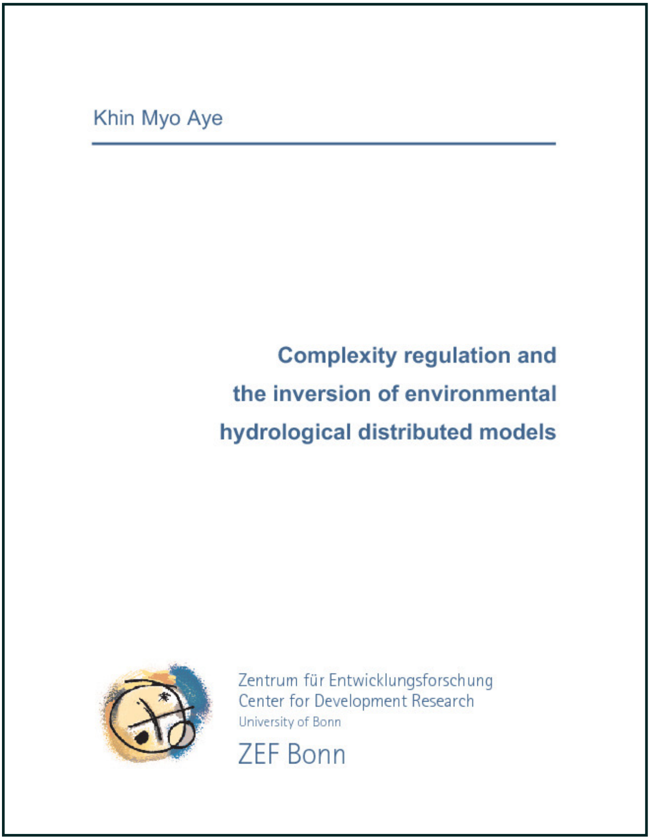
Myo Aye Khin (Autor)

Complexity regulation and the inversion of environmental hydrological distributed models



https://cuvillier.de/de/shop/publications/2019

1 INTRODUCTION

One main challenge in deriving suitable mathematical modelling for given natural systems like hydrological ones is to efficiently and robustly handle high-dimensional data. Beven(1997) mentions five main steps for such a modelling process including calibration and validation. After performing these five steps successfully, we can declare the model is acceptable. It is difficult, however, to construct a completely acceptable model in this sense.

Models are often given in terms of a mathematical equation or a system of mathematical equations relating state variables and different input parameters. Chiu (2004) discussed the rainfall-runoff model and water balance model for reservoirs. Abbott et al. (1989a), Abbott et al. (1989b), Beven (1997), Beven (2001) and De Roo et al. (2000) discussed rainfall-runoff models and Thornthwaite and Mather (1957) explained the development and evaluation of the water balance. Steenhuis and Molen (1986) discussed the Thornthwaite-Marther procedure for calculating recharge from the soil moisture balance. The code for these models was mostly written using the software in system-programming languages such as C^{++} and FORTRAN. In the present work, we derive such equations for the rainfall-runoff process based on the water balance equation (see Chapter 2). Then we write the code for these equations using one of the technical computing languages, MATLAB and run this code on a computer. We use the observed monthly data, precipitation and evapotranspiration, in accordance with the suggestion in Fennessey, 1995 (where it was stressed that accurate estimates of the reservoir-yield should be available from a monthly time step).

Another problem we are concerned with in our works is the question of handling the complexity of the systems involved. Because of the limitations on hydrological measurement techniques and measurements in space and time, the problem of reconstructing the coefficients entering a mathematical model is quite complicated. Hydrological models continue to grow in complexity. The increased complexity of hydrological models and the role of the many parameters in them are critically discussed in, e.g., Jakeman and Hornberger, 1993, however, this analysis does not have influenced general general modeling practice in a fundamental way. Hydrologists recognize that parameters in high-dimensional models may not be identifiable because of the underdeterminedness of the inverse problem (see e.g., Beven and Binley, 1992).

Until now, however, no general structural way to determine the optimal number of parameters in hydrological models has been found. Model complexity should not only match the complexity of the processes but also be commensurate with the available data. In order to find better ways to get the parameters really needed for handling a given hydrological problem, one needs a procedure to control complexity during model selection. We will examine the method for assessing the optimal dimension of a model for a set of data provided by the Center for Development Research, (ZEF) University of Bonn, and a set of data for a part of the

Irrawaddy river, which is one of the main rivers in Myanmar and also the longest. Artificial intelligence (AI) models on adaptive learning and models are usually purely formal (like polynomials neural networks) whereas in hydrological models as much physical information as possible is used. Important similarities for both types of models are that they involve very high dimensional parameter spaces with relatively small data sets and low dimensional predictions.

In the inverse problem known as model calibration, the model parameter values are usually determined by using one of the optimization algorithms. Different optimization algorithms for the rainfall-runoff modeling are mentioned in, e.g., Beven, 1997. In the present work we discuss a new model for the rainfall-runoff process for a system of one or more reservoirs. It is given in terms of nonlinear equations. Our interest is to find a suitable optimization method for the inversion of our nonlinear rainfall-runoff model for a system consisting of a large number of reservoirs using MATLAB. We thus determine these parameters using the downhill simplex method proposed by Nelder and Mead (1965) and reported by Press et al. (1992) with the aid of mathematical programming (see Chapter 3). We found that the proposed method is simple and robust although it may not be the most efficient optimization method.

An important part of the modelling process is to determine whether the acquired model is acceptable. For this purpose we need to validate the observed data. The cross-validation method (see e.g., Kohavi, 1995, Cherkassky and Mulier, 1998 and Lendasseet al., 2003) is one of the most common complexity regulation methods for estimating the empirical out of sample error, i.e., the "generalization error". In the validation, we split these data into two parts, learning data and validation data. Learning data are used to estimate the model parameters and validation data are used to evaluate the model performance. Although model validation already gives a good indication of how well a model works, it is difficult in general to obtain a perfect fit between observed and predicted variables.

This study is organized as follows: In Chapter 1, the system of equations for the rainfall-runoff process including a nonlinear rainfall-runoff model is derived from the water balance equation. We use the observed rainfall as the input data and the runoff and the actual evapotranspiration, which is calculated from the observed potential evapotranspiration, as the output data. We first normalize these input data and output data by dividing them by the maximum storage in the finite time interval and perform the computations of storage and runoff using the approach for the actual evapotranspiration discussed in Thornthwaite and Mather (1957) is used. We then continue these computations using the monthly observed data.

In Chapter 2, the mathematical properties of these models are investigated. The continuity and monotonicity of the solutions for the storage and the runoff are shown. The uniqueness of the minimum storage for which the runoff occurs in at least one month is proved. Then we show that the runoff which is strictly decreasing and continuous does not have an isolated minimum of the maximum storage different from the selected maximum storage.

In the Chapter 3, the inversion of the nonlinear rainfall-runoff model is numerically performed. The mean square error (MSE) is used as the objective function in its inversion. The downhill simplex method, which is used for minimizing the MSE, is described. The construction of the initial simplex that is suitable for our problem is described. A sufficient condition for the convergence of the minimization algorithm is mentioned. Numerical experiments on its inversion are done using two objective functions with noise and without noise. In these experiments, the data for three different periods, i.e., 2, 5 and 10 hydrological years are separately used. We use the system of ℓ reservoirs, $\ell = 1, 2, \dots, 5$, with the same maximum storages and with different maximum storages. In the case of minimizing the objective function with noise, we use three different kinds of Gaussian noise, with mean 1 and the standard deviations 5%, 10%, and 20% of the runoff. Though the MSE is a variance between the observed runoff and the model runoff, the runoff generated from the forward model is, in the present work, used instead of the observed runoff. This is so because the observed runoff contains other parameters which are not included in our model. The runoff from the forward model for 1 reservoir is fitted with the runoff from each of ℓ reservoirs, $\ell = 1, 2, \dots, 5$. Similarly, each of the runoffs from the forward model for 2, 3, 4 and 5 reservoirs is fitted with the runoff for any number ℓ of reservoirs. In the particular case of 2 groups, one with 2 reservoirs and the other with 3 reservoirs, we do the fitting using any of 10 reservoirs.

In the Chapter 4, the model is validated using the k-fold cross-validation method. The algorithm for this method is first described. Then we cross-validate the data for 5 hydrological years, and divide this data set into five independent subsets where the initial storage for each year is set as zero. We use each of them as a validation set and the union of the other four subsets as a learning set. For the cross-validation it is necessary that the MSE for each learning set is minimized. The runoff in the forward mode is generated for systems of ℓ reservoirs, $\ell = 1, 2, \ldots, 5$. This generated runoff value for each reservoir in relation with any number ℓ of reservoirs, $\ell = 1, 2, \ldots, 5$ is fitted with the model for ℓ numbers of reservoirs. We also consider two systems having two reservoirs and three reservoirs fitted with each of the systems having the number from one reservoir to ten reservoirs.

Then, we draw a conclusion over all chapters and give indications for further investigations.

The appendix to Chapter 2 contains the flow-diagram and the annotated codes of MATLAB programs for calculating the storage, the runoff and the actual evapotranspiration of one reservoir with the appropriate maximum storage from our models. In the appendix to Chapter 4 the flow diagrams and the annotated codes for the inversion of our model are described. The flow diagrams and the annotated codes for the application of the cross-validation method are mentioned in the appendix to Chapter 5.

2 DERIVATION OF A SIMPLE NONLINEAR RAINFALL-RUNOFF MODEL

The various mathematical models in hydrological modeling can be represented commonly by equations or a set of equations relating state variables and parameters. A mathematical model of a given system generally combines certain common physical and/or experimental rules along with certain specific features of the system. In order to model a real hydrological system, both a forward and an inverse problem needs to be solved. In forward problems, solutions in terms of state variables when the parameters are given are searched for, while in inverse problems, parameters from measurements of the state variables are determined.

The modelling of rainfall-runoff because of interest in hydrology and has been developed since 1986. Nowadays hydrologists strive at combining mathematical modelling with efficient computer simulation often with own software, in system programming languages such as C^{++} and FORTRAN. The increased complexity of hydrological models and the role of the many parameters used in them are critically discussed in e.g., Jakeman and Hornberger, 1993.

In this study, a particular model representing a fairly large reservoir was developed. It was assumed that there is a fixed amount of storage in the soil and that runoff only occurs when the storage exceeds the maximum amount of the storage. The storage is filled by precipitation and emptied by evapotranspiration and the actual runoff. The general behavior is that there is a small loss (actual evapotranspiration), which asymptotically empties the reservoir if there is no further precipitation. This is a typical hydrological reaction, and once the system is full, runoff is very high, frequently causing floods. The model belongs to a class of rainfall-runoff models discussed in the hydrological literature, e.g., Abbott et al. (1989a), Abbott et al. (1989b), Beven (1997), Beven (2001) and De Roo et al. (2000). In the present form it has been discussed in Thornthwaite and Mather (1957) and Steenhuis and Molen (1986). discussed rainfall-runoff models and

2.1 Equation of water balances

We consider a model for a large reservoir with the amount of storage at time $t \in [0, \infty)$, denoted by $V_S(t)$, Thus $V_S : [0, \infty) \to [0, \infty)$.

Let $V_x \geq 0$ be the maximum storage available in the reservoir. x denotes the length (depth) of the maximal storage and is independent of $t \in [0, \infty)$.

Similarly, we define the functions V_P, V_{E_a}, V_E resp. V_Q, from $[0, \infty)$ into $[0, \infty)$, which mean the volume of the precipitation, the actual evapotranspiration, the potential evapotranspiration and the runoff at time t, respectively.

According to the law of conservation, we have

$$V_P(t) = V_S(t) + V_Q(t) + V_{E_a}(t). \tag{2.1}$$

In the hydrological literature, it is customary to measure most quantities in millimeters per unit area. Given the area A (a positive constant) of the reservoir, we can transform volumes to volumes per area, a more conventional measure for precipitation, as follows:

$$h_\alpha(t) \equiv \frac{V_\alpha(t)}{A}, \tag{2.2}$$

where α stands for $S, x, P, E_a, E,$ and Q, respectively. We remark that, by definition of x, V_x:

$$h_x = \frac{V_x}{A} = x \tag{2.3}$$

(independent of $t \in [0, \infty)$).

It is assumed that these quantities, $h_\alpha(t)$ are differentiable functions of the time variable $t \geq 0$; $\alpha(t)$ is denoted by the derivative of $h_\alpha(t)$, thus

$$\alpha(t) = \frac{dh_\alpha(t)}{dt}, \tag{2.4}$$

α as before, $\alpha \neq S$.

Those customary abbreviations represent the quantities in a one-reservoir rainfall-runoff model with evaporation. In particular we have the following terms:

$E_a\,(t)$	:	Rate of actual evapotranspiration (mm per month) per unit area as a function of time $t \geq 0$
$E\,(t)$	:	Rate of potential evapotranspiration (mm per month) per unit area as a function of time $t \geq 0$
$P\,(t)$	:	Rate of precipitation (mm per month) per unit area as a function of time $t \geq 0$
$Q\,(t)$	:	Runoff (mm per month) per unit area as a function of time $t \geq 0$.

To combine runoffs in mm per unit area from subcatchments into a runoff in mm for the total catchment we need to know the relevant areas and to go back and forth between values per unit area and volumes.

The rate of actual evapotranspiration, $E_a(t)$ is assumed to be proportional to the relative storage ($\frac{S(t)}{x}, x > 0$) and the rate of the (fixed or given) potential evapotranspiration, $E(t)$,(see, eg., Thornthwaite and Mather, 1957) namely:

$$E_a(t) = \frac{S(t)}{x} E(t), \tag{2.5}$$

where $S\,(t)$ stands for $h_s(t)$ and is the actual amount of storage used in mm per unit area as a function of time $t \geq 0$, while x is the total amount of storage available in the system in mm per unit area. The law of conservation (2.1) becomes

$$\frac{dh_S}{dt} = \frac{dh_P}{dt} - \frac{dh_Q}{dt} - \frac{dh_{E_a}}{dt}. \tag{2.6}$$

By substituting the equations ((2.4)- (2.5)) in the equations (2.6), we get the mass conservation equation:

$$\frac{dS(t)}{dt} = P(t) - Q(t) - E_a(t). \tag{2.7}$$

2.2 Derivation of a simple nonlinear rainfall-runoff model

We will model the behavior of the reservoir each month separately using data averaged per month. Let t_k be the initial time for the k^{th} month. On the interval $(t_k, t_{k+1}], t_k, t_{k+1} \in [0, \infty)$, we introduce the following dimensionless variables:

$$\tau = \frac{t - t_k}{t_{k+1} - t_k} \tag{2.8}$$

$$\varsigma = \frac{V_S}{V_x} \tag{2.9}$$

$$\upsilon_\alpha = \frac{V_\alpha}{V_x} \tag{2.10}$$

for $\alpha = P, Q, E_a$, and E; $\tau \in (0, 1]$.

In these variables, the mass conservation equation (2.7) takes the form

$$\upsilon_P = \varsigma + \upsilon_{E_a} + \upsilon_Q. \tag{2.11}$$

This implies that

$$p = \xi + q + e_a \tag{2.12}$$

where

$$\xi \equiv \frac{d\varsigma}{d\tau}, \quad \text{the scaled rate of storage} \tag{2.13}$$

$$p \equiv \frac{d\upsilon_P}{d\tau}, \quad \text{the scaled rate of precipitation} \tag{2.14}$$

$$q \equiv \frac{d\upsilon_Q}{d\tau}, \quad \text{the scaled rate of runoff} \tag{2.15}$$

and

$$e_a \equiv \frac{d\upsilon_{E_a}}{d\tau}, \quad \text{the scaled rate of actual evapotranspiration,} \tag{2.16}$$

$\tau \in (0, 1]$.

However, from (2.5), we get $e_a = \varsigma e$, where $e \equiv \frac{d\upsilon_E}{d\tau}$, is the scaled rate of potential evapotranspiration.

Therefore

$$\frac{d\upsilon_{E_a}}{d\tau} = \varsigma e. \tag{2.17}$$

We assume that p and e depend only on the particular month considered and are constant during any given month. For month $k \in \mathbb{N}_0 = \{0, 1, 2, \ldots\}$, we set

$$p_k = \frac{V_P(t_{k+1}) - V_P(t_k)}{V_x} \tag{2.18}$$

and

$$e_k = \frac{V_E(t_{k+1}) - V_E(t_k)}{V_x}. \tag{2.19}$$

Next, we know that if the storage exceeds the maximum storage ($V_S \geq V_x$, i.e., $\varsigma \geq 1$), the runoff occurs and otherwise ($\varsigma < 1$), there is no runoff. Then, for the month k, we define

$$\xi_k(\tau) = \frac{d\varsigma_k(\tau)}{d\tau} \equiv \begin{cases} p_k - \varsigma_k(\tau)e_k & \text{if} \quad \varsigma_k(\tau) < 1 \\ 0 & \text{if} \quad \varsigma_k(\tau) \geq 1 \end{cases} \tag{2.20}$$

and

$$q_k(\tau) \equiv \begin{cases} 0 & \text{if} \quad \varsigma_k(\tau) < 1 \\ p_k - e_k & \text{if} \quad \varsigma_k(\tau) \geq 1. \end{cases} \tag{2.21}$$

Let $s_k \in \mathbb{R}^+, k \in \mathbb{N}$ be given values, and let $\varsigma_k : (0, 1] \to \mathbb{R}^+$ be a continuous function with the boundary conditions $\varsigma_k(0) = s_{k-1}$ and $\varsigma_k(1) = s_k$. Let ζ_k be the solution of the linear ordinary differential equation

$$\frac{d\zeta_k(\tau)}{d\tau} = p_k - \zeta_k(\tau)e_k, \quad \tau \in (0, 1] \tag{2.22}$$

with boundary condition $\zeta_k(0) = s_{k-1}$.

Then

$$\zeta_k(\tau) = s_{k-1}\exp(-e_k\tau) + \frac{p_k}{e_k}[1 - \exp(-e_k\tau)], \quad e_k \neq 0. \tag{2.23}$$

(For $e_k = 0$, we simply have $\zeta_k(\tau) = s_{k-1}$, which only satisfies the boundary condition at $\tau = 0$ for $s_{k-1} \in \mathbb{R}^+$.) Moreover, p_k and e_k can be rewritten as

$$p_k = \frac{P_k}{x}$$

and

$$e_k = \frac{E_k}{x}$$

respectively.

In terms of the original parameters, P_k and E_k, (2.23) becomes

$$\zeta_k(\tau) = s_{k-1}\exp(-a_k\tau) + B_k[1 - \exp(-a_k\tau)] \tag{2.24}$$

where $\frac{E_k}{x} = a_k$ and $\frac{P_k}{E_k} = B_k$.

Let $\widetilde{\varsigma}_k(\tau)$ be defined by

$$\widetilde{\varsigma}_k(\tau) = \begin{cases} \zeta_k(\tau) & \text{if} \quad \varsigma_k(\tau) < 1 \\ 1 & \text{if} \quad \varsigma_k(\tau) \geq 1. \end{cases} \tag{2.25}$$

From (2.17), we get

$$V_{E_a}(t_{k+1}) - V_{E_a}(t_k) = V_x \int_0^1 \widetilde{\varsigma}_k(z) e_k(z) dz. \tag{2.26}$$

Furthermore, we define the physical storage volume for a given time τ by

$$V_S(t_k + \tau[t_{k+1} - t_k]) = \widetilde{\varsigma}_k(\tau) V_x \tag{2.27}$$

and the physical discharge Q by

$$Q_S(t_k + \tau[t_{k+1} - t_k]) = \frac{V_x}{t_{k+1} - t_k} q_k(\tau) \tag{2.28}$$

(q_k being defined as in (2.15).) Then we obtain