Hydrology and Earth System Sciences

Teaching hydrological modeling with a user-friendly catchment-runoff-model software package

J. Seibert[1,2,3] and M. J. P. Vis[1]

[1]Department of Geography, University of Zurich, Winterthurerstr. 190, 8057 Zurich, Switzerland
[2]Department of Earth Sciences, Uppsala University, Villavägen 16, 752 36 Uppsala, Sweden
[3]Department of Physical Geography and Quaternary Geology, Stockholm University, 10691 Stockholm, Sweden

Correspondence to: J. Seibert (jan.seibert@geo.uzh.ch)

Abstract. Computer models, especially conceptual models, are frequently used for catchment hydrology studies. Teaching hydrological modeling, however, is challenging, since students have to both understand general model concepts and be able to use particular computer programs when learning to apply computer models. Here we present a new version of the HBV (*Hydrologiska Byråns Vattenavdelning*) model. This software provides a user-friendly version that is especially useful for education. Different functionalities, such as an automatic calibration using a genetic algorithm or a Monte Carlo approach, as well as the possibility to perform batch runs with predefined model parameters make the software interesting especially for teaching in more advanced classes and research projects. Different teaching goals related to hydrological modeling are discussed and a series of exercises is suggested to reach these goals.

1 Introduction

There are good reasons to include modeling in hydrology education. First of all, models have become standard tools to address many types of practical hydrological questions and most hydrology students will be in contact with hydrological models in some way in their professional life. Therefore, it is important that hydrology students learn how to use models, get a general understanding of modeling concepts and realize possibilities as well as limitations of hydrological modeling (Wagener and McIntyre, 2007). Furthermore, models can contribute to a better understanding of hydrological variables and their interactions in a quantitative way. By "playing around" with a model, changing parameter values and looking at model simulations, students can explore interactions and feedback mechanisms, such as how soil moisture can influence evaporation and how this in turn can influence runoff.

Conceptual models are especially useful for education (AghaKouchak and Habib, 2010; Wagener and McIntyre, 2007). They are a good compromise between black-box models, which do not allow processes to be readily transparent, and physically-based models, which are usually too complex to be easily applied and understood by students. As the available time is often limited in hydrology courses, the use of (physically-based) models, which require a significant amount of time to learn how to use the model, is not possible in practice. Furthermore, the usually short run-time of conceptual models allows approaches based on a large number of model runs to be included in student exercises (AghaKouchak et al., 2012; Wagener and McIntyre, 2007). Conceptual catchment models also continue to be used as important tools in various kinds of hydrological applications. While there are applications where more complex, fully-distributed, physically-based models are needed, lumped or semi-distributed conceptual models have several advantages, such as more moderate requirements for data to set up and run the model.

One widely-used conceptual model is the HBV model (Bergström, 1976, 1992, 1995; Lindström et al., 1997). The HBV model is named after the *Hydrologiska Byråns Vattenavdelning* unit at the Swedish Meteorological and Hydrological Institute (SMHI), where its development started in the 1970s. The HBV model has become widely used and exists in several versions. The version HBV-light was developed at

Uppsala University in 1993 using Microsoft Visual Basic and has become widely used in education at several universities. The software has also been used for many research projects both in our and other groups (e.g., Konz and Seibert, 2010; Seibert and Beven, 2009; Steele-Dunne et al., 2008).

Here we present a new version of this model software, which has recently been developed at the University of Zurich. Compared to the previous version of HBV-light (Seibert, 1997, 1999; Seibert and Beven, 2009), in the new version it is also possible to run simulations with different time steps and several subcatchments. In addition, a routine for simulation of glaciers has been implemented (Konz and Seibert, 2010). The main motivation for this model version was to provide a user-friendly implementation for education. We discuss experiences from many years of using the HBV model in university education and describe a series of exercises for courses at different levels of teaching, from a first introductory use of the model to model applications for typical hydrological questions and programming model routines. The main goals of using a model like HBV-light in teaching are to be able to answer the following questions: (1) What is a runoff model and how does it work? (2) How can a model be applied to answer a specific question? (3) How do the model routines really work? (4) What are opportunities and limitations when using a model?

2 HBV model structure

The HBV model is a semi-distributed model, which means that a catchment can be separated into different elevation and vegetation zones as well as into different subcatchments (Fig. 1). The model consists of different routines and simulates catchment discharge, usually on a daily time step, based on time series of precipitation and air temperature as well as estimates of monthly long-term potential evaporation rates. In the snow routine snow accumulation and snowmelt are computed by a degree-day method. In the soil routine groundwater recharge and actual evaporation are simulated as functions of actual water storage. In the response (or groundwater) routine, runoff is computed as a function of water storage. Finally, in the routing routine a triangular weighting function is used to simulate the routing of the runoff to the catchment outlet. The central equations of the HBV model are given below; more detailed descriptions of the model can be found elsewhere (Bergström, 1995; Lindström et al., 1997; Seibert, 1999).

HBV-light uses a warm-up period during which state variables evolve from standard initial values to their appropriate values according to meteorological conditions and parameter values. One year of warm-up is found to be sufficient in most cases. Precipitation is considered to be either snow or rain, depending on whether the temperature is above or below a threshold temperature, P_{TT} (°C). All precipitation falling during time steps when the temperature is below P_{TT}, i.e.,
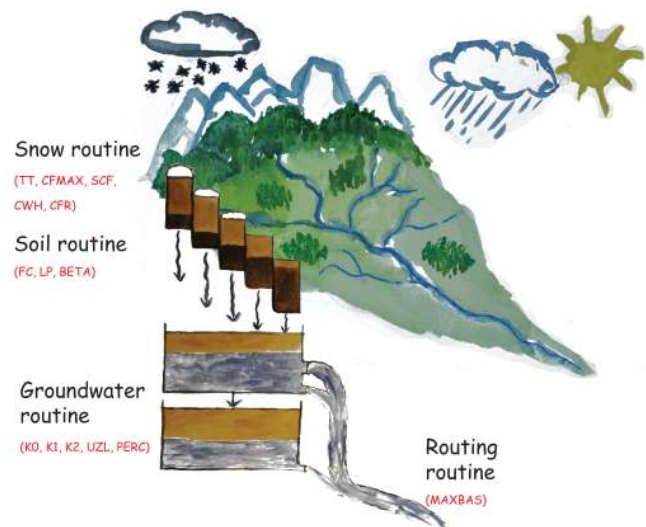

Fig. 1. Schematic structure of the HBV model.

simulated to be snow, is multiplied by a snowfall correction factor, P_{SCF} (−). This factor compensates for systematic errors in the snowfall measurements and for evaporation from the snowpack in the model, which is not simulated explicitly. Snowmelt, M (mm d^{-1}), is calculated with the degree-day method using the degree-day factor P_{CFMAX} (mm d^{-1} °C^{-1}) (Eq. 1) (note: here and in the following the units are given for the daily time step, although the model can be run also for other time steps). Meltwater and rainfall are retained within the snowpack until they exceed a certain fraction, P_{CWH} (−), of the water equivalent of the snow. When temperatures drop below P_{TT}, the amount of refreezing liquid water within the snowpack, R (mm d^{-1}), is computed using a refreezing coefficient, P_{CFR} (−) (Eq. 2). The effect of north- and south-facing slopes can optionally be considered using one parameter by which the potential melt is multiplied for the south-facing fraction of the catchment and divided for the north-facing fraction (Hottelet et al., 1993; Konz and Seibert, 2010).

$$M = P_{CFMAX} \cdot (T(t) - P_{TT}) \tag{1}$$

$$R = P_{CFR} \cdot P_{CFMAX} \cdot (P_{TT} - T(t)) \tag{2}$$

Based on the amount of input to the soil (sum of rainfall and snowmelt) at a certain time step, $I(t)$ (mm d^{-1}), the flux to the groundwater, $F(t)$ (mm d^{-1}), is computed; the remaining part of $P(t)$ is added to the soil box. The partition is a function of the ratio between current water content of the soil box ($S_{SOIL}(t)$, mm) and its maximum value (P_{FC}, mm) (Eq. 3). Actual evaporation from the soil box equals the potential evaporation if S_{SOIL}/P_{FC} is above $P_{LP} P_{FC}$, while a linear reduction is used when S_{SOIL}/P_{FC} is below this value (Eq. 4).

$$\frac{F(t)}{I(t)} = \left(\frac{S_{\mathrm{SOIL}}(t)}{P_{\mathrm{FC}}} \right)^{P_{\mathrm{BETA}}} \tag{3}$$

$$E_{\mathrm{act}} = E_{\mathrm{pot}} \cdot \min \left(\frac{S_{\mathrm{SOIL}}(t)}{P_{\mathrm{FC}} \cdot P_{\mathrm{LP}}}, 1 \right) \tag{4}$$

Groundwater recharge is added to the upper groundwater box (S_{UZ}, mm). P_{PERC} (mm d^{-1}) defines the maximum percolation rate from the upper to the lower groundwater box (S_{LZ}, mm). Runoff from the groundwater boxes is computed as the sum of two or three linear outflow equations (P_{K0}, P_{K1} and P_{K2}, d^{-1}, depending on whether S_{UZ} is above a threshold value, P_{UZL} (mm), or not (Eq. 5). This runoff is finally transformed by a triangular weighting function defined by the parameter P_{MAXBAS} (Eq. 6) to give the simulated runoff (mm d^{-1}).

$$Q_{\mathrm{GW}}(t) = P_{\mathrm{K2}} \cdot S_{\mathrm{LZ}} + P_{\mathrm{K1}} \cdot S_{\mathrm{UZ}}$$
$$+ P_{\mathrm{K0}} \cdot \max (S_{\mathrm{UZ}} - P_{\mathrm{UZL}}, 0) \tag{5}$$

$$Q_{\mathrm{sim}}(t) = \sum_{i=1}^{P_{\mathrm{MAXBAS}}} c(i) \cdot Q_{\mathrm{GW}}(t - i + 1),$$
$$\text{where } c(i) = \int_{i-1}^{i} \frac{2}{P_{\mathrm{MAXBAS}}} - \left| u - \frac{P_{\mathrm{MAXBAS}}}{2} \right| \cdot \frac{4}{P_{\mathrm{MAXBAS}}^2} \mathrm{d}u. \tag{6}$$

The long-term mean values of the potential evaporation, $E_{\mathrm{pot},M}$, for a certain day of the year are corrected to its value at day t, $E_{\mathrm{pot}}(t)$, by using the deviations of the temperature, $T(t)$, at a certain day, from its long-term mean, T_M, and a correction factor, P_{CET} ($^{\circ}$C^{-1}) (Eq. 7) (Lindström and Bergström, 1992).

$$E_{\mathrm{POT}}(t) = (1 + P_{\mathrm{CET}} \cdot (T(t) - T_M)) \cdot E_{\mathrm{POT},M}, \tag{7}$$

but $0 \leq E_{\mathrm{POT}}(t) \leq 2 \cdot E_{\mathrm{POT},M}$.

For catchments with glaciers there is a simple glacier routine (Konz and Seibert, 2010). For the catchment fraction covered by glacier ice once the snowpack has melted away, ice melt is simulated using Eq. (1) with the degree-day factor being increased by a factor representing the higher melting of ice compared to snow due to the lower albedo.

Besides the standard version several alternative model variants can be chosen in HBV-light. For instance, instead of the two linear outflows from the upper groundwater box, one non-linear outflow can be used (Eq. 8).

$$Q_{\mathrm{GW}}(t) = P_{\mathrm{K2}} \cdot S_{\mathrm{LZ}} + P_{\mathrm{K1}} \cdot S_{\mathrm{UZ}}^{1+P_{\mathrm{ALPHA}}} \tag{8}$$

The structure of the groundwater boxes can also be changed (Uhlenbrook et al., 1999). In the one-box variant there is only one groundwater box, with the upper two outflows being active only when the storage is above certain threshold values. In the three-box variant there are three linear-outflow boxes

above each other, and there are two parameters determining the maximum flow rate down to the next box. In yet another variant the simulated recharge from the soil routine is divided into two parts based on a relative portion determined by one parameter. One part is added directly to a linear storage, whereas the other part is evenly distributed over a subsequent period of a certain number of time steps and added to a second, parallel linear storage. This latter variant has been useful in catchments with deeper groundwater flow pathways (Seibert, 2000; Seibert et al., 2010).

It is not entirely obvious for which model routines computations should be performed lumped for the entire (sub)catchment or separately for each elevation/vegetation zone, and this is solved differently in the various HBV versions. In the standard variant (Lindström et al., 1997), calculations for the snow and soil routines are performed separately for each elevation/vegetation zone, whereas this is only done for the snow routine in the version developed at ETH Zurich (Braun and Renner, 1992). For a catchment in Germany, slightly improved results were obtained when also computing the water storage and flow for the upper groundwater box separately for each elevation/vegetation zone (Uhlenbrook et al., 1999). All these variants can be chosen in HBV-light.

It is difficult to provide general guidelines on which model variant and which setup, such as the number of elevation zones, to use. For simplicity, it can be recommended to start with the easiest case of the standard model structure and only one vegetation zone. Different elevation zones are strongly recommended if temperature differences in the catchment due to elevation differences are important. Typically, one elevation zone can span over about 100 m. For teaching, in most cases such a setup is suitable, whereas for research projects one might want to test and compare different variants and their performance in more detail.

3 HBV-light software

The HBV-light software is freely available and can be downloaded from http://www.geo.uzh.ch/en/units/h2k/hbv-model. From the very beginning guiding principles in the development of the HBV-light software were a focus on the core model, a user-friendly graphical user interface (GUI), the possibility to perform uncertainty analyses and making the software freely available for research and educational use.

3.1 Technical implementation

The new HBV-light software is built on the .NET Framework 3.5. The core model functionality is contained within the HBV-light Dynamic Link Library (.dll) file. There are two different executable programs, which are used to call methods in this dll. HBV-light-GUI provides the user with a graphical user interface to interact with the model, and

HBV-light-CLI is a command line interface allowing the user to run HBV-light from the command line or other applications such as PEST, which is a computer program for model-independent parameter estimation and uncertainty analysis (Doherty and Johnston, 2003; Doherty and Skahill, 2006). With HBV-light-CLI it is also possible for the user to easily program tools for repetitive simulation tasks. HBV-light-GUI uses the Microsoft Chart Controls for Microsoft .NET Framework 3.5 for the visualization of the simulations. Furthermore, a Microsoft Compiled HTML Help file (.chm) is included, providing the user with information about the usage of the graphical user interface. The help file also contains a detailed description of the HBV-light model structure. Help on the usage of the HBV-light-CLI executable is provided via the command line.

3.2   Software functionality

For detailed technical information on the use of the software, such as input data file formats, we refer to the help file included in the HBV-light software. A short overview of the software functionality is given here. After having prepared the input data files outside the HBV-light software adhering to a specified format, a catchment data set can be opened in HBV-light. The user can specify catchment and model settings as well as parameter values, and run the model simulation. Results will be written to output files and, for the HBV-light GUI version, graphs are generated of the results. Besides running a single model simulation, there are a few additional simulation tools available, which are important features of the HBV-light software. Batch simulations can be used to run the model for a list of predefined parameter sets. Furthermore, there are two different tools available for automatic calibration of the model, Monte Carlo simulations and Genetic Algorithm and Powell optimization (GAP). Monte Carlo simulations can be used to run a large number of simulations based on randomly selected parameter sets (within user-defined parameter boundaries). Objective functions, such as the Nash–Sutcliffe model efficiency coefficient, are computed for each model run and can be used to rank the different parameter sets based on their performance. The GAP algorithm consists of two steps (Seibert, 2000). First, optimized parameter sets are generated by an evolutionary mechanism of selection and recombination of a set of initial, randomly selected parameter sets (again within user-defined parameter boundaries). During the second step, parameter sets are fine-tuned using Powell's quadratically convergent method as described by Press et al. (2002).

4   HBV-light for education

HBV-light is especially useful for education, because the user-friendly interface makes the use of the model intuitive and little time is needed to learn how to run the model.

The previous version of HBV-light has been used in various courses and for thesis projects at several universities. First tests demonstrated that the new version is even more suitable for education because of its improved GUI. Students were quickly able to run the model with little instruction. The possibility to display parameter values and simulated time series at the same time in one window (Fig. 2) is very helpful when discussing model parameters and simulations in front of the screen.

5   Model exercises

In the following, we present a collection of exercises. These range from simple calibration exercises to the analysis of land-use changes, calculation of design floods and uncertainty estimation. Additionally, exercises are suggested that go beyond the use of the existing software; here students are asked to program or even develop their own model routines. The first exercises are suitable for basic hydrology courses, whereas the latter are more applicable for teaching on an advanced level in modeling courses where issues like model calibration, uncertainty estimation and model development are important. Student instructions for all exercises are provided in the Appendix. Wherever possible it is recommended to adapt these exercises by using catchments familiar to the students. In the first exercises most of the time is spent on manual calibration of the model. Although models are seldom calibrated manually in real applications nowadays, we believe manual calibration is a suitable way to become familiar with the model and to learn about the different parameters, their sensitivity and interactions.

The order of the exercises described below follows the idea of first providing students with an existing model and software package, allowing the model to be run quickly and "playing" with it. After the students become familiar with running the model, they are asked to do more advanced tasks and, finally, to develop their own model. An alternative would be to start with the development of their own model. The argument for this teaching strategy is to confront the students early on with important modeling issues, such as the many decisions one has to take when developing a model and the corresponding software, and to let them develop their own perceptual models based on their own decisions, before they become too influenced by existing models. After having programmed their own model, the students will also better understand how a model works when they use an existing model afterwards. Using the approach to start with an existing model, the students might not fully understand what a model is during the first exercises. However, things will become clearer in later exercises, which might include programming their own snow routine, testing different model routines or developing their own routines. With this approach the risk of getting stuck in the technical details and programming issues is smaller, and overwhelming the student in the beginning is avoided. The alternative approach
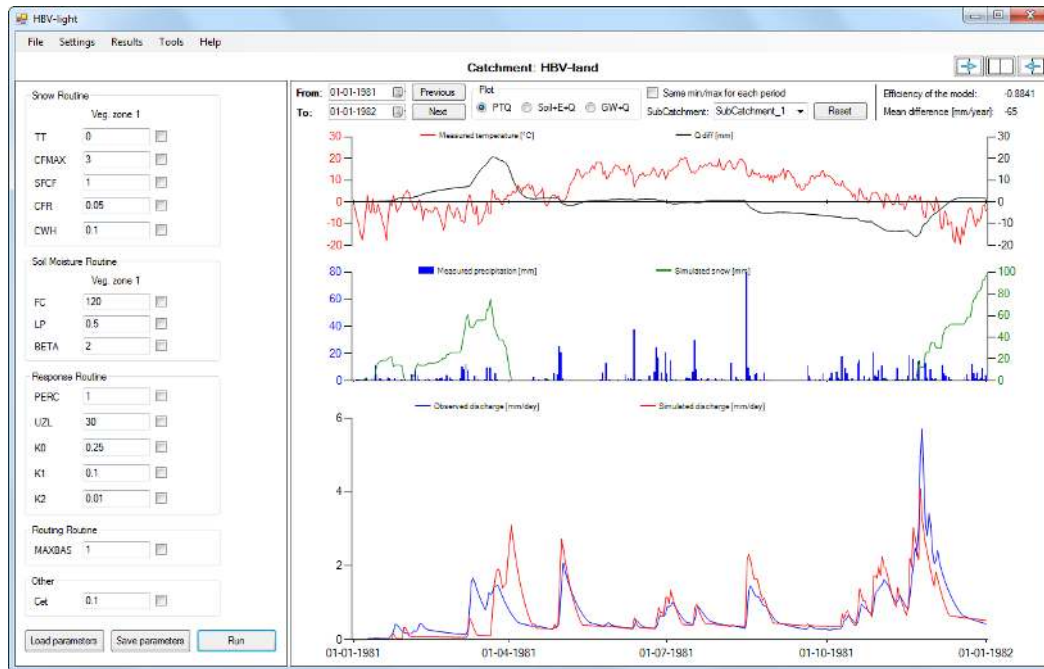
Fig. 2. Screenshot of the main window in HBV-light.

probably works best with smaller classes (i.e., more student–teacher interaction) and with numerically-trained students, although this approach has yielded positive experience in a variety of settings (H. Bormann, personal communication, 2012).

The overall teaching goals are similar regardless of the order of exercises discussed above; in both cases the students should be able to apply a model to tackle hydrological questions, to evaluate model simulations critically and to develop their own model routines. If only limited time is available for teaching hydrological modeling, e.g., in introduction courses, it might not be possible to achieve all of these goals. In this case, it is more realistic to focus on the first two goals and, thus, to use exercises such as the first ones described below.

5.1 HBVland

In this exercise, the students calibrate the HBV model to a synthetic data set for which it is possible to obtain a perfect fit, i.e., the "observed" runoff is actually a simulated runoff series. Calibration to such synthetic series is in general easier and provides the students with a quick sense of achievement. Twenty years ago one model run still took several minutes, which gave the students time and motivation to think about which parameter to change. With today's computers a model run typically takes less than a second, which makes it possible to test many parameter values. There is a risk that during manual model calibration, students will only do trial and error testing instead of thinking about why they should change

a certain parameter. Therefore, it is important to encourage the students to really discuss and motivate their parameter changes. For the same reasons we found it advisable to tone down the competitive aspect ("who gets the best fit first").

In connection with the calibration of the model to the synthetic data, parameter sensitivity can also be looked at. Here the students use the correct parameter values and then change one or two parameters to different values. The students can simply compare differences or evaluate them more systematically by, for instance, plotting model efficiency values against parameter values.

5.2 Model applications

After having calibrated the simplified case of HBVland, the next steps include exercises where the task is to calibrate the model to a real catchment. To move beyond a pure calibration, it is useful to add some simple model application for which the calibration is needed. Such applications might include using the calibrated model to simulate the runoff caused by a certain precipitation sequence (design flood, exercise 2) or to reproduce the runoff series for a period where there has been a change in the catchment (land-use change effects, exercise 3).

Calibrating to real data, the students discover that the search for some best parameter value often is a compromise. A value that might give a better fit for one period might cause poorer fits for another period. By comparing results of different groups, both in terms of calibrated parameter values and simulation results such as design floods, students also

discover the issue of parameter uncertainty or equifinality. Ending the exercise with letting the students write their parameter values and simulation results in a table on the white board and then discussing the numbers and their spread is usually a valuable way to wrap up such an exercise with some discussion on model uncertainties. An interesting variant is to ask different groups of students to focus on different simulation aspects (high flows versus low flows) and/or to use different objective functions.

In the exercise on land-use change effects, different examples can be used. The one described in the Appendix is a rather small change, where the outcomes in class are usually rather ambiguous and other examples (Seibert and McDonnell, 2010; Seibert et al., 2010) would result in clearer responses of runoff to land-use changes. However, the example here allows the issue of detection limits to be discussed.

5.3 Model uncertainty

For students who are already familiar with the HBV model, an exercise using a Monte Carlo approach (exercise 4) can provide further insights into parameter sensitivity of parameter uncertainty or equifinality. The HBV-light software allows easily performing model runs with randomly generated parameter values (Monte Carlo). This can be used both to look at parameter sensitivity, when allowing only one or two parameters to vary at a time, and parameter uncertainty, when allowing all parameters to vary simultaneously. The HBV-light software furthermore allows various types of automatic model calibration, ranging from a steepest-gradient method (using the GAP tool without the Genetic Algorithm, i.e., only Powell optimization) to a Monte Carlo calibration, to be explored. By these means the students will learn the challenge of finding a suitable compromise between the number of model runs needed to obtain a good model fit and the risk of getting stuck in local optima.

In addition to formal exercises, students at medium to advanced levels can also be given the task to explore the performance of different model variants for a certain catchment. The opportunity to easily test different model variants in HBV-light supports such explorative studies. Data for a catchment can be given to the students with the task of finding the model structure that results in the best model fit. Often this results in several model structures, which allow similarly good calibrations. In a similar way to the evaluation of parameter uncertainty, model structure uncertainty can be studied by comparing simulations of, for instance, a design flood, using different model structures that might have resulted in similar model fits for the calibration period (Uhlenbrook et al., 1999). While we otherwise argue for the value of manual calibration in teaching, for such explorative tasks automatic calibration methods as described above can be suitable. Repeated manual calibrations take a long time and also might result in more subjective, and thus less comparable, results. By exploring different model variants, the students will learn

that there are many choices in model applications, where the right solution is not always obvious. Students will discover that some model variants might be more suitable for a certain catchment, but also that often it is difficult or not possible to decide on one optimal model structure.

5.4 Model routines

Modeling exercises at a more advanced level should also go beyond the use of some existing model software and require the students to develop their own model routines. In the snow-modeling exercise (exercise 5), the aim is to program the snow routine of the HBV model. The goal of this exercise is to make the students better realize how computations in a model are actually carried out. After this exercise they will understand that models are no magic black box but a logical sequence of commands including equations, loops and conditions. They will also be aware that the model code looks more complicated than the equation in the model description, partly because obvious conditions, such as that no more than the available snow can melt, have to be programmed explicitly. There are several ways to implement the snow routine, and it is beneficial to let the students choose their approach freely rather than guiding them too much to one "best" solution. The students (and the teacher) will furthermore experience how easy it is to make mistakes when programming and how time-consuming debugging can be. This should also be considered a valuable experience, although it can be quite frustrating at times.

While the task in the snow exercise is to program a given routine, the exercise on developing an interception routine (exercise 6) goes one step further. Here the students have to start with thinking about how an interception routine could look. Depending on their background some guidance might be needed for this. The students will learn that there are different ways to formulate a hydrological process in conceptual terms, leading to different model implementations.

6 Concluding remarks

Teaching hydrological modeling is challenging, and it is important to be able to focus more fundamentally on modeling and model development instead of technical issues on how to use certain software. HBV-light is suitable in this respect, as students are able to run the model on their own after a very short introduction (~ 10 min).

This also allows a first introduction to modeling to be given in just a few teaching hours. After two or three hours of lectures and about four hours of exercises (exercises 1 and 2 or 3), the students usually have a good understanding of the HBV model and conceptual modeling in general. Exercises 4–6 can be used in advanced classes to deepen the understanding. Additional explorative exercises ranging from

small course work projects to theses can provide further insights into the modeling process.

User-friendly software like HBV-light can limit problems related to the usage of a model and, thus, help to focus on the hydrological aspects of modeling. However, the experience of a certain level of frustration is hard to avoid completely and can actually be helpful. This can include issues related to manual calibration (e.g., getting stuck in a local optimum) or debugging of one's own model code.

To summarize, the aim of using HBV-light in education as proposed in this paper is to promote a humble attitude of the students to hydrological modeling, realizing both opportunities and limitations of hydrological modeling. We believe this is an important attitude for future hydrologists dealing with hydrologic questions in a changing, and uncertain, world.

Appendix A

Examples of exercises

In this appendix a collection of six concrete exercises is provided to illustrate the use of HBV in teaching. For testing, the example data can be downloaded from the HBV-light website; for teaching, we recommend adapting the exercises using data from catchments that are related in some way to the experience of the students.

A1   Exercise 1 (HBVland)

Calibrate the HBV model for the HBVland catchment for the period 1 September 1981 to 31 August 1991 ("warm-up" period starting at 1 January 1981). This catchment behaves exactly as the HBV model sees the world, therefore you might be able to achieve a perfect fit ($R_{eff} = 1$).

1. Try to calibrate the model. It is a good idea to start with the snow routine to get the spring flood right, then work on the soil-routine parameters to get the water balance right and finally fix the response function. You will have to do this in iterations.

2. During calibration also look at different variables such as soil moisture or storage in the upper groundwater box.

3. Once you have reached a perfect fit (or have received the "true" parameter values by kindly asking your teacher), you may again change parameter values and study the effects of different parameter values.

4. Change one (or two) of the following parameters: TT, CFMAX, FC, BETA, LP, K_0, K_1, K_2, PERC, UZL, MAXBAS, SCF.

5. Discuss – before running the model – what effect you expect (i.e., more runoff during spring, slower response to rain, ...).

6. Run the model and look at the deviation of the simulated runoff (red line) from the "recorded" runoff (blue line).

7. Make a note of each change of a parameter value and its effect on the simulation.

8. Change the parameter value back to its original value.

9. Continue with 3.

A2   Exercise 2 (estimation of design flood)

A synthetic sequence of extreme precipitation has been derived by meteorologists (Table A1). Now it is your task to estimate the flood that this sequence would cause for the River Fyris at Vattholma (Uppland), Sweden. In other words, you should estimate a design flood. You have decided to use the HBV model to solve this problem. Some friendly hydrologist put all necessary files together (most importantly the "ptq.txt" file with areal precipitation, temperature and observed runoff for an eleven-year period), but the model is far from well-calibrated.

You have to complete three steps:

1. Calibration: Change the following parameters in order to get as good a fit as possible between observed (blue) and simulated (red) runoff: TT, CFMAX, SCF, FC, BETA, LP, K1, K2, PERC, MAXBAS (K0 and UZL should not be used (i.e., put them to zero), do not change the values for CFR, CWH and CET (0.05, 0.1, 0.1)). Use the period 1 September 1981 to 31 August 1987 for calibration (with the "warm-up" period starting at 1 January 1981).

2. Validation: Before you use your calibrated model for any prediction, it is important that you test your parameter set for an independent time period. Use the period 1 September 1987 to 31 December 1991 for this test. Is the fit worse? Can you give an explanation? How will your design flood be affected?

3. Simulation of flood:

   3.1 Make a backup copy of ptq.txt.

   3.2 Open the file ptq.txt in a text editor (or Excel).

   3.3 Choose a period for which you replace the observed precipitation by the synthetic sequence (Table A1).

   3.4 Save the file as "ptq.txt" (if you use Excel choose the format "*.txt" (tab-separated)).

   3.5 Reopen the catchment in order to load the new ptq.txt file and run the model. Check the peak value of your simulated flood.

   3.6 Return to the backup file, choose a different period and continue with 3.2. Do this 5–10 times.

www.hydrol-earth-syst-sci.net/16/3315/2012/

Hydrol. Earth Syst. Sci., 16, 3315–3325, 2012

Table A1. Synthetic sequence of extreme precipitation.

Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14
P (mm)	5	5	5	5	5	10	10	40	120	30	10	10	5	5

4. Discuss the following questions: What influences the size of the simulated flood? Under which seasons/conditions does the simulated flood become largest/smallest?

A3   Exercise 3 (effects of land-use change)

(1) Calibrate the HBV-model to the Norrsjön catchment, Sweden

Change the following parameters in order to simulate the runoff as well as possible: TT, CFMAX, SCF, FC, BETA, LP, K0, K1, K2, PERC, UZL, MAXBAS. Make a note of each change of a parameter value and its effect on the simulation. Three elevation zones (Table A2) but only one vegetation zone are used (lake percentage: 2 %). Use the period 1 September 1973 to 31 August 1982 for calibration ("warm-up" from 1 January 1972).

Write down your final parameter values and the model efficiency (R_{eff}) you achieved.

(2) Land-use change

Between 31 August 1982 and 31 August 1985, the area with clearcutting increased from around 10 % to above 20 % of the total catchment area. Run the model for the period 1 September 1981 to 31 August 1991 (with your parameters from the calibration) and look at the differences between simulated runoff (supposed to be the "real" runoff without the increase of clearcutting) and observed runoff. Have a look at the accumulated difference. Use "Save results" (Select < Settings > < Model Settings > from the main menu) and load the result file into Excel or MATLAB for the further analysis. Calculate and discuss the effects of the land-use (31 August 1985 to 31 August 1991) change on:

– water balance;

– runoff during different seasons;

– runoff during high flow conditions;

– runoff during low flow conditions;

– spring flood (volume, peak);

– peak runoff during autumn; and

– flow duration curve (= frequency distribution) (total period and different seasons).

Some useful MATLAB commands:

Table A2. Hypsometric information for the Norrsjön catchment.

Min. [m a.s.l.]	Max. [m a.s.l.]	Fraction
350	400	0.14
400	450	0.56
450	520	0.30

Elevation of the climate station: 250 m a.s.l.
(PCALT = 10 %/100 m, TCALT = 0.6 °C/100 m).

– cumsum, max, min

– find, e.g., $i = $ find (month > 2 & month < 5), maxspring = max (runoff (i))

– sort

– hist

A4   Exercise 4 (Monte Carlo)

HBV-light allows many model runs to be carried out easily with randomly generated parameter sets by using the tool "Monte Carlo Runs". In this exercise you are asked to perform both sensitivity studies (allowing one or two parameter values to vary) and parameter uncertainty estimations (allowing all (many) parameters to vary). Use the catchments HBVland or Vattholma (exercises 1 or 2) for this exercise.

Parameter sensitivity, one parameter:

1. In the Monte Carlo tool, set the minimum and maximum for all parameters to the optimal values from the previous exercises and save the parameters ("save settings").

2. Change the limits for one parameter (e.g., CFMAX, FC, K1...) based on the values in Table A3 and let the software do many (∼ 100–1000) model runs.

3. Open the file results\multi.txt in MATLAB or Excel and plot the model efficiency (R_{eff}) against the parameter value that you allowed to vary.

4. Go to 2 and repeat the same for other parameters. Discuss the sensitivity of the different parameters. It might also be interesting to look at the sensitivity with regard to the log-transformed efficiency ($\log R_{\mathrm{eff}}$) and volume error (meandiff).

Parameter sensitivity, two parameters: Perform the same steps as above, but allow two parameters to vary simultaneously now (e.g., TT and CFMAX, BETA and LP, K2 and

Table A3. Model parameters and their ranges to be used in the exercise.

Parameter	Explanation	Minimum	Maximum	Unit
Snow routine				
TT	Threshold temperature	−1.5	2.5	°C
CFMAX	Degree-day factor	1	10	$\mathrm{mm\,°C^{-1}\,d^{-1}}$
SCF	Snowfall correction factor	0.4	1	–
CWH	Water holding capacity	0	0.2	–
CFR	Refreezing coefficient	0	0.1	–
Soil routine				
FC	Maximum of SM (storage in soil box)	50	500	mm
LP	Threshold for reduction of evaporation (SM/FC)	0.3	1	–
BETA	Shape coefficient	1	6	–
CET	Correction factor for potential evaporation	0	0.3	$°C^{-1}$
Response routine				
K1	Recession coefficient (upper box)	0.01	0.4	d^{-1}
K2	Recession coefficient (lower box)	0.001	0.15	d^{-1}
PERC	Maximal flow from upper to lower box	0	3	$\mathrm{mm\,d^{-1}}$
MAXBAS	Routing, length of weighting function	1	7	d

PERC, ...). For visualization you now need to use 3-D or contour line plots.

Monte Carlo runs:

1. In the Monte Carlo tool, set the minimum and maximum for all parameters according to the feasible limits given in Table A3 (you might want to save these values with "save settings" to avoid putting the numbers in more than once).

2. Choose "save only if $R_{\mathrm{eff}} > 0.6$" (to avoid large files) and let the software do a large number of runs (depending on available time, 10 000–1 000 000, you may go for a coffee or lunch in the meantime).

3. Produce so-called "dotty-plots" by plotting individual parameter values against model efficiency (R_{eff}).

4. Discuss which parameters are less/more constrained. Compare these results with your conclusions from the sensitivity analysis.

A5 Exercise 5 (snow model)

Data from the Kassjöån basin in Medelpad, Sweden, are used in this exercise. In the file ex5_snow7376.dat you will find precipitation (mm), temperature (°C) and depth of the snowpack (mm water equivalent) (measured using a snow pillow). In each line of the file, there are data from one day (six columns with year, month, day, precipitation, temperature, snow).

Simulation of snow accumulation and snowmelt

1. Write a MATLAB program to simulate the accumulation and melting of snow according to the degree-day method (see below). Include storage within the snowpack and refreezing into your snow routine (the snowpack can store water up to 10 % of its water equivalent and the refreezing rate for this water is 20 times lower than the melting rate).

2. Plot both snowpack (simulated and measured) and the amount of water flowing into the soil against time (daily values). Change the parameter values (degree-day factor, threshold temperature) to fit the simulated snowpack to the observed one.

3. Discuss the results and how they are influenced by the parameter values. For instance, you plot the maximal snow water equivalent in the different years as function of TT, SCF and/or CFMAX.

Program code as starting point:

```
snow=load("z: \...\ex5_snow7376.dat");

P=snow(:,4);

T=snow(:,5);

S=snow(:,6);


sno(1)=0;

avr(1)=0
```

```
for t=2:length(P)

   % sno(t)=sno(t-1) + P(t);

end

dag=1:length(P);

plot(dag,sno,"r",dag,S,"b",dag,avr,"g")
```

Some MATLAB functions that you may find useful:

– load

– plot

– axis

– title

– xlabel ylabel

– if ... (else ...) end

– for ... end

– min

– max

A6 Exercise 6 (interception routine)

In the HBV model (as used in our class), there is no interception routine.

1. Why does the model work for forested catchments anyway?

2. Suggest an interception routine similar to the different routines in the HBV model. You may use one to three parameters.

3. Implement this interception routine in MATLAB and test it using the data provided in the ptq.txt file for the Vattholma catchment (see exercise 2).

4. Generate a new ptq.txt file using the simulated throughfall as input (i.e., precipitation) and run the HBV model. How do simulations and calibrated parameter values differ when you use simulated throughfall instead of observed precipitation as input?

References

AghaKouchak, A. and Habib, E.: Application of a Conceptual Hydrologic Model in Teaching Hydrologic Processes, Int. J. Eng. Educ., 26, 963–973, 2010.

AghaKouchak, A., Nakhjiri, N., and Habib, E.: An educational model for ensemble streamflow simulation and uncertainty analysis, Hydrol. Earth Syst. Sci. Discuss., 9, 7297–7315, doi:10.5194/hessd-9-7297-2012, 2012.

Bergström, S.: Development and application of a conceptual runoff model for Scandinavian catchments, Bulletin Series A, No. 52, Department of Water Resources Engineering, Lund Institute of Technology, University of Lund, 134 pp., 1976.

Bergström, S.: The HBV Model: Its Structure and Applications, Swedish Meteorological and Hydrological Institute (SMHI), Hydrology, Norrköping, 35 pp., 1992.

Bergström, S.: The HBV model (Chapter 13), in: Computer Models of Watershed Hydrology, edited by: Singh, V. P., Water Resources Publications, Highlands Ranch, Colorado, USA, 443–476, 1995.

Braun, L. N. and Renner, C. B.: Application of a Conceptual Runoff Model in Different Physiographic Regions of Switzerland, Hydrolog. Sci. J., 37, 217–231, 1992.

Doherty, J. and Johnston, J. M.: Methodologies for calibration and predictive analysis of a watershed model, J. Am. Water Resour. As., 39, 251–265, 2003.

Doherty, J. and Skahill, B. E.: An advanced regularization methodology for use in watershed model calibration, J. Hydrol., 327, 564–577, doi:10.1016/j.jhyrol.2005.11.058, 2006.Please check doi.

Hottelet, C., Braun, L. N., Leibundgut, C., and Rieg, A.: Simulation of snowpack and discharge in an alpine karst basin, IAHS-AISH P., 218, 249–260, 1993.

Konz, M. and Seibert, J.: On the value of glacier mass balances for hydrological model calibration, J. Hydrol., 385, 238–246, doi:10.1016/j.jhydrol.2010.02.025, 2010.

Lindström, G. and Bergström, S.: Improving the HBV and PULSE-models by use of temperature anomalies, Vannet i Norden, 16–23, 1992.

Lindström, G., Johansson, B., Persson, M., Gardelin, M., and Bergström, S.: Development and test of the distributed HBV-96 hydrological model, J. Hydrol., 201, 272–288, 1997.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P.: Numerical recipes in C++ : the art of scientific computing, 2nd Edn., Cambridge University Press, Cambridge, UK; New York, xxvii, 1002 pp., 2002.

Seibert, J.: Estimation of parameter uncertainty in the HBV model, Nord. Hydrol., 28, 247–262, 1997.

Seibert, J.: Regionalisation of parameters for a conceptual rainfall-runoff model, Agr. Forest Meteorol., 98, 279–293, 1999.

Seibert, J.: Multi-criteria calibration of a conceptual runoff model using a genetic algorithm, Hydrol. Earth Syst. Sci., 4, 215–224, doi:10.5194/hess-4-215-2000, 2000.

Seibert, J. and Beven, K. J.: Gauging the ungauged basin: how many discharge measurements are needed?, Hydrol. Earth Syst. Sci., 13, 883–892, doi:10.5194/hess-13-883-2009, 2009.

Seibert, J. and McDonnell, J. J.: Land-cover impacts on streamflow: a change-detection modelling approach that incorporates parameter uncertainty, Hydrolog. Sci. J., 55, 316–332, doi:10.1080/02626661003683264, 2010.

Seibert, J., McDonnell, J. J., and Woodsmith, R. D.: Effects of wild-fire on catchment runoff response: a modelling approach to detect changes in snow-dominated forested catchments, Hydrol. Res., 41, 378–390, doi:10.2166/Nh.2010.036, 2010.

Steele-Dunne, S., Lynch, P., McGrath, R., Semmler, T., Wang, S. Y., Hanafin, J., and Nolan, P.: The impacts of climate change on hydrology in Ireland, J. Hydrol., 356, 28–45, doi:10.1016/j.jhydrol.2008.03.025, 2008.

Uhlenbrook, S., Seibert, J., Leibundgut, C., and Rodhe, A.: Prediction uncertainty of conceptual rainfall-runoff models caused by problems in identifying model parameters and structure, Hydrolog. Sci. J., 44, 779–797, 1999.

Wagener, T. and McIntyre, N.: Tools for teaching hydrological and environmental modeling, Computers in Education Journal, 17, 16–26, 2007.