. Kuchař,* *B. Šír,* *A. Portero**

Abstract: Calibration of rainfall-runoff model parameters is an inseparable part of hydrological simulations. To achieve more accurate results of these simulations, it is necessary to implement an efficient calibration method that provides sufficient refinement of the model parameters in a reasonable time frame. In order to perform the calibration repeatedly for large amount of data and provide results of calibrated model simulations for the flood warning process in a short time, the method also has to be automated. In this paper, several local and global optimization methods are tested for their efficiency. The main goal is to identify the most accurate method for the calibration process that provides accurate results in an operational time frame (typically less than 1 hour) to be used in the flood prediction Floreon⁺ system. All calibrations were performed on the measured data during the rainfall events in 2010 in the Moravian-Silesian region (Czech Republic) using our in-house rainfall-runoff model.

Key words: *rainfall-runoff modelling, automatic calibration, optimization algorithms, genetic algorithms, parallel implementation*

1.   Introduction

The rainfall-runoff process is a continual process of water exchange between an ocean and continent or within a continental region. It consists of the following basic components: evapotranspiration and atmospherical water transport, precipitation, interception, infiltration, surface runoff, interflow, and baseflow. It is a complex process, which can be described only in a simplified form. This simplification can be modelled by a rainfall-runoff model - the quantitative relation between input and output characteristics of a hydrological system. Mathematically expressed, it is an algorithm for solving the equations describing the structure and behaviour of such system [11, 14]. Rainfall-runoff models can be classified from different points of view, e.g. the hydrological cycle representation (conceptual and physically

───────────────
*Matyáš Theuer – Corresponding author; Radim Vavřík; Vít Vondrák; Štěpán Kuchař; Boris Šír; Antonio Portero; VŠB–Technical University of Ostrava, IT4Innovations National Supercomputing Center, Ostrava, Czech Republic, E-mail: matyas.theuer@vsb.cz, radim.vavrik@vsb.cz, vit.vondrak@vsb.cz, stepan.kuchar@vsb.cz, boris.sir@vsb.cz, antonio.portero@vsb.cz

based models), the level of causality (deterministic and stochastic models), the time discretization (continual and event models), or the spatial discretization (lumped, semi-distributed, and distributed models). Lumped models consider the catchment as one spatial unit with homogeneous hydrological characteristics. In distributed models, spatial heterogeneity of these characteristics can be expressed as a mosaic of quasi-homogenous spatial units. Semi-distributed models serve as a midpoint between lumped and distributed models, where the catchment is divided into sub-catchments or sub-basins, each with homogeneous hydrologic parameters.

Every (hydrological) model is based on parameters [7, 21]. The parameters can be classified as the physical and process ones [23]. The physical parameters can be measured or estimated directly (catchment geometry, land cover, etc.), while the process ones only indirectly (e.g. runoff coefficient, average hydrological parameters etc.). The process parameters, as well as input hydro-meteorological data, are therefore affected by an uncertainty arising from spatio-temporal variability of the hydro-meteorological processes and the method of their measurement or estimation. All errors in the model parameters are then essentially transferred to the model outputs (hydrological model uncertainty is further discussed, e.g., in [4–6, 24]). This means that all hydrological models, especially conceptual lumped and semi-distributed ones, can be fairly inaccurate without calibration.

Model calibration, in the meaning of parameter optimization, is a process of searching for such a parameter combination that leads to the best fit between the simulated and observed hydrographs [1, 7, 24]. This can be mathematically expressed by an objective function that can use different metrics for model accuracy evaluation (e.g. Nash-Sutcliffe coefficient, peak discharge error, etc.). Basically, calibrations can be classified as manual and automatic. In manual calibration, trial-and-error parameter adjustment is made and requires an experienced hydrologist with the knowledge of the catchment behaviour. The manual calibration then provides hydrologically relevant results, but it is very time consuming and highly dependent on the level of expertise of the hydrologist. In automatic calibrations, model parameters are adjusted automatically according to a specific search scheme and numerical goodness-of-fit measure of the calibrated model [16,17]. Due to the complexity of the rainfall-runoff process, automatic optimization methods combined with high performance computing (HPC) create a suitable tool for multi-objective calibration. An objective function for calibration of rainfall-runoff events often contains numerous local minima that optimize only small parts of the resulting hydrograph, but finding a global minimum provides a significant increase in accuracy [12]. The conclusion of the number of studies is that the global population-evolution-based algorithms are more effective than multi-start local search procedures, which in turn perform better than pure local search methods [17]. Successfully used global optimization methods are, e.g., genetic algorithms, hill-climbing methods, shuffled complex evolution (SCE), or simulated annealing [7,17]. Advanced calibration approaches include the statistical methods which take input uncertainties into account (e.g. [22, 28]), Bayesian statistics (e.g. [13, 20, 25, 31]), or theoretical methods respecting the principle of equifinality (GLUE and Monte Carlo methodologies) [8].

Many of these studies only look at the accuracy of the calibration methods but not on their time aspect, which is very important in operational use of the

automated calibration methods and mainly during flooding situations. The main goal of this paper is to look at these methods in the context of our flood prediction system called Floreon$^+$ [18, 30], which executes rainfall-runoff simulations for all major catchments of the Moravian-Silesian region every hour.

The experiments in this paper employ our internally developed semi-distributed rainfall-runoff model called Math1D. It is based on the SCS-CN (curve number) method [7] which is commonly applied to quantifying the amount of direct runoff transformed from the corresponding rainfall in small and medium catchments. The curve numbers are empirical hydrologic parameters derived from the catchment's soil group, land use, and other hydrological parameters. To determine a distribution of runoff in time, the unit hydrograph is used and water contributions from particular streams are computed using the kinematic wave approximation and Manning's equation [29]. Based on the sensitivity analysis and available real hydrologic data, the Manning's roughness coefficient, the initial abstraction, and the curve number were identified as the parameters suitable for calibration. These values are specific for each river stream and sub-basin.

Besides the Math1D model, several rainfall-runoff models are used in the Floreon$^+$ system, and all of them should be calibrated. On top of that, the results of the rainfall-runoff models are used as inputs to the hydrodynamic models, so their high accuracy and timely execution is very important for providing good quality of the service. Therefore, this paper focuses on comparing the proposed methods with this time limitation in mind on all catchments modelled in the Floreon$^+$ system and provides a basic scalability analysis of the used methods to look at the possibilities of their deployment on the HPC cluster.

The paper is structured in the following way. Sec. 2 describes a general automated approach and algorithm to calibration of rainfall-runoff models. Sec. 3 along with Sec. 4 discusses local and global optimization methods investigated in this paper. Sec. 5 provides the methodology used in the calibration experiments and environment for their execution along with the discussion of the experimental results. Sec. 6 concludes this paper and outlines the plans for our future research.

2. Model calibration process

Let us assume a hydrological model $\mathcal{H}$ that transforms a set of time dependent input parameters $\mathbf{R}$ into a set of results $\mathbf{Q} = \mathcal{H}(\mathbf{R})$. Both $\mathbf{Q}$ and $\mathbf{R}$ consist of discrete space and time domains, therefore they are represented by matrices, where the first dimension stands for measurement or prediction stations and the second dimension expresses time steps. Typically, the output from the model $\mathcal{H}$ is not dependent only on the input parameters $\mathbf{R}$ (usually in the form of measured or predicted rainfall) but also on other hydrological parameters $\mathbf{x}$. The values of these parameters are constrained by a set of admissible hydrological parameters $\mathcal{U}_{\mathrm{adm}}$. Typically, we set the lower and upper limits on the calibrated parameters to ensure their physical meanings defining $\mathcal{U}_{\mathrm{adm}} = \{x_i : l_i \leq x_i \leq u_i, i = 1, ..., n\}$, but of course any other type of constraints is allowed. The goal of the calibration is to find such values of the parameters $\mathbf{x}$ that the parametrized hydrological model $\mathcal{H}(\mathbf{x}, \mathbf{R})$ provides the results with the minimal error. If we denote by $\mathbf{Q}(\mathbf{x}) = \mathcal{H}(\mathbf{x}, \mathbf{R})$ output from the hydrological model parametrized by the set of hydrological parameters $\mathbf{x}$ and by

$\mathbf{Q}_{\mathrm{o}}$ a set of observed quantities corresponding to $\mathbf{Q}(\mathbf{x})$, we can measure the model error by an objective function $f(\mathbf{x}) = ||\mathbf{Q}(\mathbf{x}) - \mathbf{Q}_{\mathrm{o}}||$. The choice of the norm $||.||$ is arbitrary and the most frequently used choices for rainfall-runoff modelling are presented in Sec. 5. Using such notation, we can rewrite the model calibration process as a constrained minimization problem

$$\min f(\mathbf{x}) \text{ subject to } \mathbf{x} \in \mathcal{U}_{\mathrm{adm}}. \tag{1}$$

From the computational point of view, the evaluation of the objective function represents the solution of the hydrological model with a given set of parameters $\mathbf{x}$. Depending on the selected optimization method, a certain number of the same hydrological models with different parameter setting have to be evaluated in each optimization step. A general calibration scheme is depicted in Fig. 1.
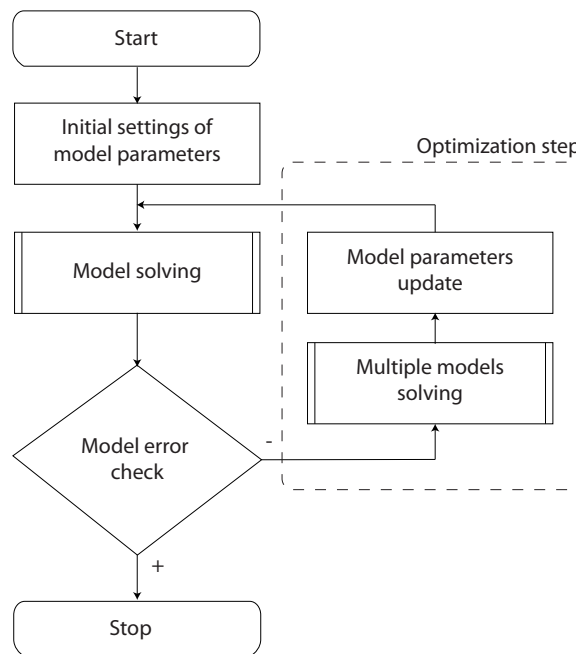
Fig. 1 *General calibration algorithm.*

From the scheme, it is obvious that the most computationally expensive part is the multiple execution of the hydrological model, which forms the essential part of the used optimization algorithm. Therefore, the efficiency of the whole calibration process is dependent on this algorithm. The parallel programming technique seems to be a possibility to speed up the whole calibration process. Unfortunately, not every optimization algorithm can benefit from it. In our paper, we have tested several different optimization methods, both the local and global ones, and we have evaluated them with respect to the resulting model accuracy. In the next section, a brief overview of the tested methods is presented.

3. Local optimization methods

Local optimization methods only find a local minimum of the objective function when starting from the specified initial input vector. To acquire the optimal calibration parameters for an observed rainfall-runoff event using this kind of methods, the objective function has to have only one minimum within the predefined constraints of the calibrated parameters, or the initial parameter values have to be between two local maxima that surround the global minimum of the objective function in each dimension of the function. While using these methods in our experiments, the initial parameters were specified by manual calibration performed by a hydrologist on a time period with stabilized flow.

3.1 Line search method

The first tested local minimization method evaluates the gradient of the objective function for the initial input vector and then changes all calibrated parameters in the direction of the gradient as seen in Eq. (2).

$$\mathbf{x}_{i+1} = \mathbf{x}_i - \mathbf{step}_i \cdot \nabla f(\mathbf{x}_i), \tag{2}$$

where $\mathbf{x}_{i+1}$ is the new vector of calibrated parameters for the next iteration, $\mathbf{x}_i$ is the vector of calibrated parameters in this iteration, $\nabla f(\mathbf{x}_i)$ is the gradient of the objective function for input vector $\mathbf{x}_i$ and $\mathbf{step}_i$ describes a step for this iteration evaluated in Eq. (3).

$$\mathbf{step}_i = \operatorname*{argmin}_{\mathbf{step}}(f(\mathbf{x}_i - \mathbf{step} \cdot \nabla f(\mathbf{x}_i))) \tag{3}$$

The constraints satisfaction is ensured by projection of the gradient on a feasible set in each step. This method iterates until a specified number of iterations has been processed or a specified threshold of the objective function value has been reached.

3.2 Interior point method

The main idea of the interior point methods is to transform a general equality and inequality constrained optimization problem

$$\min f(\mathbf{x}) \text{ subject to } h(\mathbf{x}) = 0 \text{ and } g(\mathbf{x}) \geq 0 \tag{4}$$

into a simply bounded and equality constrained minimization problem

$$\min f(\mathbf{x}) - \mu \sum_i \ln(s_i) \text{ subject to } h(\mathbf{x}) = 0 \text{ and } g(\mathbf{x}) + \mathbf{s} = 0 \tag{5}$$

adding a logarithmic barrier function $\sum_i \ln(s_i)$ and non-negative slack variables $\mathbf{s} \geq 0$. It is easy to see that the solution of the new problem (Eq. (5)) converges to the solution of the original problem (Eq. (4)) as μ tends to 0. The approximate problem (Eq. (5)) is solved in each iteration either using Newton step (solution of Karush-Kuhn-Tucker conditions) or using trust region conjugate gradient step if Newton step is not possible to perform. This method is well known by its robustness and polynomial convergence rate. More details about the interior point methods can be found in [9, 10].

4.   Global optimization methods

Global optimization methods try to find a global minimum of the objective function within given constraints. They, therefore, eliminate the limitations of local methods but at the same time are more computationally intensive.

4.1   Pattern search

Pattern search [26] finds a sequence of parameter vectors $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots$ that approach an optimal vector $\mathbf{x}_n$. The value of the objective function for each subsequent vector in the sequence either decreases or remains the same, i.e.

$$\forall i \in \{0, 1, 2, \ldots, n-1\}[f(\mathbf{x}_i) \geq f(\mathbf{x}_{i+1})]$$

Pattern search starts at the predefined initial vector $\mathbf{x}_0$ and the algorithm can be described as follows:

1. Generate a pattern of vectors P_i centralized around the current vector. This pattern is generated by adding and subtracting a value of the mesh size parameter $\mathrm{ms}_i(\mathrm{ms}_0 = 1)$ in each dimension of the current vector. For example, in a two dimensional space the pattern for the initial vector contains the following four vectors:

$$P_0 = \{(x_{0x}, x_{0y} + \mathrm{ms}_0), (x_{0x} + \mathrm{ms}_0, x_{0y}), (x_{0x}, x_{0y} - \mathrm{ms}_0), (x_{0x} - \mathrm{ms}_0, x_{0y})\} \tag{6}$$

2. Evaluate the objective function for each vector in the generated pattern.

3. If the value of the objective function for any vector in the pattern is lower than the value for the current vector, then the following happens:

   (a) The vector with the lowest objective function value becomes the new current vector for next iteration:

   $$\mathbf{x}_{i+1} = \mathbf{p} \in P_i : \forall \mathbf{q} \in P_i[f(\mathbf{p}) \leq f(\mathbf{q})]$$

   (b) The value of the mesh size parameter for the next iteration doubles:

   $$\mathrm{ms}_{i+1} = 2 \cdot \mathrm{ms}_i$$

   (c) The next iteration starts from step 1 of this algorithm.

4. If the value of the objective function for the current vector is lower than the values for all pattern vectors, then the following happens:

   (a) The value of the mesh size parameter for the next iteration is halved:

   $$\mathrm{ms}_{i+1} = \frac{\mathrm{ms}_i}{2}$$

   (b) If the mesh size for the next iteration is lower than a specified threshold, the algorithm stops and returns the current vector as the optimal point. Otherwise, the current vector is retained for the next iteration:

   $$\mathbf{x}_{i+1} = \mathbf{x}_i$$

   The next iteration starts from the step 1.

4.2 Global search

Global search [27] searches for a global minimum by using the Interior point method to find the local minima for multiple input vectors. First, a set of input vectors S is generated in a constrained region by a scatter search algorithm [15] and split into two disjoint subsets: small initialization subset S_i and main subset S_m. The global search method processes these subsets in two stages:

1. Initialization of method parameters and thresholds on the initialization subset S_i.

   (a) Evaluate objective function for all vectors in S_i and find the vector with the lowest value of the objective function:
   $$\mathbf{x}_0 = \operatorname*{argmin}_{\mathbf{x} \in S_i}(f(\mathbf{x}))$$

   (b) Use the Interior point method to find a local minimum for the found vector:
   $$\mathbf{locMin}_0 = \operatorname{mincon}(f, \mathbf{x}_0)$$

   (c) Set a score threshold to the objective function value of the found local minimum:
   $$\text{threshold} = f(\mathbf{locMin}_0)$$

   (d) Create a spherical basin of attraction defined by the found minimum as its center:
   $$\mathbf{center}_0 = \mathbf{locMin}_0;$$
   and Euclidean distance between the found vector and its minimum as its radius:
   $$\text{radius}_0 = |\mathbf{locMin}_0 - \mathbf{x}_0|$$

2. The main iterative loop executed sequentially for each vector $\mathbf{x}$ in the main subset S_m.

   (a) If $\mathbf{x}$ is not in any basin of attraction:
   $$\forall i \in \{0, 1, \dots, m\}[|\mathbf{x} - \mathbf{center}_i| > \text{distanceFactor} \cdot \text{radius}_i]$$
   (m is a number of basins of attraction and distanceFactor is a settable parameter of the method) and the objective function value for $\mathbf{x}$ is less than the score threshold:
   $$f(\mathbf{x}) < \text{threshold},$$
   then:
   - Change the score threshold value to the objective function value for $\mathbf{x}$:
   $$\text{threshold} = f(\mathbf{x})$$

- Use the Interior point method to find a local minimum for $\mathbf{x}$:

$$\mathbf{locMin}_x = \mathrm{mincon}(f, \mathbf{x})$$

- If the found local minimum was already identified in the previous iterations, update the radius for the appropriate basin of attraction:

$$\mathrm{radius}_i = \max(\mathrm{radius}_i, |\mathbf{x} - \mathbf{center}_i|);$$

otherwise save the newly found local minimum into the results and create a new basin of attraction around it:

$$\mathbf{center}_{m+1} = \mathbf{locMin}_x$$

$$\mathrm{radius}_{m+1} = |\mathbf{locMin}_x - \mathbf{x}|.$$

(b) Otherwise if the evaluated vectors belonged to one basin of attraction for a number of consecutive iterations (specified by waitCycle parameter), then this basin is scaled down by multiplying its radius by $(1 - \mathrm{radiusFactor})$. If the objective function values of the evaluated vectors were higher than the score threshold for a number of consecutive iterations, then the threshold is increased to enable more successful evaluations:

$$\mathrm{threshold} \longleftarrow \mathrm{threshold} + \mathrm{thresholdFactor} \cdot (1 + |\mathrm{threshold}|),$$

where waitCycle, radiusFactor and thresholdFactor are settable parameters of the method.

This algorithm finishes after it runs out of vectors in the main set or after a specified period of execution time. All resulting local minima are then compared to each other and the lowest of them is provided as the global minimum.

4.3 Multi-start Interior point method

This global method uses the Interior point method as well but searches for the local minima of the objective function for multiple independent, randomly generated input vectors in the constrained region. The lowest of the found local minima is provided as the global minimum as in the previous method.

One of the advantages of this method is that all generated input vectors are independent and can be deployed to a computer cluster and evaluated in parallel. This can significantly decrease the execution time needed to find the minimum.

4.4 Genetic algorithm

The genetic algorithm looks for a global minimum by changing and combining vectors from initial randomly generated population P_0 in the constrained region. Each iteration of the algorithm's main loop takes the current population P_i and performs the following steps:

1. Evaluate the objective function for all vectors in P_i and select a multiset of parents S_i. This selection is done by the stochastic uniform sampling algorithm [2].

2. Create an elite multiset E_i containing the eliteCount number of elite vectors from P_i, i.e. vectors with the best objective function values. The vectors in this set will be passed to the next generation population without any changes.

3. Create a crossover multiset C_i containing $\lfloor(|P_i|-\text{eliteCount})\cdot\text{crossoverFraction}\rfloor$ number of crossover vectors created from the set of parents S_i. Each crossover vector is created by combining the vectors of two randomly selected parents. This combination is done by the uniform crossover method, i.e. each vector element is taken from one of the parents randomly.

4. Create a mutation multiset M_i containing $|P_i| - \text{eliteCount} - |C_i|$ number of mutated vectors created from the set of parents S_i. Each mutated vector is created from a random parent by adding a random number from the Gaussian distribution $N(0, \sigma_i)$ to each vector element. The standard deviation starts at a value of the settable parameter scale:

$$\sigma_0 = \text{scale}$$

and decreases with each generation by a settable parameter shrink:

$$\sigma_{i+1} = \sigma_i \cdot (1 - \text{shrink} \cdot \frac{i+1}{\text{generationCount}})$$

5. Create a population for the next iteration P_{i+1} by combining the elite, crossover, and mutation multisets:

$$P_{i+1} = E_i + C_i + M_i$$

This method stops after a generationCount number of iterations have been processed, after a specified period of execution time, or if the best objective function value is less than or equal to the value of fitnessValue.

This algorithm can be partially parallelized by computing the objective function for each member of the population concurrently. After all objective functions are computed, the algorithm has to gather them all and use them to create another generation. The creation of individual members of the next generation can also be done concurrently. Therefore, problems with computationally intensive evaluation of their objective function or those with big populations, in particular, will benefit from the parallelization of this method.

5. Numerical experiments

In this section, the methodology used in the calibration experiments and software and hardware environment for executing these experiments will be described along with the discussion of the results.

5.1 Runtime environment

Numerical experiments for evaluating the optimization methods were implemented and performed in programming language and computing environment MathWorks MATLAB R2013a [19]. Because they are computationally intensive, we used the MATLAB Parallel Computing Toolbox, MATLAB Distributed Computing Server, and MATLAB built-in parallel functions to improve the code performance by speeding up executions. The *fmincon* solver using the *interior-point* algorithm, and own implementation of the line search algorithm using the *fminbnd* minimizer for a step optimization were utilized in the local optimization methods. Similarly, *patternsearch*, *GlobalSearch* using the *interior-point* algorithm, *MultiStart* using the *interior-point* algorithm, and *ga* (genetic algorithm) were utilized for the global optimization methods. The experiments were executed on the computing nodes of the Anselm cluster at IT4Innovations National Supercomputing Center. Each node of the cluster is a powerful x86-64 computer, equipped with two 8-core Intel Sandy Bridge E5-2665 2.4 GHz processors and 64 GBRAM. The nodes are interconnected by fully non-blocking fat-tree Infiniband network (3600 MB/s).

256 MATLAB workers using 256 processor cores (16 nodes with 16 cores) were used for running each calculation (job). Each job processed one of the four modelled catchments in the Moravian-Silesian region (Odra, Opava, Ostravice, Olše).

5.2 Episodes for evaluation

The Math1D model is usually used for 168 hour episodes which consist of 120 hours of past simulation based on measured data and 48 hours of prediction. Since the purpose of our test was not the accuracy of the model prediction but the improvement by calibration, we decided to use 168 hour simulations based on measured data to minimize the uncertainty of weather predictions.

As for the selection of episodes, we analysed the model results for the whole year 2010 to test the calibration on a wide range of rainfall events including heavy rainfall that caused a significant flooding episode in the Moravian-Silesian region in May. We only picked events with the peak discharge value higher than a median of year discharges. It was also necessary to choose such events when the discharge is not heavily affected by precipitations right before the start of the simulation period. Only periods with complete precipitation and discharge data sets were used in this experiment as we wanted to minimize the error produced by interpolation or extrapolation of the data.

5.3 Model accuracy evaluation

Six different combinations of norms were used to evaluate the accuracy of our model. At first, we used one of the three following norms to measure the error of modelled hydrographs at i-th measurement station. Nash-Sutcliffe coefficient is widely used and simple statistics for assessing the goodness of fit of hydrologic models. This coefficient uses observed and modelled values for assessing the model efficiency. The Nash-Sutcliffe coefficient is sensitive to extreme values (peak flow), and therefore it is important to focus on the culmination part of the hydrograph (normed peak error). Another important characteristic of the flood wave is volume

and thus we focus on it as well evaluating the normed volume error. Overall, the results from the Nash-Sutcliffe coefficient show complex evaluation of the calibration process, whereas peak and volume error are determined to monitor only a particular part of the calibrated hydrograph.

By $\overline{\mathbf{Q}_{\mathrm{o}}^{\mathrm{i}}} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{Q}_{\mathrm{o}}^{\mathrm{i,t}}$, we denote mean of observed discharges over time steps t, $\mathbf{Q}_{\mathrm{m}}^{\mathrm{i,t}}$ denotes modelled discharge at time step t, and $\mathbf{Q}_{\mathrm{o}}^{\mathrm{i,t}}$ observed discharge at time step t.

- Adapted Nash-Sutcliffe model efficiency coefficient:

$$\mathrm{err}_i^A = \frac{\sum_{t=1}^{T} \left(\mathbf{Q}_{\mathrm{o}}^{\mathrm{i,t}} - \mathbf{Q}_{\mathrm{m}}^{\mathrm{i,t}}\right)^2}{\sum_{t=1}^{T} \left(\mathbf{Q}_{\mathrm{o}}^{\mathrm{i,t}} - \overline{\mathbf{Q}_{\mathrm{o}}^{\mathrm{i}}}\right)^2}.$$

  The original Nash-Sutcliffe coefficient is simply given by $\mathrm{NSC} = 1 - \mathrm{err}_i^A$. The modification was used purely for easier expression of the cost function for the minimization methods.

- Normed peak error:

$$\mathrm{err}_i^B = \frac{\left|\mathbf{Q}_{\mathrm{o}}^{\mathrm{i,t}_p^i} - \mathbf{Q}_{\mathrm{m}}^{\mathrm{i,t}_p^i}\right|}{\mathbf{Q}_{\mathrm{o}}^{\mathrm{i,t}_p^i}}, \ \ \text{where } t_p^i = \argmax_{t \in \{1,...,T\}} \left(\mathbf{Q}_{\mathrm{o}}^{\mathrm{i,t}}\right).$$

- Normed volume error:

$$\mathrm{err}_i^C = \frac{\sum_{t=1}^{T} \left|\mathbf{Q}_{\mathrm{o}}^{\mathrm{i,t}} - \mathbf{Q}_{\mathrm{m}}^{\mathrm{i,t}}\right|}{\sum_{t=1}^{T} \left|\mathbf{Q}_{\mathrm{o}}^{\mathrm{i,t}}\right|}.$$

In the second phase, we computed the overall error using L^2-norm and L^∞-norm of all described error norms $\mathrm{err}^{\mathbf{X}}$ for each measurement station. N is a number of measurement stations in the modelled catchment.

$$\left\|\mathbf{err}^{\mathbf{X}}\right\|_2 = \sqrt{\sum_{i=1}^{N} \left|\mathrm{err}_i^X\right|^2}, \qquad \left\|\mathbf{err}^{\mathbf{X}}\right\|_\infty = \max_{i \in \{1,...,N\}} \mathrm{err}_i^X.$$

5.4   Schematization processing strategies

Two types of strategies describing the calibration procedure based on the structure of modelled catchments were used for the numerical experiments. Each catchment is partitioned to many sub-basins as specified by the semi-distributed approach used in the experimental model. The way how sub-basins are calibrated differs for these two strategies.

Overall strategy.   In this approach, the parameters for the whole basin are calibrated at once, and errors of modelled hydrographs are computed for each discharge measurement station. The overall error is evaluated using one of the two norms described in the second phase of the model accuracy evaluation (L^2-norm and L^∞-norm). The principle of this overall strategy is shown in Fig. 2a.
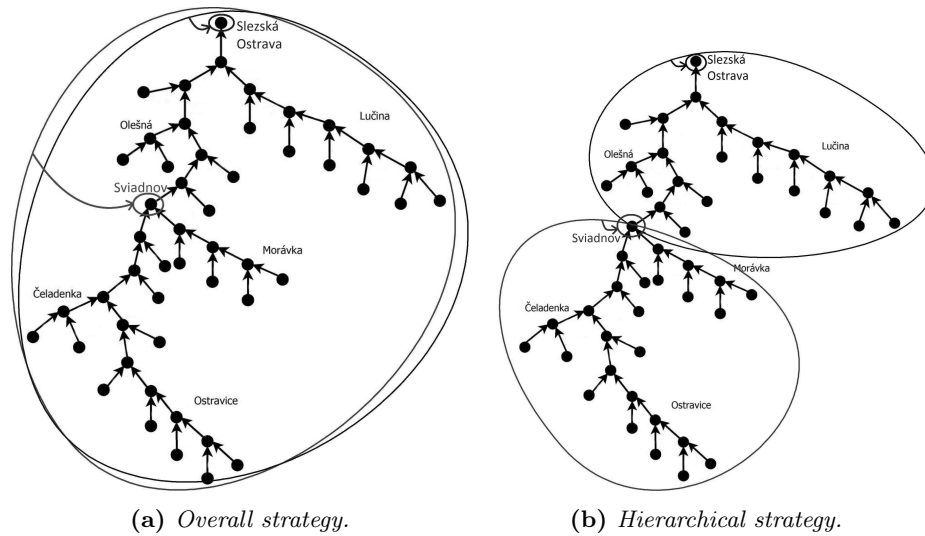
(a) *Overall strategy.* **(b)** *Hierarchical strategy.*

Fig. 2 *Schematic diagrams for schematization processing strategies.*

Hierarchical strategy.   This approach takes into account the natural hierarchical behaviour of all catchments, i.e. upstream parts of the catchment that contribute to an upstream measurement gauge should be calibrated for this gauge and should not be influenced by downstream gauges. Therefore, the parameters of the sub-basins that are closest to the river springs are calibrated first and their calibrated parameters are then taken as fixed while calibrating the downstream parts of the catchment. The errors for individual measurement gages and the overall error are computed using the same norms as in the overall strategy. The principle of the hierarchical strategy shows Fig. 2b.

5.5   Testing parameters

All minimization methods were constrained to finish computation in some reasonable time under one hour. Both local minimization algorithms were bounded by a number of evaluations of the cost function. The global minimization methods were bounded by time limit of 1800 seconds.

5.6   Comparison of optimization methods

We compared 6 methods on 32 selected episodes in 6 combinations of norms and two calibration approaches and evaluated the improvement of the original modelled discharge error without calibration.

$$\text{improvement} = 1 - \frac{\text{error after calibration}}{\text{error before calibration}}$$

If the error is reduced to zero, the improvement is 1. If the error remains unchanged, the improvement is zero. Improvement of one half means that the error after calibration is one half of the original error.

5.7   Results

Adapted Nash-Sutcliffe model efficiency coefficient.   The medians of both local optimization methods are comparable (Tab. I and II) but the standard deviation of improvement of the Line search method is twice as high as the improvement for the Interior point method. It shows that the Line search method sometimes fails to provide any better improvement. This is even more clear if we consider the L^∞-norm (Fig. 4a). The hierarchical approach provides slightly better results for almost all methods (except for Multi-start). The best results for both approaches are provided by the Pattern search algorithm.

Normed peak error.   Tabs. III, IV and Fig. 4b show that the best improvement can be achieved in the normed peak error. This is caused by the fact that the error is measured only in one point. It turns out that the Pattern search method and Genetic algorithm can easily find the parameters to fit the value at the peak. This, of course, does not mean that the modelled hydrograph is in good agreement with the observed hydrograph at the rest of the episode (see Fig. 3). We can also observe from Figs. 6c, 6d, 5c, and 5d that there is a positive correlation between the mean of observed discharge and improvement of error for nearly all methods.

Normed volume error.   Calibration for the normed volume error provides smallest improvement but this does not necessarily mean that the absolute results are the worst. Local methods and the Pattern search provide the best improvement (Fig. 4c and Tabs. V and VI). The hierarchical approach is slightly better for almost all methods. Figs. 5e, 5f, 6e and 6f show that the improvement tends to be smaller as the mean of observed discharge rises.

The Pattern search algorithm provides the best improvement in all combinations of the norms. As for the approaches, the hierarchical approach slightly dominates the overall one.

	hierarchical approach			overall approach		
	mean	median	std	mean	median	std
Local methods						
Linesearch	0.811	0.855	0.183	0.787	0.848	0.230
Interior point	0.852	0.864	0.099	0.843	0.851	0.104
Global methods						
Pattern search	**0.892**	**0.907**	0.080	0.878	0.867	0.085
Global search	0.802	0.840	0.159	0.821	0.818	0.112
Multistart	0.677	0.769	0.304	0.743	0.795	0.235
Genetic alg.	0.768	0.772	0.132	0.739	0.758	0.170

**Tab. I** *Statistical results for improvement of* $\left\|\mathbf{err^A}\right\|_2$.
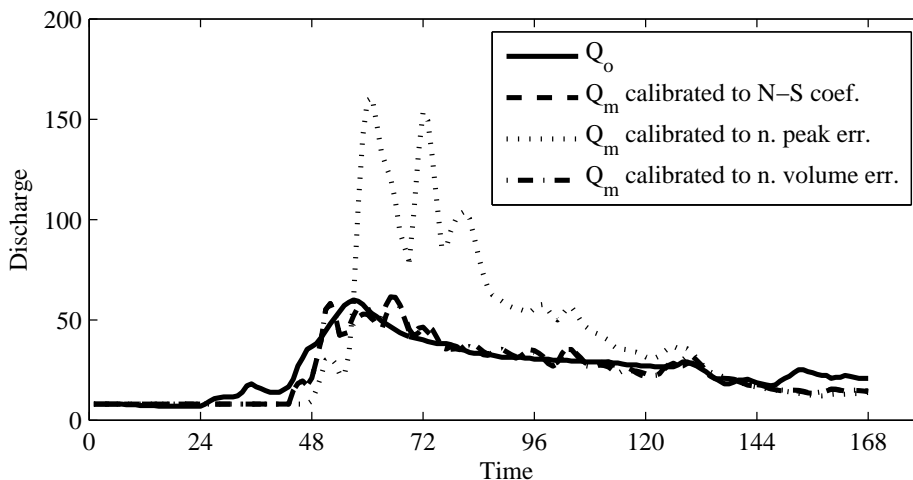
Fig. 3 *Comparison of modelled discharge calibrated in the N-S norm, the peak error norm, and the volume error norm. Note that the N-S norm and the volume error norm are overlapping almost perfectly.*

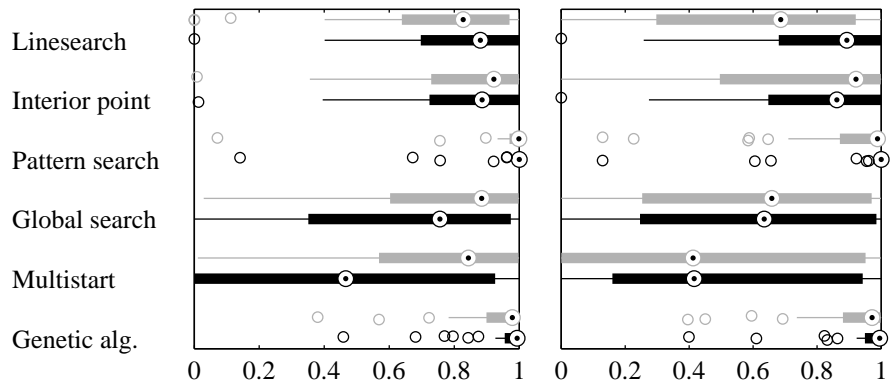|	hierarchical approach			overall approach		
	mean	median	std	mean	median	std
Local methods						
Linesearch	0.796	0.848	0.215	0.712	0.821	0.324
Interior point	0.848	0.860	0.105	0.838	0.847	0.111
Global methods						
Pattern search	**0.882**	**0.904**	0.097	0.877	0.875	0.097
Global search	0.806	0.826	0.145	0.797	0.829	0.160
Multistart	0.637	0.776	0.337	0.756	0.802	0.205
Genetic alg.	0.751	0.755	0.152	0.724	0.752	0.182

Tab. II *Statistical results for improvement of $\left\|\mathbf{err^A}\right\|_\infty$.*

5.8 Parallel scalability

We performed several tests of parallel scalability of the calibration method for episodes with medium (Tab. VII) and heavy rain (Tab. VIII). The stopping criterion was set to the 50 % reduction of the initial error of non-calibrated model. This criterion was based on the previous experience with maximal possible error reduction of the tested calibration methods. All calibration methods were able to achieve such reduction. As expected, the performance of Line search and Pattern search algorithm is not affected by the number of cores provided for the parallelization. The results of the Global Search and Multi-start algorithm are almost identical since both method use the Interior point as the local solver. We expected better
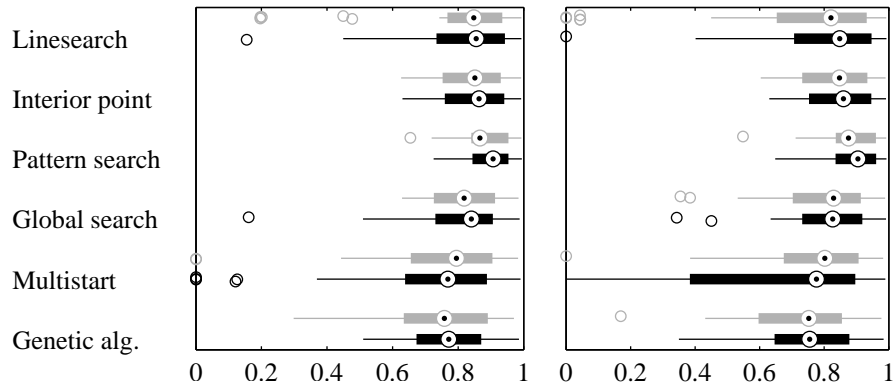
(a) *Relative improvement of* $\left\|\mathbf{err^A}\right\|_2$ *(left) and* $\left\|\mathbf{err^A}\right\|_\infty$ *(right).*

(b) *Relative improvement of* $\left\|\mathbf{err^B}\right\|_2$ *(left) and* $\left\|\mathbf{err^B}\right\|_\infty$ *(right).*
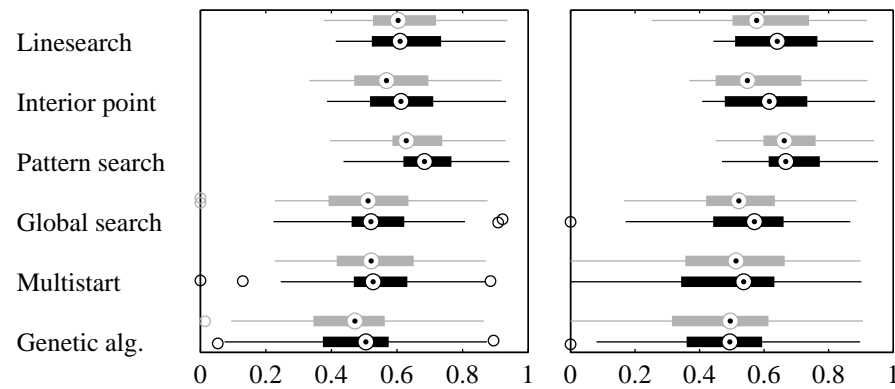
(c) *Relative improvement of* $\left\|\mathbf{err^C}\right\|_2$ *(left) and* $\left\|\mathbf{err^C}\right\|_\infty$ *(right).*

Fig. 4 *Relative improvements of error in* L^2*-norm and* L^∞*-norm for overall (grey) and hierarchical (black) approach.*

(a) *Relative improvement of* $\left\|\mathbf{err^A}\right\|_2$.

(b) *Relative improvement of* $\left\|\mathbf{err^A}\right\|_\infty$.

(c) *Relative improvement of* $\left\|\mathbf{err^B}\right\|_2$.

(d) *Relative improvement of* $\left\|\mathbf{err^B}\right\|_\infty$.

(e) *Relative improvement of* $\left\|\mathbf{err^C}\right\|_2$.

(f) *Relative improvement of* $\left\|\mathbf{err^C}\right\|_\infty$.
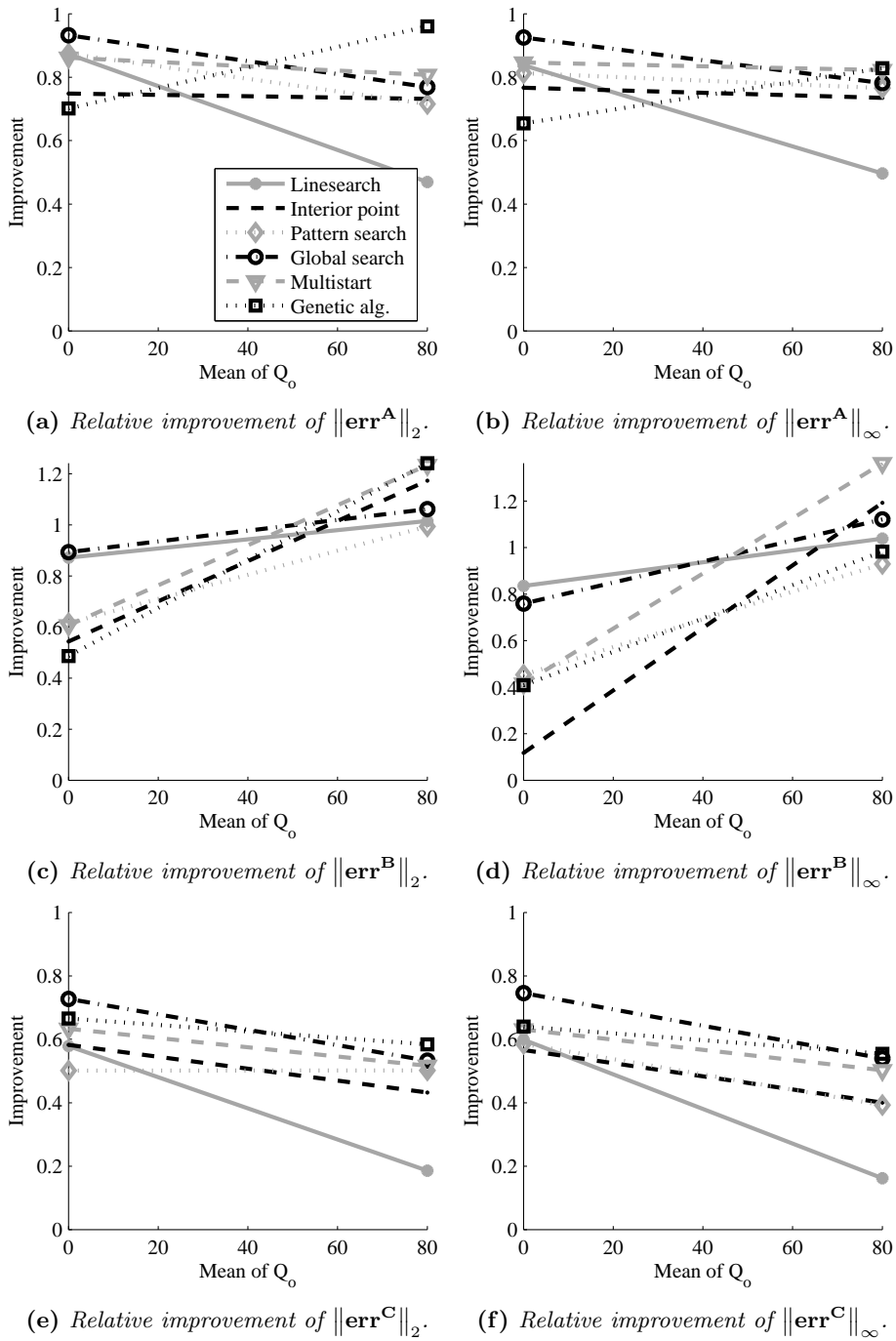
Fig. 5 *Correlations between mean of measured discharge and improvement of error for overall approach.*

(a) *Relative improvement of* $\left\| \mathbf{err^A} \right\|_2$.

(b) *Relative improvement of* $\left\| \mathbf{err^A} \right\|_\infty$.

(c) *Relative improvement of* $\left\| \mathbf{err^B} \right\|_2$.

(d) *Relative improvement of* $\left\| \mathbf{err^B} \right\|_\infty$.

(e) *Relative improvement of* $\left\| \mathbf{err^C} \right\|_2$.

(f) *Relative improvement of* $\left\| \mathbf{err^C} \right\|_\infty$.
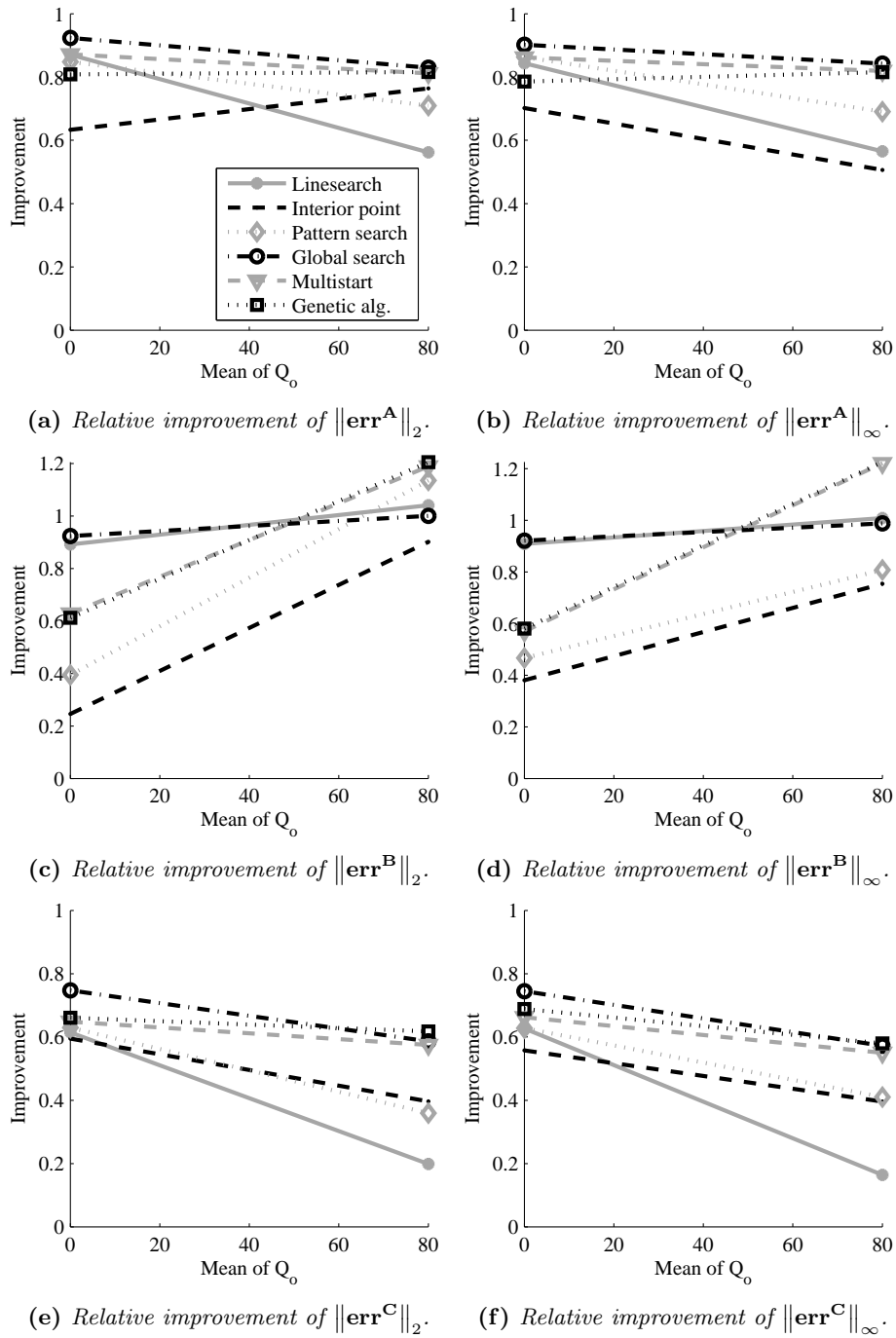
Fig. 6 *Correlations between mean of measured discharge and improvement of error for hierarchical approach.*

	hierarchical approach			overall approach		
	mean	median	std	mean	median	std
Local methods						
Linesearch	0.809	0.880	0.238	0.737	0.828	0.291
Interior point	0.814	0.886	0.233	0.815	0.922	0.244
Global methods						
Pattern search	**0.949**	**1.000**	0.164	0.949	0.999	0.167
Global search	0.640	0.756	0.348	0.741	0.884	0.289
Multistart	0.464	0.466	0.423	0.752	0.843	0.269
Genetic alg.	0.941	0.994	0.118	0.920	0.979	0.140

Tab. III *Statistical results for improvement of* $\left\|\mathbf{err^B}\right\|_2$.

	hierarchical approach			overall approach		
	mean	median	std	mean	median	std
Local methods						
Linesearch	0.794	0.893	0.260	0.599	0.686	0.366
Interior point	0.786	0.861	0.258	0.730	0.922	0.356
Global methods						
Pattern search	**0.943**	**1.000**	0.174	0.880	0.989	0.223
Global search	0.580	0.634	0.387	0.611	0.658	0.381
Multistart	0.505	0.416	0.380	0.475	0.412	0.422
Genetic alg.	0.942	0.996	0.127	0.903	0.972	0.159

Tab. IV *Statistical results for improvement of* $\left\|\mathbf{err^B}\right\|_\infty$.

	hierarchical approach			overall approach		
	mean	median	std	mean	median	std
Local methods						
Linesearch	0.646	0.610	0.153	0.639	0.603	0.147
Interior point	0.624	0.612	0.145	0.595	0.568	0.157
Global methods						
Pattern search	**0.694**	**0.684**	0.115	0.662	0.628	0.132
Global search	0.540	0.520	0.164	0.502	0.512	0.215
Multistart	0.529	0.527	0.184	0.533	0.521	0.159
Genetic alg.	0.476	0.505	0.185	0.448	0.472	0.192

Tab. V *Statistical results for improvement of* $\left\|\mathbf{err^C}\right\|_2$.

	hierarchical approach			overall approach		
	mean	median	std	mean	median	std
Local methods						
Linesearch	0.652	0.641	0.156	0.612	0.577	0.161
Interior point	0.625	0.616	0.153	0.588	0.548	0.166
Global methods						
Pattern search	**0.687**	**0.667**	0.124	0.677	0.662	0.131
Global search	0.556	0.570	0.186	0.522	0.522	0.165
Multistart	0.504	0.537	0.226	0.511	0.513	0.196
Genetic alg.	0.473	0.494	0.204	0.453	0.495	0.223

Tab. VI *Statistical results for improvement of* $\left\|\mathbf{err}^{\mathbf{C}}\right\|_\infty$.

speedup of the Multi-start algorithm, but the optimization reached desired improvement within the run of one local solver so the parallelization of the global aspect of the Multi-start algorithm did not manifest. The Genetic algorithm is the fastest used algorithm for higher number of cores, but methods based on the Interior point algorithm seem to have the best parallel scalability. The scalability graphs of the tested algorithms are shown in Figs. 7 and 8.

	Number of cores				
	1	2	4	8	16
Linesearch	**27**	**27**	**27**	27	27
Interior point	190	95	53	29	18
Pattern search	62	62	61	61	61
Global search	303	153	84	46	30
Multistart	296	155	84	45	29
Genetic alg.	195	71	31	**17**	**16**

Tab. VII *Times in sec of calibration for episode with medium rain.*

6. Conclusion and open problems

The main goal of the paper was to evaluate several approaches to automated calibration of rainfall-runoff modelling. Two local minimization methods (line search and interior point methods) and four global minimum searching algorithms (pattern search, global search, multistart method and genetic algorithm) were tested on two schematization processing strategies. The error improvement of the resulting model was chosen as the main criteria of this evaluation. For the error measurement, three different generally used norms were used: Nash-Sutcliffe measurement model, normed peak error, and normed volume error. As a very important constraint on

	Number of cores				
	1	2	4	8	16
Linesearch	**44**	43	43	43	43
Interior point	207	106	58	31	20
Pattern search	397	396	397	397	397
Global search	285	145	81	44	27
Multistart	284	147	81	44	27
Genetic alg.	106	**28**	**15**	**17**	**8**

Tab. VIII *Times in sec of calibration for episode with heavy rain.*
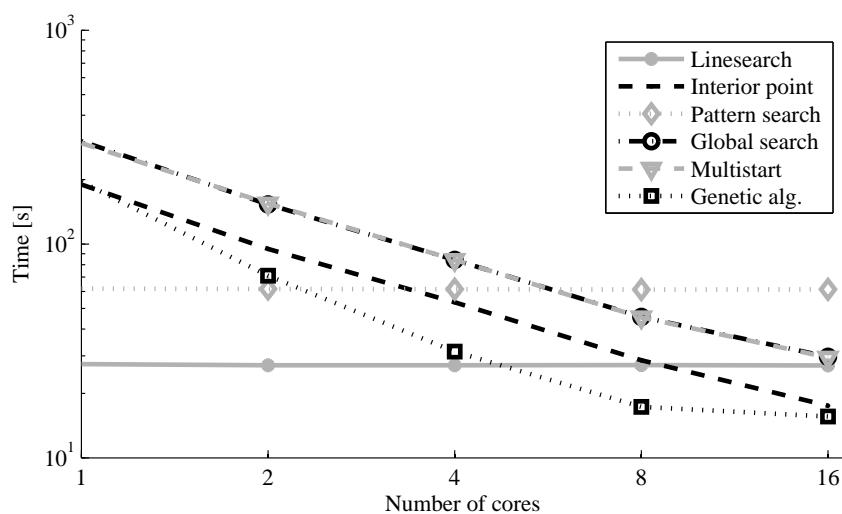
Fig. 7 *Speedup of calibration for episode with medium rain.*

the tested calibration methods, we have used 30 min time limit that is natural requirement of the Floreon$^+$ system which serves an operational flood forecast every one hour. Our numerical experiments show that the hierarchical schematization processing strategy gives more accurate results. In terms of error improvement, the pattern search global optimization method shows the best improvement for all tested error norms. From the other side, both the hierarchical strategy and pattern search method are the most time consuming techniques. Testing the time performance and scalability of the optimization algorithms gives another view on the calibration efficiency. The line search local optimization method and genetic algorithm as a global optimization method give the 50 % error improvement in the shortest times and the genetic algorithm also benefits from very good parallel scalability up to 16 CPU cores.

Our future work will focus on massively parallel implementation and testing of optimization algorithms as well as calibration on large sets of discharge measuring
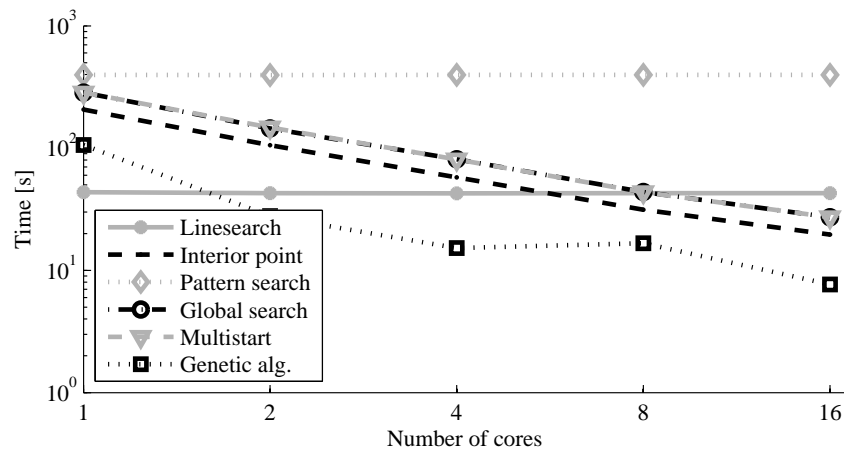
Fig. 8 *Speedup of calibration for episode with heavy rain.*

stations. Significant parallel speedup of less accurate global algorithms can give them more time to improve their accuracy and thus overtake the most accurate pattern search method with the worst parallel scalability. According to criteria designed for specific Floreon[+] requirements, the selected calibration methods and approaches will be fully implemented using the C/C++ language in the parallel software package for the automatic calibration of rainfall-runoff models. After verification of the implemented methods, the software will be deployed to the operation process of the Floreon[+] system.

Acknowledgement

References

[1] ABBOTT M.B., REFSGAARD J.C. Distributed Hydrological Modelling. Dordrecht: Kluwer Academic. 1996, doi: 10.1007/978-94-009-0257-2.

[2] BAKER J.E. Reducing Bias and Inefficiency in the Selection Algorithm. Proceedings of the Second International Conference on Genetic Algorithms and their Application. Hillsdale, New Jersey: L. Erlbaum Associates. 1987, pp. 14–21.

[3] BERAN M. Hydrograph prediction: How much skill? Hydrology and Earth System Sciences. 1999, 3, pp. 305–307, doi: 10.5194/hess-3-305-1999.

[4] BEVEN K.J. A manifesto for the equifinality thesis. Journal of Hydrology. 2006, 320, pp. 18–36, doi: 10.1016/j.jhydrol.2005.07.007.

[5] BEVEN K.J. On doing better hydrological science. Hydrological Processes. 2008, 22, pp. 3549–3553, doi: 10.1002/hyp.7108.

[6] BEVEN K.J. Preferential flows and travel time distributions: Defining adequate hypothesis tests for hydrological process models. Hydrological Processes. 2010, 24, pp. 1537–1547, doi: 10.1002/hyp.7718.

[7] BEVEN K.J. Rainfall-Runoff Modelling. The Primer. Second Edition. Wiley-Blackwell, 2012, doi: 10.1002/9781119951001.

[8] BEVEN K.J., BINLEY A.M. The future of distributed models: Model calibration and uncertainty prediction. Hydrological Processes. 1992, 6(3), pp. 279–298, doi: 10.1002/hyp.3360060305.

[9] BYRD R.H., HRIBAR M.E., NOCEDAL J. An Interior Point Algorithm for Large-Scale Nonlinear Programming, *SIAM Journal on Optimization*. 1999, 9(4), pp. 877–900, doi: 10.1137/s1052623497325107.

[10] BYRD R.H., GILBERT J.C., NOCEDAL J. A Trust Region Method Based on Interior Point Techniques for Nonlinear Programming, *Mathematical Programming*. 2000, 89(1), pp. 149-185, doi: 10.1007/pl00011391.

[11] CLARKE R.T.: Mathematical models in hydrology. Irrigation and Drainage paper No. 19, FAO, Rome, 1973.

[12] DUAN Q., SOROOSHIAN S., GUPTA V. Effective and efficient global optimization for conceptual rainfall-runoff models. Water Resour. Res. 1992, 28(4), pp. 1015–1031, doi: 10.1029/91wr02985.

[13] FEYEN L., VRUGT J.A., O'NUALLAIN B., VAN DER KNIJFF J., DE ROO A. Parameter optimisation and uncertainty assessment for large-scale streamflow simulation with the LISFLOOD model. Journal of Hydrology. 2007, 332, pp. 276–289, doi: 10.1016/j.jhydrol.2006.07.004.

[14] FLEMING G. Deterministic models in hydrology. Irrigation and Drainage paper, 32, FAO, Rome, 1979, 80 s.

[15] LAGUNA M., MARTI R. Scatter Search: Methodology and Implementations in C. Kluwer Academic Publishers, 2003.

[16] MADSEN H. Automatic calibration of conceptual rainfall-runoff model using multiple objectives. Journal of Hydrology. 2000, 235/3-4, pp. 276–288, doi: 10.1016/s0022-1694(00)00279-1.

[17] MADSEN H., WILSON G., AMMENTORP H.C. Comparison of different automated strategies for calibration of rainfall-runoff models. Journal of Hydrology. 2002, 261/1-4, pp. 48–59, doi: 10.1016/s0022-1694(01)00619-9.

[18] MARTINOVIČ J., KUCHAŘ Š., VONDRÁK I., VONDRÁK V., ŠÍR B., UNUCKA J. Multiple Scenarios Computing In The Flood Prediction System FLOREON. 24th European Conference on Modelling and Simulation. 2010, pp. 182–188, doi: 10.7148/2010-0182-0188.

[19] MATLAB and Statistics Toolbox Release 2013a, The MathWorks, Inc., Natick, Massachusetts, United States.

[20] MORADKHANI H., SOROOSHIAN S., GUPTA H.V., HOUSER P.R. Dual state-parameter estimation of hydrological models using ensemble Kalman filter. Advanced Water Resource. 2005a, 28(2), pp. 135–147, doi: 10.1016/j.advwatres.2004.09.002.

[21] MORADKHANI H., SOROOSHIAN S. General Review of Rainfall-Runoff Modeling: Model Calibration, Data Assimilation and Uncertainty Analysis. In: Sorooshian, S. (Ed.) Hydrological Modelling and the Water Cycle. Coupling the Atmospheric and Hydrological Models. Springer, 2008, doi: 10.1007/978-3-540-77843-1_1.

[22] SEIBERT J., MCDONNELL J.J. On the dialog between experimentalist and modeler in catchment hydrology: Use of soft data for multicriteria model calibration. Water Resources Research. 2002, 38, pp. W01241, doi: 10.1029/2001WR000978.

[23] SOROOSHIAN S., GUPTA V.K. Model calibration. In: Singh, V. P. (Ed.), Computer Models of Watershed Hydrology, Water Resources Publications. Highlands Ranch, 1995, CO, pp. 23–67.

[24] SOROOSHIAN, S. (Ed.) Hydrological Modelling and the Water Cycle. Coupling the Atmospheric and Hydrological Models. Springer, 2008, doi: 10.1007/978-3-540-77843-1.

[25] THIEMANN M., TROSSET M., GUPTA H., SOROOSHIAN S. Bayesian recursive parameter estimation for hydrologic models. Water Resources Research. 2001, 37(10), pp. 2521–2535, doi: 10.1029/2000wr900405.

[26] TORCZON V. On the convergence of pattern search algorithms. SIAM Journal on Optimization. 1997, 7, pp. 1–25, doi: 10.1137/s1052623493250780.

[27] UGRAY Z., LASDON L., PLUMMER J., GLOVER F., KELLY J., MARTI R. Scatter Search and Local NLP Solvers: A Multistart Framework for Global Optimization. INFORMS Journal on Computing. 2007, 19(3), pp. 328–340, doi: 10.1287/ijoc.1060.0175.

[28] VACH K.B., MCDONNELL J.J. A process-based rejectionist framework for evaluating catchment runoff model structure. Water Resources Research. 2006, 42, pp. W02409, doi: 10.1029/2005WR004247.

[29] VIEUX B.E. Distributed Hydrologic Modeling using GIS. 2nd Edition. Springer Netherlands, 2004, doi: 10.1007/1-4020-2460-6.

[30] VONDRÁK I., MARTINOVIČ J., KOŽUSZNIK J., ŠTOLFA S., KOZUBEK T., KUBÍČEK P., VONDRÁK V., UNUCKA J. A Description of a Highly Modular System for the Emergent Flood Prediction. 7th Computer Information Systems and Industrial Management Applications. 2008, pp. 219–224, doi: 10.1109/cisim.2008.22.

[31] VRUGT J. A., GUPTA H.V., BOUTEN W., SOROOSHIAN S. A Shuffled Complex Evolution Metropolis algorithm for optimization and uncertainty assessment of hydrologic model parameters. Water Resources Research. 2003, 39(8), 1201, doi: 10.1029/2002WR001642.