Function Space Optimization: A symbolic regression method for estimating parameter transfer functions for hydrological models

M. Feigl[1], M. Herrnegger[1], D. Klotz[1,2] and K. Schulz[1]

[1]Institute for Hydrology and Water Management, University of Natural Resources and Life Sciences, Vienna, Austria

[2]LIT AI Lab & Institute for Machine Learning, Johannes Kepler University Linz, Linz, Austria

Corresponding author: Moritz Feigl (moritz.feigl@boku.ac.at)

Key Points:

- This study introduces a method to infer transfer functions for the multiscale parameter regionalization approach.
- We demonstrate its ability in a case study using synthetic runoff data.
- We show that multi-criteria optimization can improve the estimation of parameter transfer functions.

Abstract

Estimating parameters for distributed hydrological models is a challenging and long studied task. Parameter transfer functions, which define model parameters as functions of geo-physical properties of a catchment, might improve the calibration procedure, increase process realism and can enable prediction in ungauged areas. We present the Function Space Optimization (FSO), a symbolic regression method for estimating parameter transfer functions for distributed hydrological models. FSO is based on the idea of transferring the search for mathematical expressions into a continuous vector space that can be used for optimization. This is accomplished by using a text generating neural network with a variational autoencoder architecture, that can learn to compress the information of mathematical functions. To evaluate the performance of FSO, we conducted a case study using a parsimonious hydrological model and synthetic discharge data. The case study consisted of two FSO applications: Single-criteria FSO, where only discharge was used for optimization and multi-criteria FSO, where additional spatiotemporal observations of model states were used for transfer function estimation. The results show that FSO is able to estimate transfer functions correctly or approximate them sufficiently. We observed a reduced fit of the parameter density functions resulting from the inferred transfer functions for less sensitive model parameters. For those it was sufficient to estimate functions resulting in parameter distributions with approximately the same mean parameter values as the real transfer functions. The results of the multi-criteria FSO showed that using multiple spatiotemporal observations for optimization increased the quality of estimation considerably.

Plain Language Summary

Hydrological models are widely used tools for predicting river runoff or other components of the hydrological cycle that are important for the management of water resources. Typically, processes in those models use parameters to characterize the unique aspect of the studied area. Usually, these parameters are optimized to produce a well performing prediction model. This potentially leads to a loss of their physical meaning. Preserving the physical meaning of model parameters can be achieved by defining them with a relationship to properties of the modelled area (soil properties, topography, …). These relationships are given as mathematical equations that compute parameters from a set of geo-physical properties. We here present a method to automatically estimate such equations, called Function Space optimization (FSO). FSO transfers the search for mathematical equations into an optimization problem by using a Neural Network to encode the information of potential equations. We show FSOs ability in a case study using a hydrological model and synthetic runoff data. The results show that FSO is able to approximate the true relationship sufficiently. Furthermore, we show that additional spatial observation data can increase FSO performance.

1 Introduction

Distributed hydrological models are widely used tools to model spatiotemporal processes in catchments. The modelled processes include the simulation of spatially distributed land surface fluxes (e.g. Rakovec et al., 2016), estimating the hydrological response to climate change (e.g. Hattermann et al., 2017; Kay et al., 2015) or hydrological response to land use changes (e.g. Hundecha & Bárdossy, 2004; Wijesekara et al., 2012). In general, process-based distributed hydrological models can be classified in two groups: conceptual models and physically-based models (Devia & Ganasri, 2015). Both depend to some extend on parameter calibration (Beven, 2001; Kirchner, 2006). Thus, in practice both approaches need to be calibrated and demand substantial expertise. Physically based models frequently lack observations necessary to define parameters correctly. In such situations, the uncertain parameters are either treated as physical constants (Clark et al., 2017), i.e. a fixed value for a larger area, or are optimized as well. Both methods most likely result in reduced process realism, while still producing reasonable runoff predictions.

A solution to retain the process realism of hydrological models is to relate landscape properties to hydrologic behaviour (Clark et al., 2016). This can be accomplished by using geo-physical information for defining model parameters. This is however non-trivial. As a matter of fact, Clark et al. (2017) described this as one of the major unsolved challenges in hydrologic parameter estimation. Most recently Blöschl et al. (2019) also mentions the “disentanglement and reduction of structural/parameter/input uncertainty in hydrological models” as one of the twenty-three unsolved problems in hydrology. Defining model parameters using the spatially distributed geo-physical properties of a basin, would: reduce parameter uncertainty, increase process realism and the predictive ability of the model, and allow for runoff prediction in ungauged basins. This challenge is closely related to the idea of regionalization, which can be summarized as the geographical migration of hydrological model structures (Buytaert & Beven, 2009). Due to the problem of parameter equifinality in hydrological models (Beven, 2006), finding a relationship after parameter optimization might result in a weak or false regionalization (Hundecha & Bárdossy, 2004; Kumar et al., 2013; Samaniego et al., 2010). To prevent this, the *simultaneous regionalization* method (Abdulla & Lettenmaier, 1997; Hundecha & Bárdossy, 2004; Parajka et al., 2005) was developed. This method tries to overcome this restriction by defining the relationship a priori in form of a *transfer function* and evaluating it in a set of validation basins.

Samaniego et al. (2010) introduced the multiscale parameter regionalization (MPR) as an extension of simultaneous regionalization. MPR defines the parameters on the scale of geo-physical observations before aggregating them to the model scale, thus including small scale variations in their computed parameters. Instead of defining transfer functions using a regression approach, they define mathematical functions of geo-physical properties of a catchment in MPR. This results in a constrained form of parameter calibration that preserves the physical interpretation of the parameter values and produces seamless parameter fields, i.e. they do not exhibit artificial spatial discontinuities often observed in distributed hydrological models (Samaniego et al., 2017).

These mathematical functions are usually unknown (in many cases we do not even know if they exist in the first place). Hence, the main restriction of MPR today is the selection of suitable parameter transfer functions (Samaniego et al., 2017). Potential candidates for transfer functions for hydrological models could be pedotransfer functions. They relate soil properties to

soil parameters and were already investigated extensively in the past (see for example Van Looy et al., 2017). Besides those, functional relationships between model parameters and geo-physical properties are not well known and we still lack methods that perform adequate estimation. At the same time, we assume that they exist.

Klotz et al. (2017) were the first to investigate a symbolic regression approach to automatically estimate transfer functions. The term symbolic regression refers to methods that search the space of mathematical expressions while minimizing some error metrics, usually based on evolutionary computation (Bongard & Lipson, 2007; Cornforth & Lipson, 2015; Schmidt & Lipson, 2009). By using a simple model and synthetic data, Klotz et al. (2017) showed that it is possible to automatically estimate transfer functions from stream data in a virtual setting.

While the general idea of Klotz et al. seemed to work it had two main difficulties: a bias towards overly simple transfer functions and the need to solve a difficult high dimensional discrete optimization problem. Both problems result from the representation of transfer functions as a discrete vector of a context free grammar (CFG). These limitations will be explained in detail in the methods part of this publication and are a main motivation for this work.

To overcome these limitations the proposed method is based on the interpretation of mathematical functions as text, where each symbol of a function is seen as a "word". Recent developments in Natural Language Processing (NLP) resulted in powerful Artificial Intelligence (AI) architectures which are able to translate (e.g. Srivastava et al., 2018), generate (e.g. Lu et al., 2018) and classify (e.g. Yang et al., 2019) text. While most symbolic regression methods are based on evolutionary algorithms, the method presented here is based on transferring the semantic information of text into a continuous space to adequately define nearness between functions. This is accomplished by a text generating neural network. The advantage we expect is that the search becomes more efficient and unbiased due to the continuous space and its properties. To our knowledge, only Gómez-Bombarelli et al. (2018) investigated an approach with a similar idea where they transferred discrete representations of molecules into a continuous vector representation. The application we present is focused on a relevant problem in the hydrological sciences, nevertheless it could potentially also be applied to other fields where functional relationships have to be derived from observational data.

The search for parameter transfer functions is a complex task, therefore investigating ways to reduce its complexity and further constrain it is desirable. In recent years, multiple publications showed the value of using observations of spatially distributed fluxes and storage components for parameter calibration, additionally to stream data (e.g. Baroni et al., 2019; Demirel et al., 2018; Francke et al., 2018; Huang et al., 2019; Nijzink et al., 2018; Rakovec et al., 2016; Stisen et al., 2011, 2018; Zink et al., 2018). Those additional observations can be included in the optimization procedure by a multi-criteria objective function. This constrains the parameter optimization and can improve the representation of hydrologic states and fluxes in a model (Zink et al., 2018). In this publication we will investigate the usefulness of multi-criteria optimization for finding transfer functions.

This publication presents the Function Space Optimization (FSO) as a method for estimating parameter transfer functions for distributed hydrological models and applies the FSO method in a case study. The case study uses synthetic runoff data and consists of two tests. In the first test only runoff data is used for estimating transfer functions. In the second test we

investigate how the use of additional spatial-temporal information in a multi-objective optimization can improve the transfer function estimation.

2 Methods

2.2 The MPR method

Let the two spatial scales used in the MPR approach be denoted by $\mathcal{O}$ and $\mathcal{M}$, for the spatial scales of observations and the model, respectively. Note that $\mathcal{O} < \mathcal{M}$ is a necessary condition for MPR. Let $\theta^{\mathcal{O}} \in \mathbb{R}^n$ be the model parameters on the spatial scale of observation, defined by

$$\theta^{\mathcal{O}} = f_{tf}(\boldsymbol{X}^{\mathcal{O}}, \beta). \quad (1)$$

We call $f_{tf}: \mathbb{R}^{n \times sp} \rightarrow \mathbb{R}^n$ a *transfer function*. It uses a set of k numerical parameters $\beta \in \mathbb{R}^k$ to map the matrix $\boldsymbol{X}^{\mathcal{O}}$ to $\theta^{\mathcal{O}}$. Here, $n \in \mathbb{N}$ is the number of grid cells defining the catchment. The matrix $\boldsymbol{X}^{\mathcal{O}} \in \mathbb{R}^{n \times sp}$ contains the $sp \in \mathbb{N}$ physical properties of the catchment on the spatial scale of observations for each grid cell. We refer to those properties as *spatial predictors*. While $\theta^{\mathcal{O}}$ and $\boldsymbol{X}^{\mathcal{O}}$ are spatially distributed, β is a vector of global parameters.

Model parameters on the spatial scale of the model, $\theta^{\mathcal{M}} \in \mathbb{R}$, can thus be defined as

$$\theta^{\mathcal{M}} = f_a(\theta^{\mathcal{O}}). \quad (2)$$

Where f_a denotes an *aggregation* function which upscales the values of $\theta^{\mathcal{O}}$ to $\theta^{\mathcal{M}}$. Theoretically, any kind of aggregation function is possible. Samaniego et al. (2010) give the following examples for possible upscaling functions: Arithmetic mean, geometric mean, harmonic mean, maximum difference and the majority. There are no explicit averaging rules for various model parameters (Samaniego et al., 2010) and in the case of no existing theories, trying different basins and spatial scales might be the only procedure to identify them adequately (Samaniego et al., 2017). Another possible approach was described in a recent publication by Schweppe et al. (2019). They implemented MPR using the generalized mean with the form $M_p(x_1, \dots, x_n) = (\frac{1}{n}\sum_{i=1}^{n} x_i^p)^{\frac{1}{p}}$. The exponent $p \in \mathbb{R}$ can be interpreted as a weighting, which gives either more importance to large values ($p > 1$) or smaller values ($p < 1$). The special case of $p = 1$ is the arithmetic mean. This general form of averaging can be optimized and therefore included in any optimization routine.

The problem of inferring the transfer of spatial predictors to model parameters can roughly be divided into two parts: (a) finding the correct transfer function and global parameters and (b) finding the correct aggregation function. We here apply the arithmetic mean aggregation and focus mainly on the estimation procedure of transfer functions and global parameters. For (b), one can either define an aggregation function using knowledge from previous investigations about the parameter, trying different aggregation functions; or use the generalized mean as an additional parameter to optimize.

With this we can define the aim of any transfer function estimation procedure:

$$arg\,min_{f_{tf},\beta}\, \varepsilon = arg\,min_{f_{tf},\beta}\, f_{loss}(Q_{sim}, Q_{obs}) \quad (3)$$

Where ε is the model loss, defined by a loss function f_{loss}, which is dependent on the model simulated discharge Q_{sim} and the observed discharge Q_{obs}. A suitable loss function must

be chosen depending on the specific problem, e.g. Nash-Sutcliffe efficiency (Nash & Sutcliffe, 1970) for rainfall-runoff modelling.

Considering these definitions, assumptions and restrictions, we developed a method to infer parameter transfer functions and their global numerical parameters simultaneously from data.

2.2 Function Space Optimization (FSO)

The main difficulty of inferring f_{tf}, lies in the transfer of the task into an optimizable problem. In general, this means transferring it into a searchable numerical space. To make it searchable, close points in this space should also be close in their loss function, hence producing a smooth response surface. Since it is not possible to estimate the loss functions of all relevant transfer functions (a case where optimization would not be necessary), we have to find other properties which induce this closeness of loss function.

This leads to the main idea of FSO: define a numerical space which defines distance between functions by (1) semantic closeness and (2) closeness in the resulting parameter distributions. Property (1) specifically includes the interpretation of functions as text, in which function symbols (e.g. "+", "-", "elevation", …) are interpreted as words. Property (2) is necessary since physical catchment properties are often highly correlated. Hence, the functions "$sand \times 0.3 + 1.3$" and "$clay \times -0.38 + 1.6$" produce nearly the exact same parameters, even though their semantics are different.

Property (2) implies the a priori choice of global parameters β (the numerical values in the function) and results in distinguishing f_{tf} also by their specific β values. This allows for the simultaneous optimization of f_{tf} and β, since they are both represented in the numerical space. For brevity, we will use the term f_{tf} or transfer function, as a synonym for f_{tf} and β.

By transferring the problem in a numerical space with the above mentioned properties, any continuous global optimization method would be applicable. The steps for creating such a space and its use in the estimation of a transfer function will be described in the following sections.

2.2.1 Defining relevant transfer functions

In the first step of creating a search space, it is necessary to define a realm of possible transfer functions. A context-free grammar (CFG) (Knuth, 1965) is used for this purpose. In general, a CFG is a set of variables, operators and structural rules that can produce strings. It consists of nonterminal symbols and their corresponding mapping. A nonterminal symbol, in contrast to a terminal symbol, refers to a symbol that can still be further evaluated in the CFG (i.e. it has a mapping). An example of a simple CFG is given in Figure 1. A detailed and formal definition of CFGs can be found in Klotz et al. (2017).

In the simple example in Figure 1 only two options are available for all nonterminals. However, in any actual application this number will be larger. The nonterminals of a CFG can be interpreted as pathways in a decision tree and the corresponding pathway options can be used to represent a function as a discrete vector. The lower left part of Figure 1 shows how a function can be derived from the CFG by choosing a certain pathway in its decision tree representation. The lower right part shows how the same function can be represented as a discrete vector. The

entries of this vector correspond to the chosen option for each of the nonterminal symbols. The discrete vector has the same length for every function in a grammar. Therefore, if a nonterminal (e.g. the second **a** in the vector representation in Figure 1) is not used in a function pathway, its entry is not affecting the resulting function. For that reason, one of the entries of the discrete vector in Figure 1 can either be 1 or 2 and still produce the same function.

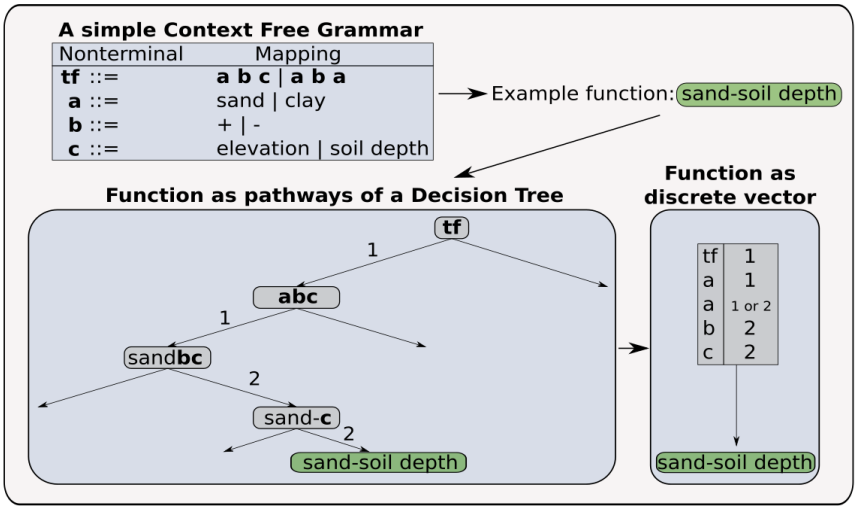


Figure 1. Example of a simple context free grammar (CFG) with an example function in its decision tree and vector representation.

Klotz et al. (2017) used the vector representation of transfer functions as search space for solving the problem of f_{tf} estimation. Such process converts the search for the optimal transfer function into a discrete optimization problem. Even though this is a straightforward approach, it results in an ill-defined optimization space and a bias towards very simple solutions. Both issues result from the properties of the vector representation of a CFG. For simplicity we will here refer to the CFG vector representation of a transfer function as V_{CFG}.

Looking at the characteristics of V_{CFG} regarding its ability to map functions to integers, two important properties can be noticed: (1) any distance metric for numerical vectors (e.g. Euclidean distance) does not reflect the closeness of the resulting functions in the objective function and (2) the representation of a function as V_{CFG} is not unique. Both properties result from the fact that we use a discrete vector to represent a directed graph.

These two properties influence the optimization of V_{CFG} significantly. Property (1) results in a very difficult and ill-posed optimization problem, considering that close points in the vector space most likely will not reflect similar results in terms of the objective function. Property (2) results in an optimization problem which is strongly biased towards simple functions. Simple functions can generally be represented with less dimensions than more complex ones. Since V_{CFG} has the same dimensionality for all functions, this results in a large part of the V_{CFG} that has no effect on the resulting functions. Hence, many different V_{CFG} will produce the same function. The resulting increased probability of finding simpler functions compared to more complex functions leads to a bias in the optimization.

These issues were one of the main motivations for developing FSO. The main advantage of using a CFG for FSO, is the possibility of sampling functions, while preserving the defined function properties. Thereby, we use it to create a (very large) realm of possible function for the transfer function search.

2.2.2 Variational Autoencoder

One type of generative models often used in Natural language processing (NLP) are autoencoders (Le Cun & Fogelman-Soulié, 1987). Autoencoders consist of two neural networks, an encoder network which maps the input to a continuous vector representation and a decoder network which reconstructs the encoded input from the continuous vector representation (see figure 2). A main advantage of using an autoencoder is the resulting low dimensional continuous vector representation of the inputs. This continuous vector representation is called the *latent representation* or *latent space* of the input information. After training an autoencoder to correctly encode and decode the information of a set of strings, the decoder can be used to generate strings from the latent space.

When left unconstrained, the latent space could potentially be sparse, meaning that large areas within the space would not produce any valid functions. Consequently, it is necessary to constrain it. To include a constraint on the latent space, we use a variational autoencoder (VAE) architecture (Kingma & Welling, 2013). VAEs add stochasticity to the latent space, which results in a latent representation that is more robust to small variations. Furthermore, it enforces a certain distributional behaviour (usually Gaussian) onto the space by adding a penalty term. A definition of the VAE architecture is given in the supporting information (Text S1). A detailed definition and derivation of VAEs and their properties can be found in Kingma & Welling (2013).

A simplified representation of the FSO VAE is shown in Figure 2. The encoder consists of a combination of word embeddings (Mikolov et al., 2013), convolutional layers (CNN) (LeCun et al., 1989) and feedforward neural network (FNN) layers (White & Rosenblatt, 1963) with selu (scaled exponential linear unit) activation functions (Klambauer et al., 2017). The decoder is a combination of FNN layers with selu activation functions and a long short-term memory (LSTM) network (Hochreiter & Schmidhuber, 1997). The chosen architecture for the encoding and decoding of function strings was inspired by an architecture developed by Gan et al. (2018). The additional encoding and decoding of the parameter distributions aims to further condition the latent space to also include the information about the resulting parameter distribution in the latent space. A detailed description of the FSO VAE is given in the supporting information (Text S2, Figure S1).

To incorporate both semantic and parameter distribution information in the autoencoder, two kinds of inputs/outputs are used for training: the transfer function strings and the parameter distribution resulting from that transfer function. The transfer function is given as a vector of symbols, e.g. “sand”, “+”, “slope”. The dimension of this vector equals the maximum length of a transfer function created by the CFG. The parameter distribution is given as a numeric vector containing the 0.1 to 0.9 quantiles in 0.1 steps and is estimated from the spatial predictors of the catchment. The FSO VAE encodes this information into a 6-dimensional numerical space that we call *Function Space*.

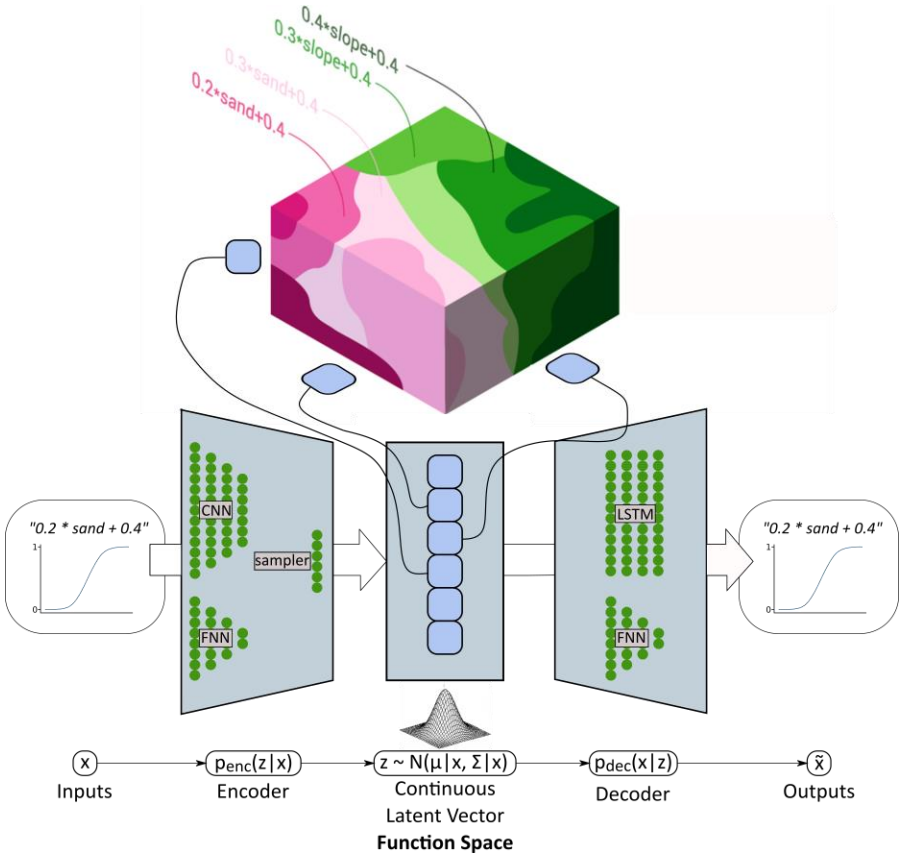


Figure 2. A simplified depiction of the FSO Variational Autoencoder with an example function that gets transferred to the Function Space and reconstructed to its original form. The inputs are transferred to the Function Space using the encoder network. The Function Space representation is then passed through the decoder network to reconstruct the inputs. The Function Space is a 6-dimensional continuous vector space with a Gaussian distribution.

The FSO VAE is trained to minimize three type of losses: (1) The cross-entropy loss resulting from the reconstruction of functions strings, (2) the mean squared error of the parameter distribution reconstruction and (3) the Kullback-Leibler divergence (Kullback & Leibler, 1951) between the Function Space and a Multivariate normal distribution. They are weighted with factors to balance their importance during training. A detailed description of the loss function is given in the supporting information (Text S2).

After training, the VAE is able to reconstruct the input data, and the decoder can generate a function from every point in Function Space, which has the same properties as defined by the CFG.

2.2.3 Normalization

To enable the unbiased estimation of universally applicable transfer functions, a cascade of scaling is necessary to:

- make the spatial predictors of the catchment comparable,

- make the transfer functions usable in areas where the range of spatial predictors is outside of the observed range of the catchment used for transfer function estimation,
- be able to predict a certain model parameter in the correct range of its feasible values.

Scaling for any values x to an arbitrary interval $[a, b]$ is done in FSO by applying the min-max scaling function: $x_{[a,b]} = a + \frac{(x-\min(x))(b-a)}{\max(x)-\min(x)}$.

To be able to compare the information from multiple spatial predictors of the catchment, they are scaled to the interval $[0,1]$ (i.e. data normalization). An issue of trying to find parameter transfer functions that are universally applicable is the dependency on the scale of the catchment that was used to derive it. To avoid this restriction, the scaling to the interval $[0,1]$ is done by using their physically possible or reasonable minimum and maximum values. E.g. all slope values are scaled from the interval $[0,90]$ to $[0,1]$. Thus, catchment characteristics outside the observation range of the current data set will still be in the range $[0,1]$.

To be able to estimate the parameters in their corresponding scale, the values resulting from applying the transfer functions are rescaled to the parameter bounds. The parameter bounds need to be chosen for each parameter and should reflect the values in which the parameters have any (physical) meaning. This allows for global scale application.

2.2.4 Optimization in Function Space (FSO)

The full workflow of FSO is shown in Figure 3. It consists of two main parts: the assembly phase and the optimization phase. The steps of the assembly phase were already described in sections 2.1.1-2.1.3. It includes the selection of parameters, the selection of spatial predictors, the CFG definition and the training of the VAE.

The optimization phase of the FSO is a fully automatic procedure that uses the text generating VAE which was trained in the assembly phase. It searches for the optimal point in the Function Space (i.e. transfer function(s)) to minimize the loss function. In each iteration a new function is generated from the Function Space, which is used to produce a parameter field. This new parameter field is used in the hydrological model and results in a loss function output. After a previously defined number of iterations, the function with minimum loss is chosen as the estimation for the parameter transfer function.

In general, any continuous optimization algorithm can be applied in the optimization phase. We experimented with three commonly used algorithms: Genetic Algorithm (Holland, 1975), Dynamically Dimensioned Search (Tolson & Shoemaker, 2007) and the Particle Swarm Optimization (Kennedy & Eberhart, n.d.). All of them were able to solve the given optimization problem equally well. Our tests showed that the Dynamically Dimensioned Search (DDS) performed slightly more consistently than the other two. Consequently, we decided on using the DDS for the optimization in function space.

FSO can optimize multiple parameters at the same time. This can be done by optimizing multiple Function Spaces. Since each function is represented as a 6-dimensional continuous vector in function space, optimizing two transfer functions results in a 12-dimensional continuous optimization problem.

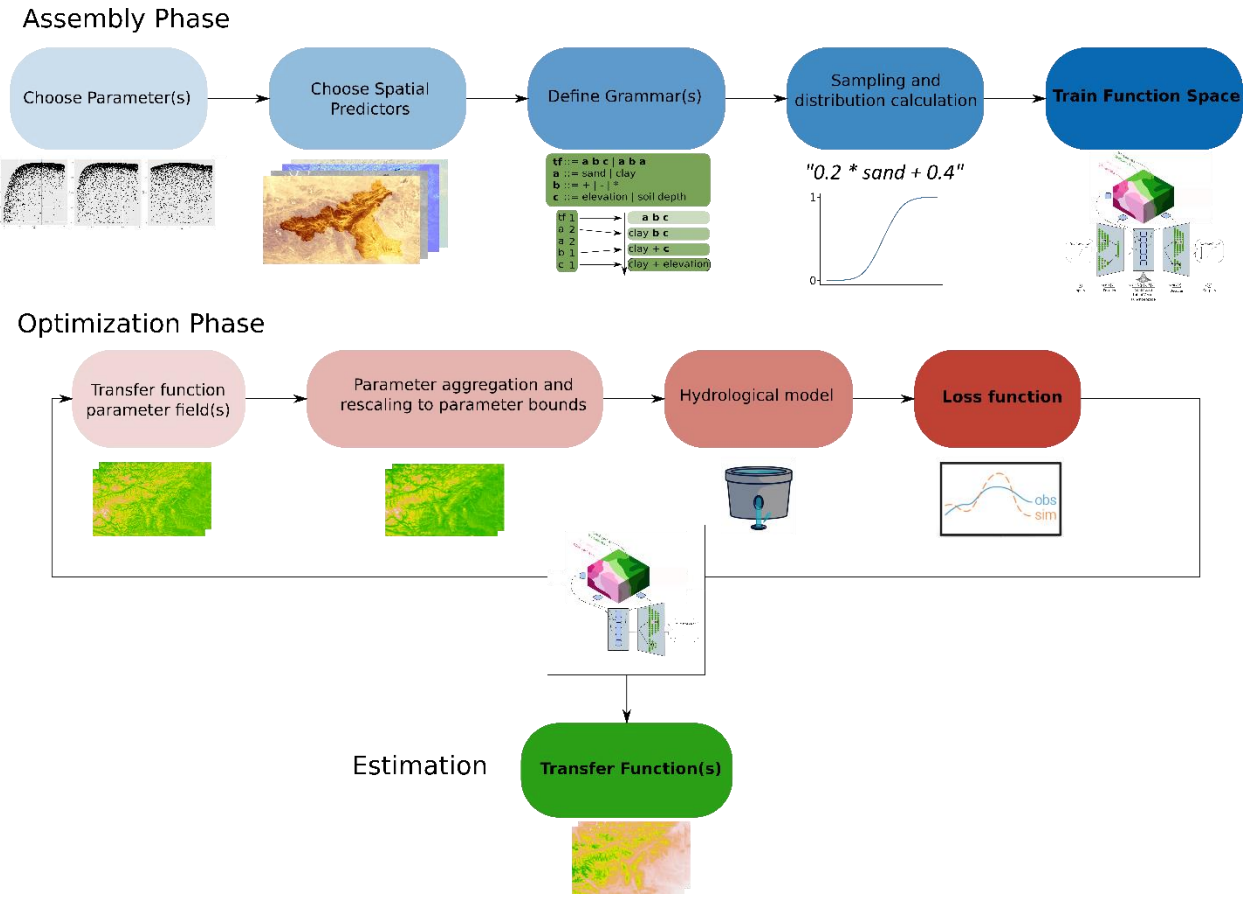


Figure 3. The Function Space Optimization workflow. It consists of two phases, the assembly phase with all the necessary steps to train the FSO VAE and the optimization phase in which a continuous optimization algorithm is used to optimize in Function Space. The decoder part of the VAE is used in the optimization loop to generate new functions from Function Space.

3 Case Study

To test whether FSO is able to sufficiently approximate transfer functions, we conducted a test in a virtual reality setting. This case study applies FSO on a parsimonious distributed model using synthetic runoff data.

3.1 Mur catchment

The case study was performed using hydrological, meteorological, climate and geographic data from the Mur catchment (see Figure 4), which is located in the south-eastern part of Austria and has an area of 10,420 km^2. For testing FSO, we intended to use data from a catchment with a wide range in physical properties to explore its applicability on a large scale. As the Mur consists of high alpine areas in the north and significantly lower areas in the south and very diverse geology, it fulfills this condition. Another reason for selecting this catchment was to compare the further development in the regionalization method with the work by Klotz et al. (2017)

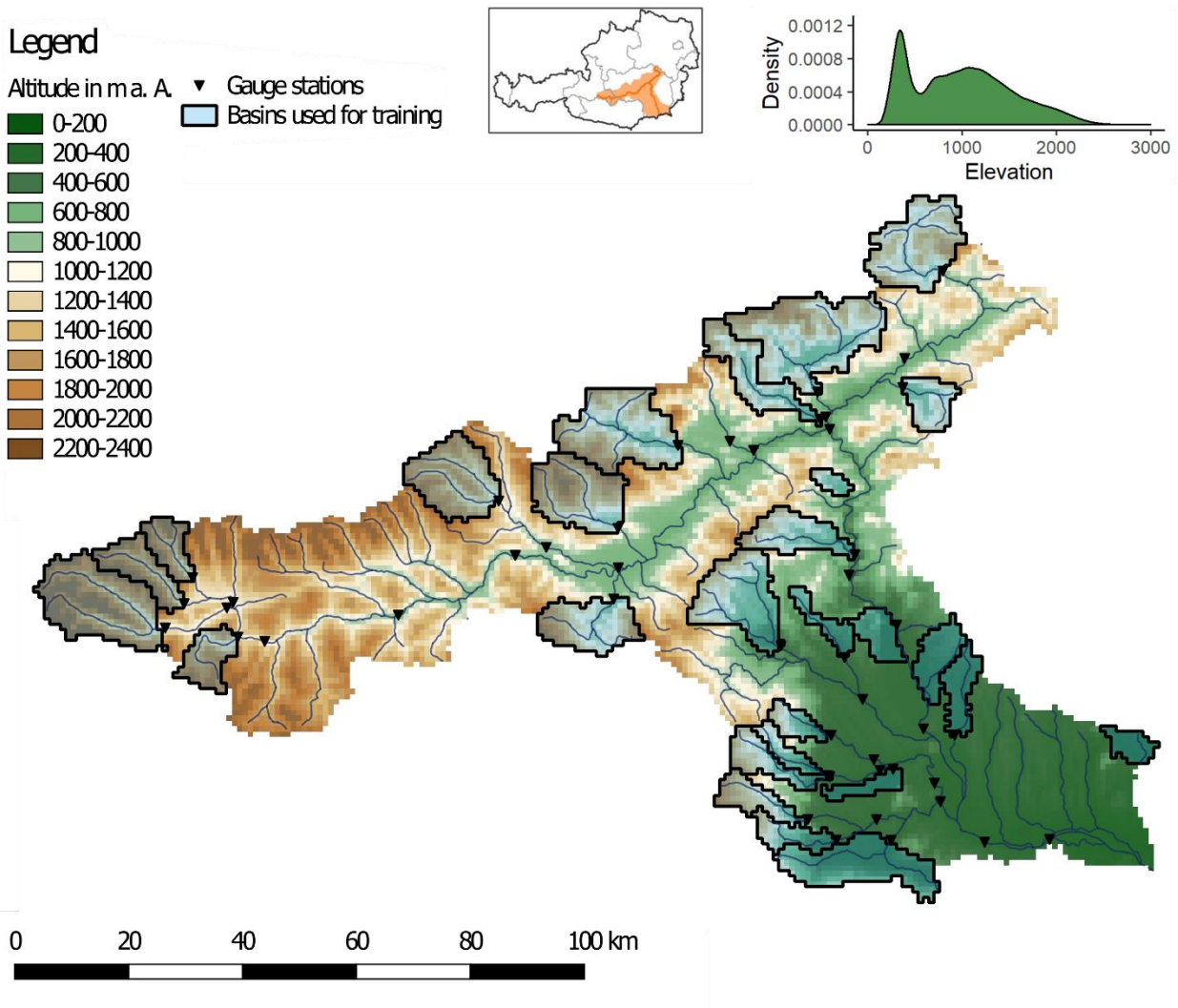


Figure 4. The digital elevation model of the Mur catchment on a 2 km grid. The 27 headwater basins used for training FSO are shown in blue. The top right part of the figure depicts the catchments locations in Austria and its elevation distribution.

The 250 m gridded geo-physical properties used in this case study are: height above sea level (*elevation*), slope (*slope*), height above nearest drainage (*hand*), percentage of clay (*clay*), percentage of sand (*sand*), soil depth (*bdim*), the enhanced vegetation index (*evi*) and a noise layer (*noise*). Topographic properties (*elevation, slope, hand*) were calculated from a digital elevation model obtained from Rechenraum e.U (2012) and soil information (*clay, sand, bdim*) was obtained from SoilGrids (Hengl et al., 2017). SoilGrids is a system for digital soil mapping using state-of-the-art machine learning. The *evi* layer was derived by averaging an *evi* time series for the years 2000-2017 from Didan (2015). For further information about the enhanced vegetation index refer to Huete et al. (2002). In addition to the observation data, we generated a noise layer *(noise)* that consists of values sampled from a uniform distribution over the interval [0,1]. The *noise* layer was created for further testing FSO by providing irrelevant information as a possible predictor.

All geo-physical properties are strongly correlated with an overall mean absolute correlation coefficient of 0.55 (these values do not include *noise*). Very high correlation coefficients could be observed for clay/sand (-0.95), clay/elevation (-0.83) and elevation/bdim (-0.82) and the lowest was observed for evi/hand (-0.35).

The meteorological data used in this case study are air temperature, and precipitation from the INCA analysis (Haiden et al., 2011, 2014). The potential evapotranspiration was computed using the Thornthwaite equation (Thornthwaite & Mather, 1957).

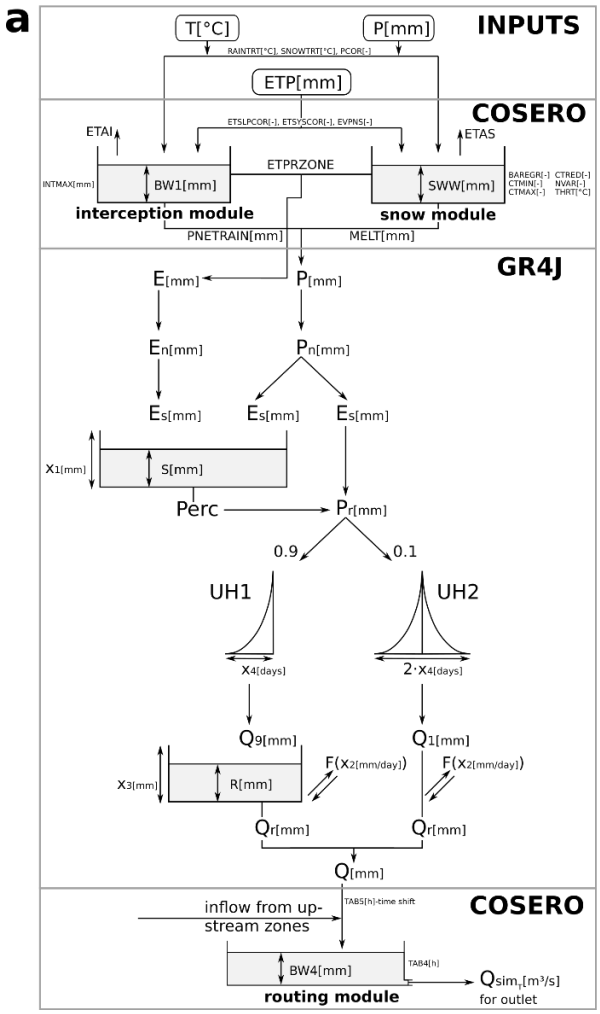
To evaluate the predictive capability of the algorithm, 2 data splits were applied. First, we split the time series and used the period 01.2003- 08.2009 for training and the years 09.2009 – 12.2012 for testing. Secondly, we split the basins and used 27 headwater basins (marked as blue in Figure 4) for training and 95 basins for testing. Thereby, we not only estimate the ability to predict an independent time period, but also the ability to predict runoff in ungauged basins.

3.2 Distributed GR4J

For testing purposes, the parsimonious hydrological model GR4J (Perrin et al., 2003) was chosen. GR4J is a lumped hydrological model for predicting daily mean runoff. It is a simple 4 parameter model, consisting of two storages and two unit hydrographs (Sherman, 1932). The GR4J model structure can be seen in the middle part of Figure 5a. To implement GR4J as a distributed model and to include snow and interception processes, we combined it with the routing, interception and snow module from the COSERO (Continuous SEmidistributed RunOff model) model. It is a HBV-type model which was developed by Nachtnebel et al. (1993) and was applied in lumped and semi-distributed (Kling et al., 2015; Stanzel et al., 2008) and in distributed settings (Frey & Holzmann, 2015; Herrnegger et al., 2012, 2018; Kling et al., 2006; Kling & Nachtnebel, 2009; Wesemann et al., 2018). We will refer to this extended version of GR4J as d-GR4J. The complete d-GR4J model structure can be seen in Figure 5a.

GR4J consists of 4 parameters: X_1 production store maximal capacity (mm), X_2 catchment water exchange coefficient (mm/day), X_3 one-day maximal capacity of the routing reservoir (mm) and X_4 unit hydrograph time base (days). A more detailed description of the GR4J model is given by Perrin et al. (2003). All parameters from the COSERO part of d-GR4J were taken from previous calibration of COSERO for the Mur catchment, leaving only the 4 GR4J parameters to be optimized.

In order to demonstrate that d-GR4J is generally able to adequately describe catchment hydrological processes, we performed an initial conventional parameter optimization against observed discharge data, using the DDS algorithm. To further investigate whether the GR4J parameters are a reasonable choice for the parameter transfer function estimation, we examined their sensitivity with a Monte Carlo parameter simulation and a global parameter sensitivity estimation using the Fourier amplitude sensitivity test (FAST) (Cukier et al., 1978) which was already applied on multiple hydrological models (e.g. Francos et al., 2003; Y. Gan et al., 2014; Ratto et al., 2001; Reusser et al., 2011). While a Monte Carlo simulation provides useful insight it does not provide a quantitative sensitivity estimation (Wang, 2012). FAST estimates the fractional contribution of individual parameters to the variance of the output and therefore quantifies the sensitivity of individual parameters.



b

Case study model setup

Spatial scales
Observation scale $\mathcal{O}$ = 250 x 250 m
Model scale $\mathcal{M}$ = 2 x 2 km

Data splits
Training: 01.2003 - 08.2009, 27 basins
Testing: 09.2010 - 12.2012, 95 basins

Spatial predictors

elevation [m]	sand [%]
slope [°]	bdim [m]
hand [m]	EVI [-]
clay [%]	noise [-]

Parameters to estimate
x_1 [mm] production store maximal capacity
x_3 [mm] maximal capacity of the routing reservoir
x_4 [days] unit hydrograph time base (days)

Workflow
1. Choose true parameter fields
2. Compute synthetic Q_{obs}
3. FSO parameter estimation

Figure 5. **a** d-GR4J model structure, divided into parts from COSERO and GR4J. **b** Overview of the case study model setup.

3.3 Design of a "virtual hydrological reality"

Prior to real world applications, it is necessary to test the principle functionality of FSO under more controlled conditions to avoid possible sources of errors such as measurement errors, wrong assumptions for model parameters or missing spatial predictors. We used real observation data for the catchment properties, but generated the discharge values synthetically, using a priori defined "true" transfer functions. These true transfer functions are used to generate d-GR4J parameter fields and, with observed climate data as input, time series of discharge data for each sub-basins. In the following, these discharge data were treated as observations from which the true transfer functions were re-estimated.

Our objective is to re-estimate transfer functions for three GR4J parameters: X_1, X_3 and X_4. The parameter X2 allows for water inflow or outflow of the basin. Because X2 is not reflected in precipitation or discharge, was set to 0. Hence, we assumed that there is no unobserved water in- or outflow in the catchment.

We chose the true underlying transfer functions that use one or two different spatial predictors. The transfer functions for the 3 parameters are chosen to reflect the possible physical interpretation of the parameters, e.g. large values of *evi* and soil depth (*bdim*) result in a large production store. Their resulting parameter fields are shown in Figure 6a-c. The chosen true transfer functions are:

$$X_1 = 0.5 + evi \cdot 1.5 + exp(bdim) \cdot 0.9 \quad (4)$$

$$X_3 = -1.3 - log(slope) \quad (5)$$

$$X_4 = -1.5 - log(hand) \cdot 0.2 - slope \cdot 1.5 \quad (6)$$

The CFG for the case study is chosen to be complex, to create a large search space. It included multiple recursive nonterminals, exp/log functions, power functions, reciprocal functions and linear combinations. Additionally, to include the search for the global parameters β, the numerical values -1.5 to 1.5 in steps of 0.1 are added as terminal symbols. Since all spatial predictors are in the interval [0,1], the range [-1.5, 1.5] for numerical values should include enough complexity to simulate a real task. Nevertheless, in a real world application it might be helpful to increase that range and reduce the step size, which would only increase computation time in the assembly phase of FSO and not for the optimization. The complete CFG used in the case study is shown in the supporting information (Figure S2). The three true transfer functions can be generated from the CFG and are therefore included in our possible realm of transfer functions.

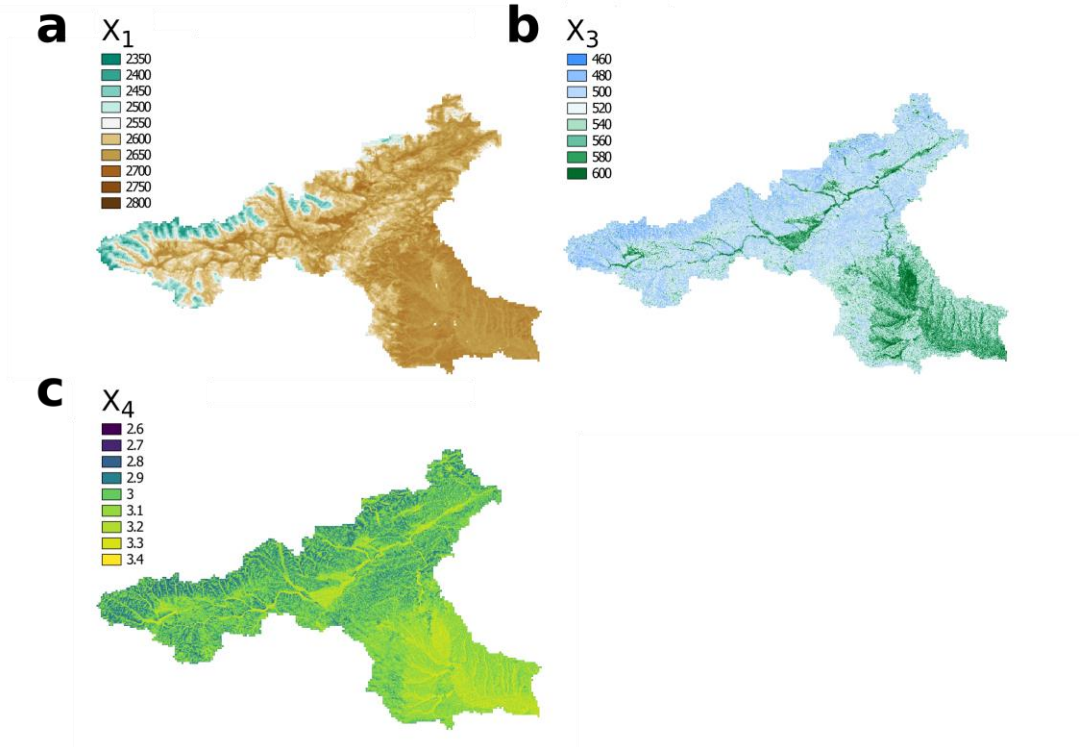


Figure 6. True parameter field for the 3 d-GR4J parameters **a** X1, **b** X3 and **c** X4.

From this CFG we sampled 5 million unique functions. We used 80% as the training set and 20% as the validation set for the VAE. The minimum validation loss was reached after 132 epochs of training with a batch size of 1000.

After selecting the true transfer functions, the resulting parameter fields are used to produce a synthetic discharge time series using d-GR4J on a 2 km grid. Also, the internal states of the storage S and R for each grid and each time step are calculated. An overview of the case study model setup is shown in Figure 5b.

3.4 Applying FSO

The performance of FSO is strongly dependent on the choice of the loss function f_{loss}. Therefore, two different tests were conducted using:

1. a single-objective criterion, with loss only dependent on discharge, and
2. a multi-objective criterion, with loss dependent on discharge and state "observations".

For both tests, the optimization procedure was repeated five times to evaluate the variations in the retrieval process, while a maximum number of 3000 iterations for the DDS global optimization algorithm was chosen.

3.4.1 Test 1: single-objective criteria

Test 1 focuses on estimating transfer functions by considering a loss that is only dependent on the predicted and observed discharge. The loss function is formulated similar to the Nash-Sutcliffe efficiency $NSE = \frac{\sum_{it=1}^{T}(Q_m[t] - Q_o[t])^2}{\sum_{it=1}^{T}(Q_m[t] - \overline{Q_o})^2}$ (Nash & Sutcliffe, 1970), using a weighted mean NSE value of the form

$$NSE_{wm} = \frac{\sum_{i=1}^{m} w_i NSE(Q_{s,i}, Q_{p,i})}{\sum_{i=1}^{m} w_i}, \quad (7)$$

with the weights $w_i = 1 - NSE(Q_{s,i}, Q_{p,i})$ for all $i \in \{1, \ldots, m\}$, which is a weighted arithmetic mean over m basins. $Q_{s,i}$ and $Q_{p,i}$ are the synthetic and predicted time series of discharge for basin i, respectively. By using this form of averaging, the basins with the lowest NSE values get the highest weight, while basins with NSE close to 1 become unimportant. This forces the optimization procedure to estimate transfer functions that operate equally well in all catchments.

Finally, an additional penalty term is added to the loss function to reduce the possibility of overfitting. This term penalizes for the transfer function length, i.e. the number of symbols used in a function, which can be interpreted as the function complexity. We defined it as $loss_{size} = \text{transfer function length} \cdot 0.001$ resulting in

$$f_{loss} = -NSE_{wm} + loss_{size}. \quad (8)$$

3.4.2 Test 2: multi-objective criteria

To implement a multi-criteria optimization in FSO, we adapt f_{loss} to include the loss from additional sources. In Test 2, we assume the existence of an additional observation of time series of our GR4J system states S (production store) and R (routing store). and define a multi-criteria weighted mean NSE as

$$NSE_{wm} = \frac{\sum_{i=1}^{n} w_i NSE_{multi,i}}{\sum_{i=1}^{n} w_i}, \quad (9)$$

with $NSE_{multi,i} = 0.5 \cdot (NSE(\mathrm{Q_{s,i}}, \mathrm{Q_{p,i}}) + \mathrm{NSE}(\mathrm{State_{s,i}}, \mathrm{State_{p,i}}))$ and the weights $w_i = 1 - NSE_{multi,i}$ $for\ all\ i \in \{1, \ldots, n\}$. $\mathrm{State_{s,i}}$ and $\mathrm{State_{p,i}}$ are the mean synthetic and mean predicted time series of model states for basin i, respectively. $NSE_{multi,i}$ is the arithmetic mean of the NSE values used in the multi-criteria objective of basin i. Hence, when optimizing discharge and a model state, it results in using the mean of two NSE values. Using equation 8 again, we can thus define our multi-criteria loss function.

We applied the multi-criteria FSO using S, R and both simultaneously. For brevity, we here present only the optimization using the time series of state S. The results for the other two can be found in the supporting information (Figures S5 and S6).

4 Results

4.1 Function Space

Firstly, we will illustrate the properties of the FSO function space by analyzing the generated functions and resulting parameter distributions when moving on a straight line between two points in the space. As distance in function space should not only reflect semantic closeness, but also closeness in their resulting parameter distributions, we expect to see a gradual change in both properties in this linear interpolation, thus indicating an appropriate searchability.

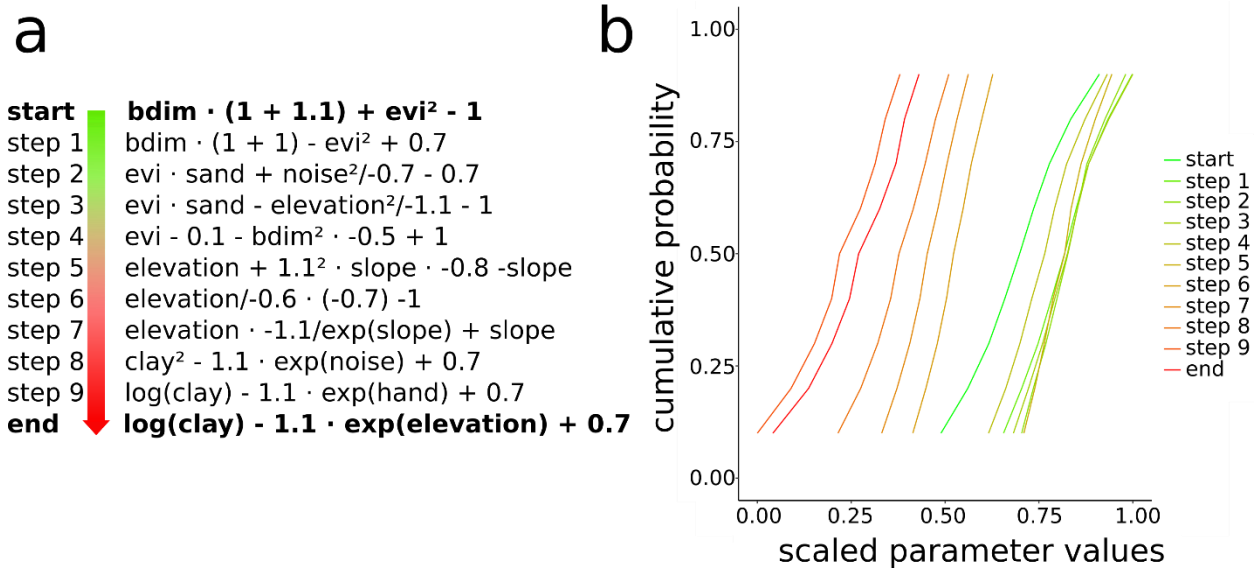


Figure 7. A linear interpolation in Function Space. **a** The functions generated along a straight line between start and end points in function space. **b** The corresponding quantiles of scaled parameter values.

We chose two random points in Function Space, F_1 and F_2, and linearly interpolate between them $F(w) = w \cdot F_1 + (1 - w)\, F_2$, with weights w ranging from 0.1 to 0.9 with 0.1 steps between them. Hence, we produce new functions from FS while moving on a line between F_1 and F_2 on every tenth of the way. The corresponding functions are shown in Figure 7. Figure 7a shows the generated function strings. The further away we move from the starting point, the

more function strings resemble the end point function. Figure 7b shows the corresponding parameter distributions resulting for all functions in this linear interpolation. Here we observe that functions closer to each other in FS are also closer in terms of produced parameter distribution.

4.2 Global Sensitivity and Model Performance

The conventional optimization on real observation data showed that d-GR4J is able to map observed runoff dynamics resulting in a mean basin NSE of 0.78 with a minimum NSE of 0.57 and a maximum NSE of 0.91. Therefore, we can assume a general ability of d-GR4J to model rainfall-runoff processes in catchments. A relevant property of d-GR4J regarding the estimation of transfer functions is the parameter sensitivity. Figure 8a-c shows the parameter response surfaces resulting from the Monte-Carlo simulations. We can see peaks for all 3 parameters and a clearly defined response surface. Figure 8d shows the results from the parameter sensitivity analysis using FAST. X1 has the highest sensitivity with ~40% contribution to the output variance, while X3 and X4 are less sensitive with ~20% and ~5%, respectively.

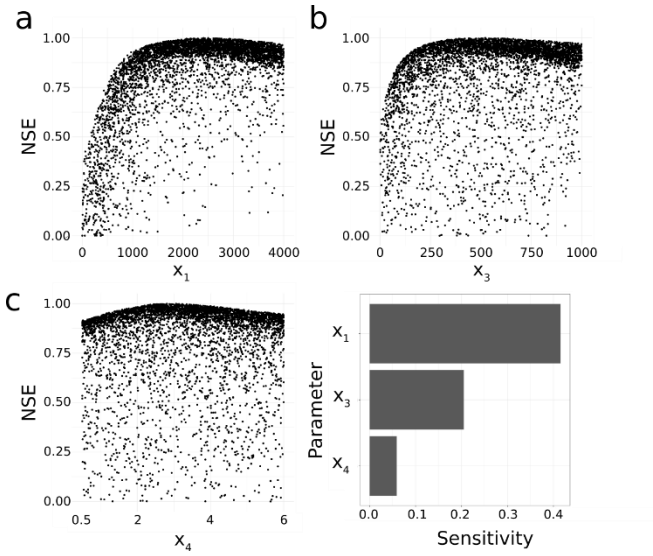
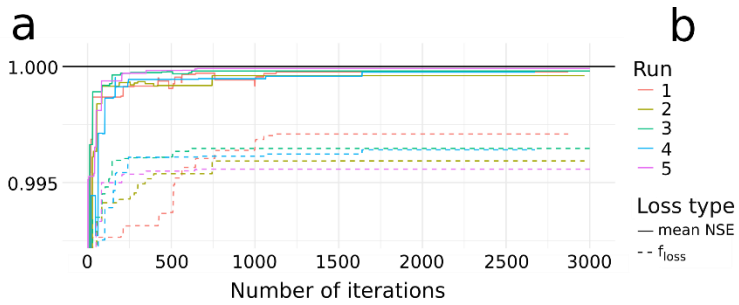


Figure 8. Parameter sensitivity of one sub-basin of the Mur catchment. **a-c** Parameter response surface for all three d-GR4J parameters with NSE values cut off below 0. **d** Results from the FAST sensitivity analysis showing the percentage contribution of individual parameters to the variance of the output.

4.3 Case study: single-criteria FSO

Figure 9a shows the training performance of all 5 single-criterium FSO optimization runs (see 3.4), i.e. results of the training basins for the training time period. The different runs are distinguished by their color. The two line-types show the mean NSE (solid) and f_{loss} (dashed) as defined in equation 5. It is clearly visible that the performance of FSO is stable, with a spread of mean basin NSE of less than 0.001. Furthermore, all runs arrive at a solution with NSE > 0.995 in less than 250 iterations.



Testing time period results

Training basins		Testing basins	
NSE_{mean}	> 0.999	NSE_{mean}	> 0.999
NSE_{wm}	> 0.999	NSE_{wm}	> 0.999
f_{loss}	= 0.995	f_{loss}	= 0.995

Figure 9. **a** Single-criteria FSO training results for all 5 runs. **b** Summary of performance during testing time period for training and testing basins.

For brevity, we will show the detailed results of one run only, which describes the general behaviour of all runs. Figure 9b shows the model performance in the testing time period. It is notable that we can observe the same quality of results for training ("gauged") and test ("ungauged") basins and both are close to the possible maximum, with an NSE of > 0.999. Naturally, such high NSE values are only possible in a synthetic setting in which we use the correct model and error free observation data.

The comparison between true f_{tf} and single-criteria FSO estimated f_{tf} can be seen in Table 1. FSO was able to predict the correct spatial predictor for X3 and one of the two spatial predictors of X1. Examining the true f_{tf} for X1 we can note that due to the exponent, *bdim* has a larger influence of the parameter values and is therefore much easier to find than *evi*.

Parameter	True f_{tf}	FSO estimated f_{tf}
X1	$0.5 + evi \cdot 1.5 + \exp(\mathbf{bdim}) \cdot 0.9$	$1.1 + hand \cdot 1.3 - \text{elevation} + \exp(\mathbf{bdim})$
X3	$-1.3 - log(\boldsymbol{slope})$	$1.5 - \boldsymbol{slope}/0.2$
X4	$-1.5 - log(hand) \cdot 0.2 - \text{slope} \cdot 1.5$	$0.7 + log(clay)$

Table 1. Comparison of the true f_{tf} and the single-criteria FSO estimated f_{tf}.

Figure 10 shows the estimated and true parameter distributions and scatterplots for all 3 optimized parameters. The means of the predicted parameter values are nearly the same for all three parameters, with X1 having the largest difference (10.88 or 3.67% of the total parameter value range). We can see that for X1 and X3, the parameter distributions of true and predicted f_{tf} are nearly identical for smaller values and more diverging for larger values. Figure 10b shows the linear relationship corresponding to a correlation of 0.98 for the predicted and true values of X1. Here, the predicted values of X3 seem to have a non-linear relationship but nevertheless a correlation of 0.96. For X4 we see a larger difference in the parameter density (correlation of 0.71), but it is notable that the mean value is nearly identical.

A comparison of the estimated parameter fields and the true parameter fields are shown in the supporting information (Figure S3).

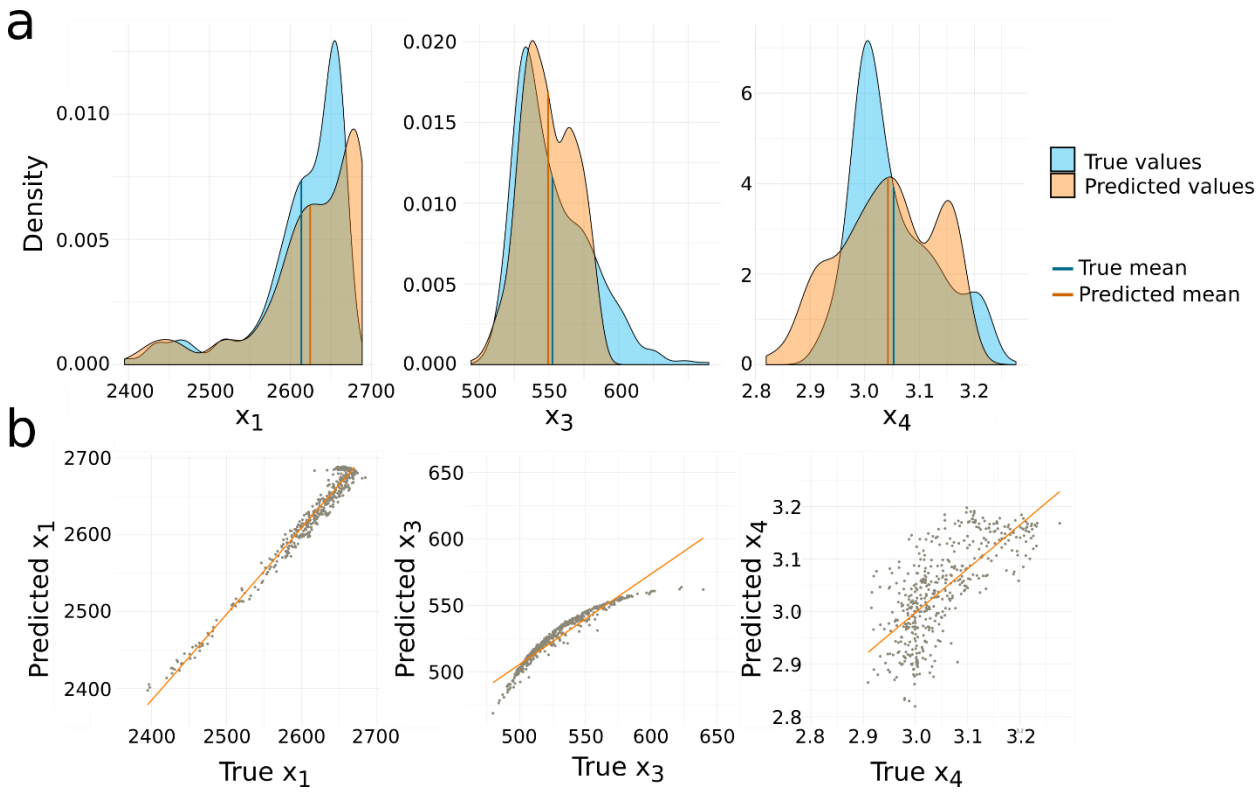


Figure 10. Single-criteria FSO results for all 3 optimized d-GR4J parameters on the 2 km model scale. **a** Estimated and true parameter densities with their mean values. **b** Scatterplots of true vs. estimated parameters and fitted linear model.

4.4 Case study: multi-criteria FSO

Figure 11a shows the training performance of all 5 multi-criteria FSO optimization runs using the d-GR4J state S as additional optimization criteria. The state S is only controlled by the parameter X1, hence we expect an improvement in estimating f_{tf} for X1. Compared to the single-criteria FSO, multi-criteria FSO has a slightly increased variance in f_{loss}. It is still very stable in regards to the mean NSE. Only run 3 is somewhat different with a training mean NSE of 0.988.

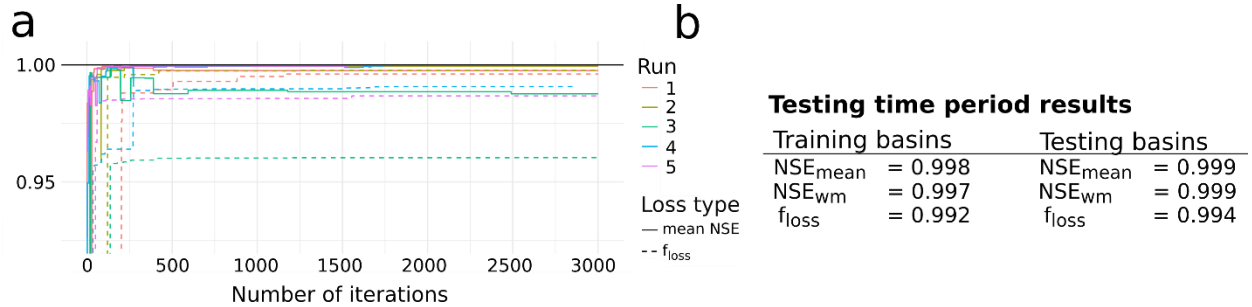


Figure 11. **a** multi-criteria FSO trainings results for all 5 runs. **b** Summary of performance during testing time period for training and testing basins.

For brevity, we will show the detailed results of one run only, which describes the general behaviour of all runs. Figure 11b shows the model performance in the testing time period. It is notable that we can observe the same quality of results for gauged and ungauged basins with both having an NSE ≥ 0.998.

The comparison between true f_{tf} and multi-criteria FSO estimated f_{tf} can be seen in Table 2. FSO was able to predict both spatial predictors of X1 correctly. The slope was included in the estimated f_{tf} of X3 and X4 correctly, otherwise the structure of the functions is different.

Parameter	True f_{tf}	FSO estimated f_{tf}
X1	$0.5 + \mathbf{evi} \cdot 1.5 + \exp(\mathbf{bdim}) \cdot 0.9$	$0.8 + \exp(\mathbf{evi}) + \mathbf{bdim}^2/0.8$
X3	$-1.3 - \log(\mathbf{slope})$	$\log(\text{bdim}) + \mathbf{slope}$
X4	$-1.5 - \log(\text{hand}) \cdot 0.2 - \mathbf{slope} \cdot 1.5$	$\log(\text{clay}) - (\text{bdim} \cdot \log(\mathbf{slope}) + \mathbf{slope})$

Table 2. Comparison of the true f_{tf} and the multi-criteria FSO estimated f_{tf}.

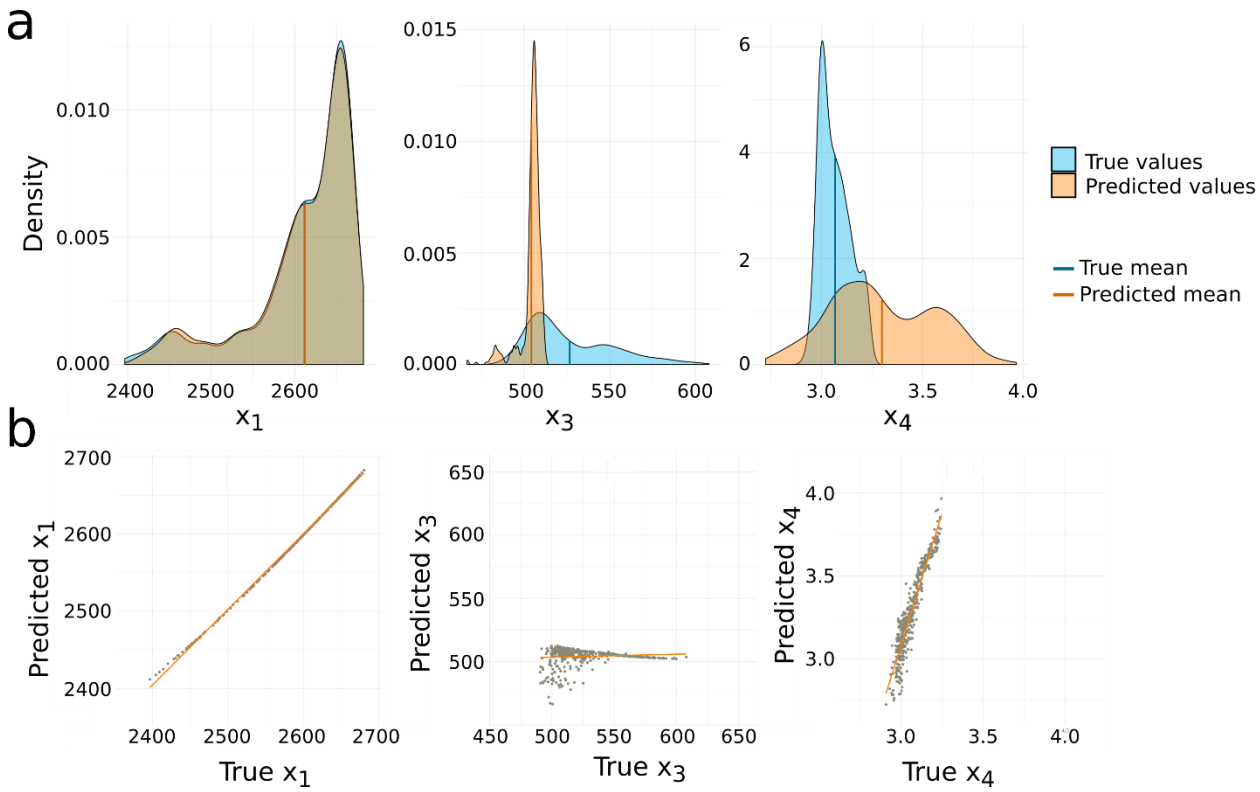


Figure 12. Results from multi-criteria FSO, using the information from the S of the d-GR4J model, which is controlled by the parameter X1. All 3 optimized d-GR4J parameters are compared to the true parameters on the 2 km model scale. Estimated and true parameter densities are shown in **a**. Scatterplots of true vs. estimated parameters and fitted linear model are shown in **b**.

Figure 12 shows the estimated and true parameter distributions and scatterplots for all 3 multi-criteria optimized parameters. The distribution of X1 is perfectly matched and the

predicted values have a nearly perfect linear relationship with a correlation coefficient of 1. However, X3 and X4 are less well matched compared to the single-criteria FSO results with most X3 values being underestimated and most X4 values being overestimated. Due to the small, but existing variance in the estimation procedure, some runs performed better than the one shown here, but most results were similar to the one shown in Figure 12. We chose this run as it shows the general trend of estimation results.

A comparison of maps of the estimated parameter fields and the true parameter fields are shown in the supporting information (Figure S4).

5 Discussion and outlook

In this study, we present a method to automatically estimate parameter transfer functions for distributed hydrological models. Defining parameters as functions of the geo-physical properties of a basin results in an increased physical interpretability of the model parameters, seamless parameter fields and the possibility of prediction in ungauged basins. Our approach is based on the compression of functions from a context free grammar into a searchable continuous space (Function Space), which subsequently can be used for continuous optimization.

To demonstrate the predictive ability of FSO, we conducted a case study using synthetic data to avoid any influence of potential sources of errors, such as: measurements errors and model assumptions, on the estimation procedure. The underlying true transfer functions were defined a priori and used for generating synthetic parameter fields that, in combination with the rainfall-runoff model d-GR4J, result in synthetic runoff and storage data.

We demonstrated that the developed Function Space has the desired properties of being "searchable" and that our chosen model parameters are sensitive. FSO is then tested in a case study using synthetic parameter fields and corresponding synthetic runoff and/or storage data.

The case study consists of two tests. Firstly, we apply a single-criteria calibration, optimizing the transfer functions only on runoff data in the calibration procedure. Secondly, this is then extended to additionally include spatially distributed time series of storage data in a multi-criteria optimization. For both tests we could find transfer functions that produce a nearly perfect discharge prediction with an NSE of 0.999 in "ungauged" basins.

The results of the single-criteria optimization show that FSO can find transfer functions that result in a perfect match for runoff and that FSO results are stable and multiple runs vary only insignificantly in their resulting mean basin NSE.

The results of the multi-criteria optimization showed an increasing performance when estimating the parameter (X3) that is associated with the storage observations that are added to the loss function. Looking at the results of the different optimization runs, it is noteworthy that having the additional term in the loss function increases the difficulty of the optimization problem (see Figure 11a), leading to one optimization run with a slightly lower mean basin NSE in the training period.

Both single- and multi-criteria FSO did not show a decrease in performance in the testing time period for the test basins. This shows that there is no overfitting on the training data and that prediction in ungauged basins is possible and performing as well as in a gauged basin. This is most likely due to the chosen penalty for complex functions in the loss function and the use of a weighted mean basin NSE value. Without the weighted NSE, single basins might have a bad fit

while the overall training NSE is close to 1. Without the penalty for the length of the transfer functions, we could potentially find very complex functions that can approximate every other function without having any association to the process. The results let us conclude that, if FSO has a reasonable set of representative training basins, prediction in ungauged basins is possible.

Comparing the estimated transfer functions with the underlying true functions for the 3 chosen model parameters, it is notable that the most sensitive parameter X1 is usually estimated with the smallest deviation from the true parameter values.

From the results of the case study we can define the phenomena of transfer function equifinality, i.e. non-unique best fitting transfer functions. We identified three main reasons for the occurrence of this form of equifinality. Reason one - Parameter sensitivity: A low parameter sensitivity results in a reduced ability to identify the true transfer function, since small variations in parameter values are irrelevant for the resulting model loss; Reason two - Information loss due to aggregation to the spatial scale of the model: Since aggregation generally results in loss of information, depending on the scale difference of observation and model scale, multiple transfer functions can potentially produce the same aggregated parameter field; Reason three - Correlation of geo-physical properties: Due to high correlation of geo-physical properties (see 3.1), different transfer functions using different spatial predictors can potentially produce similar parameter fields, e.g. functions using sand instead of clay. Due to these three reasons, FSO estimated transfer functions might differ in structure from the true underlying transfer functions, but still perform well. Nevertheless, compared to the classical parameter equifinality, transfer function equifinality does not remove the physical interpretability of the estimated functions.

Assuming that a certain parameter can be described in mathematical form by some geo-physical properties of a catchment, two important requirements are necessary for finding its true underlying transfer function. The CFG must be able to generate the function and the correct geo-physical parameters must be included in the CFG. In cases where these assumptions are violated, it can still be expected that FSO will find transfer functions which are associated with the physical processes described by the parameters. This is the case, because of the correlation of geo-physical properties. Meaning, that even if we have not included the “correct” geo-physical properties, we might still produce the correct parameter fields. Hence, we can assume association between model parameters and geo-physical properties, even if there is not causality. This certainly increases the range of geo-physical properties that can be used, but it is still necessary to have some that are related to the process described by a model parameter. Regarding the CFG, our results show that it is possible to include a wide range of different functions and function complexity in the CFG and still be able to search through the resulting Function Space. It is thus possible to define a (very) large space of possible functions for FSO, and therefore have a high probability of including the true or a sufficiently approximating function in it.

Knowing the restrictions due to equifinality and assumptions related to FSO, we could show that the multi-criteria test increases its predictive capability. FSO was able to estimate a transfer function which included the correct spatial predictors and had the exact same parameter field on the model scale as the true one. Therefore, similar to other studies that showed an increased model performance by using multi-criteria parameter estimation, we could demonstrate an improvement in the search for a transfer function. The only disadvantage resulting from using multi-criteria optimization is the increased complexity of the optimization task, which potentially increases the number of iterations needed.

Having shown the FSO performance in a synthetic setting, in future work we will apply this methodology to a more complex hydrological model using real runoff data. This will provide further insight in the predictive capabilities of FSO and the difficulties of estimating transfer functions in a real-world setting. Additionally, we plan to apply FSO to different regions and a large spectrum of geo-physical properties.

Acknowledgments, Samples, and Data

This work was funded by the Austrian Science Fund FWF, project number P 31213. We are very thankful to the VERBUND Trading GmbH for their support as well as interest in our work during the project. All programming was done in R (R Core Team, 2019), where the deep learning model development relied heavily on TensorFlow (Allaire & Tang, 2019) and Keras (Allaire & Chollet, 2019), and the visualizations on ggplot2 (Wickham, 2016).

The R code used to generate all results for this publication can be found under https://github.com/MoritzFeigl/FSO_paper. The data for geo-physical properties is available from Feigl et al., (2020) and the discharge data used in this study is available at https://www.ehyd.gv.at. The meteorological data from the INCA dataset cannot be made public, because the rights belong to the Zentralanstalt für Meteorologie und Geodynamik (ZAMG). It can be acquired from https://www.zamg.ac.at.

References

Abdulla, F. A., & Lettenmaier, D. P. (1997). Development of regional parameter estimation equations for a macroscale hydrologic model. *Journal of Hydrology*, *197*(1–4), 230–257. https://doi.org/10.1016/S0022-1694(96)03262-3

Allaire, J. J., & Chollet, F. (2019). keras: R Interface to "Keras." Retrieved from https://cran.r-project.org/package=keras

Allaire, J. J., & Tang, Y. (2019). tensorflow: R Interface to "TensorFlow." Retrieved from https://cran.r-project.org/package=tensorflow

Baroni, G., Schalge, B., Rakovec, O., Kumar, R., Schüler, L., Samaniego, L., et al. (2019). A Comprehensive Distributed Hydrological Modeling Intercomparison to Support Process Representation and Data Collection Strategies. *Water Resources Research*, *55*(2), 990–1010. https://doi.org/10.1029/2018WR023941

Beven, K. (2001). How far can we go in distributed hydrological modelling? *Hydrology and Earth System Sciences*, *5*(1), 1–12. https://doi.org/10.5194/hess-5-1-2001

Beven, Keith. (2006). A manifesto for the equifinality thesis. *Journal of Hydrology*, *320*(1–2), 18–36. https://doi.org/10.1016/J.JHYDROL.2005.07.007

Blöschl, G., Bierkens, M. F. P., Chambel, A., Cudennec, C., Destouni, G., Fiori, A., et al. (2019). Twenty-three unsolved problems in hydrology (UPH) – a community perspective. *Hydrological Sciences Journal*, *64*(10), 1141–1158. https://doi.org/10.1080/02626667.2019.1620507

Bongard, J., & Lipson, H. (2007). Automated reverse engineering of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences of the United States of America*,

104(24), 9943–8. https://doi.org/10.1073/pnas.0609476104

Buytaert, W., & Beven, K. (2009). Regionalization as a learning process. *Water Resources Research*, *45*(11). https://doi.org/10.1029/2008WR007359

Clark, M. P., Schaefli, B., Schymanski, S. J., Samaniego, L., Luce, C. H., Jackson, B. M., et al. (2016). Improving the theoretical underpinnings of process-based hydrologic models. *Water Resources Research*, *52*(3), 2350–2365. https://doi.org/10.1002/2015WR017910@10.1002/(ISSN)1944-9208.COMHES1

Clark, M. P., Bierkens, M. F. P., Samaniego, L., Woods, R. A., Uijlenhoet, R., Bennett, K. E., et al. (2017). The evolution of process-based hydrologic models: historical challenges and the collective quest for physical realism. *Hydrology and Earth System Sciences*, *21*(7), 3427–3440. https://doi.org/10.5194/hess-21-3427-2017

Cornforth, T. W., & Lipson, H. (2015). A hybrid evolutionary algorithm for the symbolic modeling of multiple-time-scale dynamical systems. *Evolutionary Intelligence*, *8*(4), 149–164. https://doi.org/10.1007/s12065-015-0126-x

Cukier, R. I., Levine, H. B., & Shuler, K. E. (1978). Nonlinear sensitivity analysis of multiparameter model systems. *Journal of Computational Physics*. https://doi.org/10.1016/0021-9991(78)90097-9

Le Cun, Y., & Fogelman-Soulié, F. (1987). Modèles connexionnistes de l'apprentissage. *Intellectica. Revue de l'Association Pour La Recherche Cognitive*, *2*(1), 114–143. https://doi.org/10.3406/intel.1987.1804

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

Devia, G. K., & Ganasri, B. P. (2015). A Review on Hydrological Models. *Aquatic Procedia*, *4*, 1001–1007. https://doi.org/10.1016/J.AQPRO.2015.02.126

Didan, K. (2015). MOD13Q1 MODIS/Terra vegetation indices 16-day L3 global 250m SIN grid V006. *NASA EOSDIS Land Processes DAAC*.

Feigl, M., Herrnegger, M., Klotz, D., & Schulz, K. (2020). Data for "Function Space Optimization: A symbolic regression method for estimating parameter transfer functions for hydrological models." Zenodo. https://doi.org/10.5281/zenodo.3676053

Francke, T., Baroni, G., Brosinsky, A., Foerster, S., López-Tarazón, J. A., Sommerer, E., & Bronstert, A. (2018). What Did Really Improve Our Mesoscale Hydrological Model? A Multidimensional Analysis Based on Real Observations. *Water Resources Research*, *54*(11), 8594–8612. https://doi.org/10.1029/2018WR022813

Francos, A., Elorza, F. J., Bouraoui, F., Bidoglio, G., & Galbiati, L. (2003). Sensitivity analysis of distributed environmental simulation models: Understanding the model behaviour in hydrological studies at the catchment scale. In *Reliability Engineering and System Safety* (Vol. 79, pp. 205–218). https://doi.org/10.1016/S0951-8320(02)00231-4

Frey, S., & Holzmann, H. (2015). A conceptual, distributed snow redistribution model. *Hydrology and Earth System Sciences*, *19*(11), 4517–4530. https://doi.org/10.5194/hess-19-

4517-2015

Gan, Y., Duan, Q., Gong, W., Tong, C., Sun, Y., Chu, W., et al. (2014). A comprehensive evaluation of various sensitivity analysis methods: A case study with a hydrological model. *Environmental Modelling and Software*, *51*, 269–285. https://doi.org/10.1016/j.envsoft.2013.09.031

Gan, Z., Pu, Y., Henao, R., Li, C., He, X., & Carin, L. (2018). Learning Generic Sentence Representations Using Convolutional Neural Networks, 2390–2400. https://doi.org/10.18653/v1/d17-1254

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., et al. (2018). Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Central Science*, *4*(2), 268–276. https://doi.org/10.1021/acscentsci.7b00572

Haiden, T., Kann, A., Wittmann, C., Pistotnik, G., Bica, B., Gruber, C., et al. (2011). The Integrated Nowcasting through Comprehensive Analysis (INCA) System and Its Validation over the Eastern Alpine Region. *Weather and Forecasting*, *26*(2), 166–183. https://doi.org/10.1175/2010WAF2222451.1

Haiden, T., Kann, A., & Pistotnik, G. (2014). Nowcasting with INCA During SNOW-V10. *Pure and Applied Geophysics*, *171*(1–2), 231–242. https://doi.org/10.1007/s00024-012-0547-8

Hattermann, F. F., Krysanova, V., Gosling, S. N., Dankers, R., Daggupati, P., Donnelly, C., et al. (2017). Cross-scale intercomparison of climate change impacts simulated by regional and global hydrological models in eleven large river basins. *Climatic Change*, *141*(3), 561–576. https://doi.org/10.1007/s10584-016-1829-4

Hengl, T., Mendes de Jesus, J., Heuvelink, G. B. M., Ruiperez Gonzalez, M., Kilibarda, M., Blagotić, A., et al. (2017). SoilGrids250m: Global gridded soil information based on machine learning. *PLOS ONE*, *12*(2), e0169748. https://doi.org/10.1371/journal.pone.0169748

Herrnegger, M., Nachtnebel, H. P., & Haiden, T. (2012). Evapotranspiration in high alpine catchments - An important part of the water balance! *Hydrology Research*, *43*(4), 460–475. https://doi.org/10.2166/nh.2012.132

Herrnegger, M., Senoner, T., & Nachtnebel, H. P. (2018). Adjustment of spatio-temporal precipitation patterns in a high Alpine environment. *Journal of Hydrology*, *556*, 913–921. https://doi.org/10.1016/j.jhydrol.2016.04.068

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Holland, J. (1975). Adaptation in natural and artificial systems. an introductory analysis with applications to biology, control and artificial intelligence. *Ann Arbor: University of Michigan Press, 1975*. Retrieved from http://adsabs.harvard.edu/abs/1975anas.book.....H

Huang, S., Eisner, S., Magnusson, J. O., Lussana, C., Yang, X., & Beldring, S. (2019). Improvements of the spatially distributed hydrological modelling using the HBV model at 1 km resolution for Norway. *Journal of Hydrology*, *577*, 123585. https://doi.org/10.1016/J.JHYDROL.2019.03.051

Huete, A., Didan, K., Miura, T., Rodriguez, E. ., Gao, X., & Ferreira, L. . (2002). Overview of the radiometric and biophysical performance of the MODIS vegetation indices. *Remote Sensing of Environment*, *83*(1–2), 195–213. https://doi.org/10.1016/S0034-4257(02)00096-2

Hundecha, Y., & Bárdossy, A. (2004). Modeling of the effect of land use changes on the runoff generation of a river basin through parameter regionalization of a watershed model. *Journal of Hydrology*, *292*(1–4), 281–295. https://doi.org/10.1016/J.JHYDROL.2004.01.002

Kay, A. L., Rudd, A. C., Davies, H. N., Kendon, E. J., & Jones, R. G. (2015). Use of very high resolution climate model data for hydrological modelling: baseline performance and future flood changes. *Climatic Change*, *133*(2), 193–208. https://doi.org/10.1007/s10584-015-1455-6

Kennedy, J., & Eberhart, R. (n.d.). Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks* (Vol. 4, pp. 1942–1948). IEEE. https://doi.org/10.1109/ICNN.1995.488968

Kingma, D. P., & Welling, M. (2013). Auto-Encoding Variational Bayes. Retrieved from http://arxiv.org/abs/1312.6114

Kirchner, J. W. (2006). Getting the right answers for the right reasons: Linking measurements, analyses, and models to advance the science of hydrology. *Water Resources Research*, *42*(3). https://doi.org/10.1029/2005WR004362

Klambauer, G., Unterthiner, T., Mayr, A., & Hochreiter, S. (2017). Self-Normalizing Neural Networks. Retrieved from http://arxiv.org/abs/1706.02515

Kling, H., & Nachtnebel, H. P. (2009). A method for the regional estimation of runoff separation parameters for hydrological modelling. *Journal of Hydrology*, *364*(1–2), 163–174. https://doi.org/10.1016/j.jhydrol.2008.10.015

Kling, H., Fürst, J., & Nachtnebel, H. P. (2006). Seasonal, spatially distributed modelling of accumulation and melting of snow for computing runoff in a long-term, large-basin water balance model. *Hydrological Processes*, *20*(10), 2141–2156. https://doi.org/10.1002/hyp.6203

Kling, H., Stanzel, P., Fuchs, M., & Nachtnebel, H.-P. (2015). Performance of the COSERO precipitation–runoff model under non-stationary conditions in basins with different climates. *Hydrological Sciences Journal*, *60*(7–8), 1374–1393. https://doi.org/10.1080/02626667.2014.959956

Klotz, D., Herrnegger, M., & Schulz, K. (2017). Symbolic Regression for the Estimation of Transfer Functions of Hydrological Models. *Water Resources Research*, *53*(11), 9402–9423. https://doi.org/10.1002/2017WR021253

Knuth, D. E. (1965). On the translation of languages from left to right. *Information and Control*, *8*(6), 607–639. https://doi.org/10.1016/S0019-9958(65)90426-2

Kullback, S., & Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics*, *22*(1), 79–86. https://doi.org/10.1214/aoms/1177729694

Kumar, R., Samaniego, L., & Attinger, S. (2013). Implications of distributed hydrologic model parameterization on water fluxes at multiple scales and locations. *Water Resources*

Research, *49*(1), 360–379. https://doi.org/10.1029/2012WR012195

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, *1*(4), 541–551. https://doi.org/10.1162/neco.1989.1.4.541

Van Looy, K., Bouma, J., Herbst, M., Koestel, J., Minasny, B., Mishra, U., et al. (2017). Pedotransfer Functions in Earth System Science: Challenges and Perspectives. *Reviews of Geophysics*, *55*(4), 1199–1256. https://doi.org/10.1002/2017RG000581

Lu, S., Zhu, Y., Zhang, W., Wang, J., & Yu, Y. (2018). Neural Text Generation: Past, Present and Beyond. Retrieved from http://arxiv.org/abs/1803.07133

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. Retrieved from http://arxiv.org/abs/1301.3781

Nachtnebel, H., Baumung, S., & Lettl, W. (1993). Abflussprognosemodell für das Einzugsgebiet der Enns und Steyer.

Nash, J. E., & Sutcliffe, J. V. (1970). River flow forecasting through conceptual models part I — A discussion of principles. *Journal of Hydrology*, *10*(3), 282–290. https://doi.org/10.1016/0022-1694(70)90255-6

Nijzink, R. C., Almeida, S., Pechlivanidis, I. G., Capell, R., Gustafssons, D., Arheimer, B., et al. (2018). Constraining Conceptual Hydrological Models With Multiple Information Sources. *Water Resources Research*, *54*(10), 8332–8362. https://doi.org/10.1029/2017WR021895

Parajka, J., Merz, R., & Blöschl, G. (2005). A comparison of regionalisation methods for catchment model parameters. *Hydrology and Earth System Sciences Discussions*, *9*(3), 157–171. Retrieved from https://hal.archives-ouvertes.fr/hal-00304814/

Perrin, C., Michel, C., & Andréassian, V. (2003). Improvement of a parsimonious model for streamflow simulation. *Journal of Hydrology*, *279*(1–4), 275–289. https://doi.org/10.1016/S0022-1694(03)00225-7

R Core Team. (2019). R: A Language and Environment for Statistical Computing. Vienna, Austria. Retrieved from https://www.r-project.org/

Rakovec, O., Kumar, R., Mai, J., Cuntz, M., Thober, S., Zink, M., et al. (2016). Multiscale and Multivariate Evaluation of Water Fluxes and States over European River Basins. *Journal of Hydrometeorology*, *17*(1), 287–307. https://doi.org/10.1175/JHM-D-15-0054.1

Ratto, M., Tarantola, S., & Saltelli, A. (2001). Sensitivity analysis in model calibration: GSA-GLUE approach. *Computer Physics Communications*, *136*(3), 212–224. https://doi.org/10.1016/S0010-4655(01)00159-X

Rechenraum e.U. (2012). Der oe3d Datensatz, basierend auf Daten © BEV, Aster, ein Produkt von METI and NASA, und OpenStreetMap, ist Creative Commons lizensiert (http://creativecommons.org/licenses/by-sa/3.0/at/). Retrieved from http://www.oe3d.at

Reusser, D. E., Buytaert, W., & Zehe, E. (2011). Temporal dynamics of model parameter sensitivity for computationally expensive models with the Fourier amplitude sensitivity test. *Water Resources Research*, *47*(7). https://doi.org/10.1029/2010WR009947

Samaniego, L., Kumar, R., & Attinger, S. (2010). Multiscale parameter regionalization of a grid-

based hydrologic model at the mesoscale. *Water Resources Research*, *46*(5). https://doi.org/10.1029/2008WR007327

Samaniego, L., Kumar, R., Thober, S., Rakovec, O., Zink, M., Wanders, N., et al. (2017). Toward seamless hydrologic predictions across spatial scales. *Hydrology and Earth System Sciences*, *21*(9), 4323–4346. https://doi.org/10.5194/hess-21-4323-2017

Schmidt, M., & Lipson, H. (2009). Distilling free-form natural laws from experimental data. *Science (New York, N.Y.)*, *324*(5923), 81–5. https://doi.org/10.1126/science.1165893

Schweppe, R., Thober, S., Attinger, S., & Samaniego, L. (2019). Development of a stand-alone Multiscale Parameter Regionalization (MPR) tool for the estimation of effective model parameters for any distributed model. In *Geophysical Research Abstracts* (Vol. 21, p. 12195).

Sherman, L. K. (1932). Stream flow from rainfall by the unit-hydrograph record: Eng. News-Record.

Srivastava, S., Shukla, A., & Tiwari, R. (2018). Machine Translation : From Statistical to modern Deep-learning practices. Retrieved from http://arxiv.org/abs/1812.04238

Stanzel, P., Kahl, B., Haberl, U., Herrnegger, M., & Nachtnebel, H. P. (2008). Continuous hydrological modelling in the context of real time flood forecasting in alpine Danube tributary catchments. *IOP Conference Series: Earth and Environmental Science*, *4*, 012005. https://doi.org/10.1088/1755-1307/4/1/012005

Stisen, S., McCabe, M. F., Refsgaard, J. C., Lerer, S., & Butts, M. B. (2011). Model parameter analysis using remotely sensed pattern information in a multi-constraint framework. *Journal of Hydrology*, *409*(1–2), 337–349. https://doi.org/10.1016/J.JHYDROL.2011.08.030

Stisen, S., Koch, J., Sonnenborg, T. O., Refsgaard, J. C., Bircher, S., Ringgaard, R., & Jensen, K. H. (2018). Moving beyond run-off calibration-Multivariable optimization of a surface-subsurface-atmosphere model. *Hydrological Processes*, *32*(17), 2654–2668. https://doi.org/10.1002/hyp.13177

Thornthwaite, C. W., & Mather, J. R. (1957). Instructions and Tables for Computing Potential Evapotranspiration and the Water Balance. Drexel Institute of Technology. Laboratory of Climatology. *Publications in Climatology*, *10*(3), 181–289.

Tolson, B. A., & Shoemaker, C. A. (2007). Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. *Water Resources Research*, *43*(1). https://doi.org/10.1029/2005WR004723

Wang, Y. (2012). Uncertain parameter sensitivity in Monte Carlo Simulation by sample reassembling. *Computers and Geotechnics*, *46*, 39–47. https://doi.org/10.1016/j.compgeo.2012.05.014

Wesemann, J., Herrnegger, M., & Schulz, K. (2018). Hydrological modelling in the anthroposphere: predicting local runoff in a heavily modified high-alpine catchment. *Journal of Mountain Science*, *15*(5), 921–938. https://doi.org/10.1007/s11629-017-4587-5

White, B. W., & Rosenblatt, F. (1963). Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. *The American Journal of Psychology*. https://doi.org/10.2307/1419730

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. Retrieved from https://ggplot2.tidyverse.org

Wijesekara, G. N., Gupta, A., Valeo, C., Hasbani, J.-G., Qiao, Y., Delaney, P., & Marceau, D. J. (2012). Assessing the impact of future land-use changes on hydrological processes in the Elbow River watershed in southern Alberta, Canada. *Journal of Hydrology*, *412–413*, 220–232. https://doi.org/10.1016/J.JHYDROL.2011.04.018

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q. V. (2019). XLNet: Generalized Autoregressive Pretraining for Language Understanding. Retrieved from http://arxiv.org/abs/1906.08237

Zink, M., Mai, J., Cuntz, M., & Samaniego, L. (2018). Conditioning a Hydrologic Model Using Patterns of Remotely Sensed Land Surface Temperature. *Water Resources Research*, *54*(4), 2976–2998. https://doi.org/10.1002/2017WR021346