Technical Report No. 17

Documentation for the Hydrological Discharge Model

STEFAN HAGEMANN[*] AND LYDIA DÜMENIL

Max Planck Institute for Meteorology
Bundesstraße 55
D-20146 Hamburg, Germany

[*]e-mail: Hagemann@dkrz.de
Tel.: +49 40 41173-101
Fax: +49 40 41173-366

September 1998

Contents

1. Introduction

To improve the representation of hydrological land surface processes, which has so far been treated inadequately in global models of the atmospheric general circulation (GCMs), a model for the lateral waterflows from the continents into the ocean on the global scale was developed. The model describes the translation and retention of the lateral discharge as a function of the spatially distributed land surface characteristics that are globally available. Here, global scale refers to the resolution of 0.5˚ and lower, corresponding to a typical GCM gridbox area of about 2500 km^2. This model is called the **Hydrological Discharge model** or **HD model**.

The HD model computes the discharge only at 0.5˚ resolution. All model input fields (runoff and drainage, see Sect. 3.1.) from the various GCM resolutions are interpolated to the same 0.5˚ grid. Thus, input fields may be used from any available resolution, if the corresponding interpolation routine to the 0.5 degree grid is provided. Since the HD model uses a time step of one day, a temporal resolution of one day is sufficient for the input fields.

The HD model adequately simulates the mean annual cycle of monthly mean discharges for large catchments of the earth (cf. *Hagemann*, 1998). But in principle the HD model is also able to simulate the discharge for some smaller catchments if the datasets used are accurate enough for the corresponding region, as shown for the Odra flooding in *Hagemann* (1998).

Some basic information about hydrological modelling providing the background to the model equations of the HD model are given in Sect. 2. The model structure and the model parameterizations as functions of land surface characteristics are presented in Sect. 3. Technical features of the model source code and an example how to run the model on the DKRZ CRAY J90 computer system are given in Sect. 4. In Sect. 5. several programs are described that help to evaluate the output from the HD model.

2. Hydrological modelling background

In general, hydrological discharge models mathematically relate the discharge of a given catchment to a specific input. Different variables may be used as model input such as precipitation, surface runoff, soil water content or the inflow from other catchments. In order to simulate the discharge, system-theoretical approaches are often applied which, for example, are also common in the scientific field of electronics. Here, the catchment is represented by a system function h(t) which transforms an input I(t) into an output O(t).

If the system function is used to represent the discharge behaviour of a catchment, the catchment characteristics, such as area, topography, geology, etc., will directly affect the shape and the parameter of the system function. For a specific catchment, the output O(t) is computed by a convolution of the input I(t) with the system function h(t) of the catchment:

$$O(t) = \int_0^\infty I(\tau) \cdot h(t - \tau) \cdot d\tau \tag{1}$$

For the discrete case and under the causality assumption that no output may occur before the corresponding input has happened, the output of the time step j may be computed as:

$$O_j = \sum_{i=0}^{j} I_i \cdot h_{j-i} \tag{2}$$

The system function is also designated as impulse response function or unit hydrograph because it is equivalent to the output or response of a system to the input of a delta impulse with unity length. In order to derive the system function for a specific catchment, measured time series of input and output must exist. The derivation of the system function from these time series may be difficult and time consuming. Therefore an appropriate model approach is usually applied to represent the catchment for which the system function is known. Some of these approaches are described by *Dooge* (1977) and *Singh* (1988).

The lag time τ is an important value which characterizes a hydrologic system or a model approach, respectively. τ is defined as the delay time between the centres of the input and the output of the system. If τ is known the mean flow velocity in the can be computed according to equation 3, where Δx is a characteristic extension length of the system.

$$v = \frac{\Delta x}{\tau} \tag{3}$$

The linear reservoir is the most used model approach for the representation of flow processes in hydrology. The outflow Q from the reservoir is proportional to its content S. Q can be

expressed as the content S divided by the retention time k of the reservoir (equation 4, t = time). This retention time k is also called retention coefficient and it is defined as the average residence time of water within the reservoir (Singh, 1988). The lag time of the reservoir is equal to k.

$$Q(t) = \frac{S(t)}{k} \tag{4}$$

A general reservoir must fulfil the continuity equation 5 which relates Q to the inflow I into the reservoir.

$$\frac{dS(t)}{dt} = I(t) - Q(t) \tag{5}$$

The equations 4 and 5 lead to a linear differential equation for the discharge Q with one parameter k:

$$k \cdot \frac{dQ(t)}{dt} = I(t) - Q(t) \tag{6}$$

The system function $h(t)$ of the linear reservoir can be expressed as:

$$h(t) = \frac{1}{k} \cdot e^{-\frac{t}{k}} \tag{7}$$

Therefore equation 8 is yielded for the discharge as a function of time if $Q(t=0) = 1/k$ and $I(t>0) = 0$:

$$Q = \frac{1}{k} \cdot e^{-\frac{t}{k}} \tag{8}$$

The cascade of n equal linear reservoirs describes the serial arrangement of n equal linear reservoirs which have the same retention coefficient k. Here, the outflow from the reservoir i equals the inflow into the reservoir $i+1$ and equation 6 is applied to each reservoir. This yields a system of n equations which are connected by the in- and outflows of the linear reservoirs.

$$k \cdot \frac{dQ_i(t)}{dt} = I_i(t) - Q_i(t) \qquad\qquad i = 1, ..., n \tag{9}$$

$$Q_i = I_{i+1} \quad , \quad I_1 = I(t) \quad , \quad Q_n = Q(t)$$

5

For the system function of the reservoir cascade, equation 10 is valid:

$$h(t) = \frac{t^{n-1}}{k^n \cdot (n-1)!} \cdot e^{-\frac{t}{k}} \tag{10}$$

The cascade of linear reservoirs has the two parameters n and k. Sometimes the number of reservoirs n is considered as real value, so that in equation 10 the term $(n\text{-}1)!$ has to be replaced by the gamma function $\Gamma(n)$. The lag time of this system is:

$$\tau = n \cdot k \tag{11}$$

3. The Hydrological Discharge model

Sect. 3.1. describes the model structure of the Hydrological Discharge model (HD model). *Hagemann and Dümenil* (1998) have described the basic principles of the HD model and a first parameterization approach using gridbox characteristics. Based on investigations in a large Swedish catchment as well as on global sensitivity tests (cf. *Hagemann*, 1998), the final HD model parameterizations were derived. These parameterizations and its model equations are presented in Sect. 3.2.

For certain gridboxes, the influence of wetlands and lakes on the discharge is important. Parameterizations representing this influence are described in Sect. 3.3. and 3.4. The location and size of wetlands and lakes is strongly affected by the general climate conditions. Their inclusion in the HD model parameterizations causes an implicit dependence on the climate. For discharge simulations in the current climate, wetlands and lakes datasets that describe their actual global distribution are sufficient. For different climate conditions (paleo or climate change scenarios), such datasets have to be developed. One possibility to do this is offered by a model of Coe and Bonan (1997), which may be used to compute surface water distributions dependent on the given climate.

3.1. The HD model

Fig. 1 shows the model structure of the HD model as developed by *Hagemann and Dümenil* (1998). It separates the lateral waterflow into the three flow processes of overland flow, baseflow and riverflow. Overland flow and baseflow are both represented by a single linear reservoir, and riverflow is represented by a cascade of n equal linear reservoirs. Overland flow uses runoff as input, baseflow is fed by drainage and the inflow from other gridboxes contributes to riverflow. The sum of the three flow processes equals the outflow from a gridbox.

For overland flow, the retention coefficient k_o of the linear reservoir is a function of the average slope within a gridbox and the gridbox length Δx. The gridbox length is defined as the distance between the centres of two adjacent gridboxes in the direction of the flow. For riverflow, the retention coefficient k_r of the linear reservoirs is a function of the topography gradient Δh between two adjacent gridboxes in the direction of the flow and the gridbox length Δx. The parameters of overland flow and riverflow are influenced by the wetland fraction (see Sect. 3.3.) and the lake area (see Sect. 3.4.) within a particular gridbox. The retention time k_g of the baseflow reservoir is a function of the gridbox length. Its value for a typical 0.5° gridbox ($\Delta x =$ 50 km) was set to 300 days. Here, the exact typical value of k_g is not very important because the HD model is not very sensitive to small changes in k_g.

As a general strategy the HD model computes the discharge only at 0.5° resolution. All model input fields (such as runoff and drainage) from the various GCM resolutions are therefore

interpolated to the same 0.5° grid. The origin of the model grid is located at the North Pole and the date line, while the centre of the corresponding gridbox has the coordinates -179.75° W and 89.75° N. The HD model uses a daily time step. For riverflow the time step is six hours to pay regard to the minimum travel time through a 0.5° gridbox which is limited by the time step chosen.
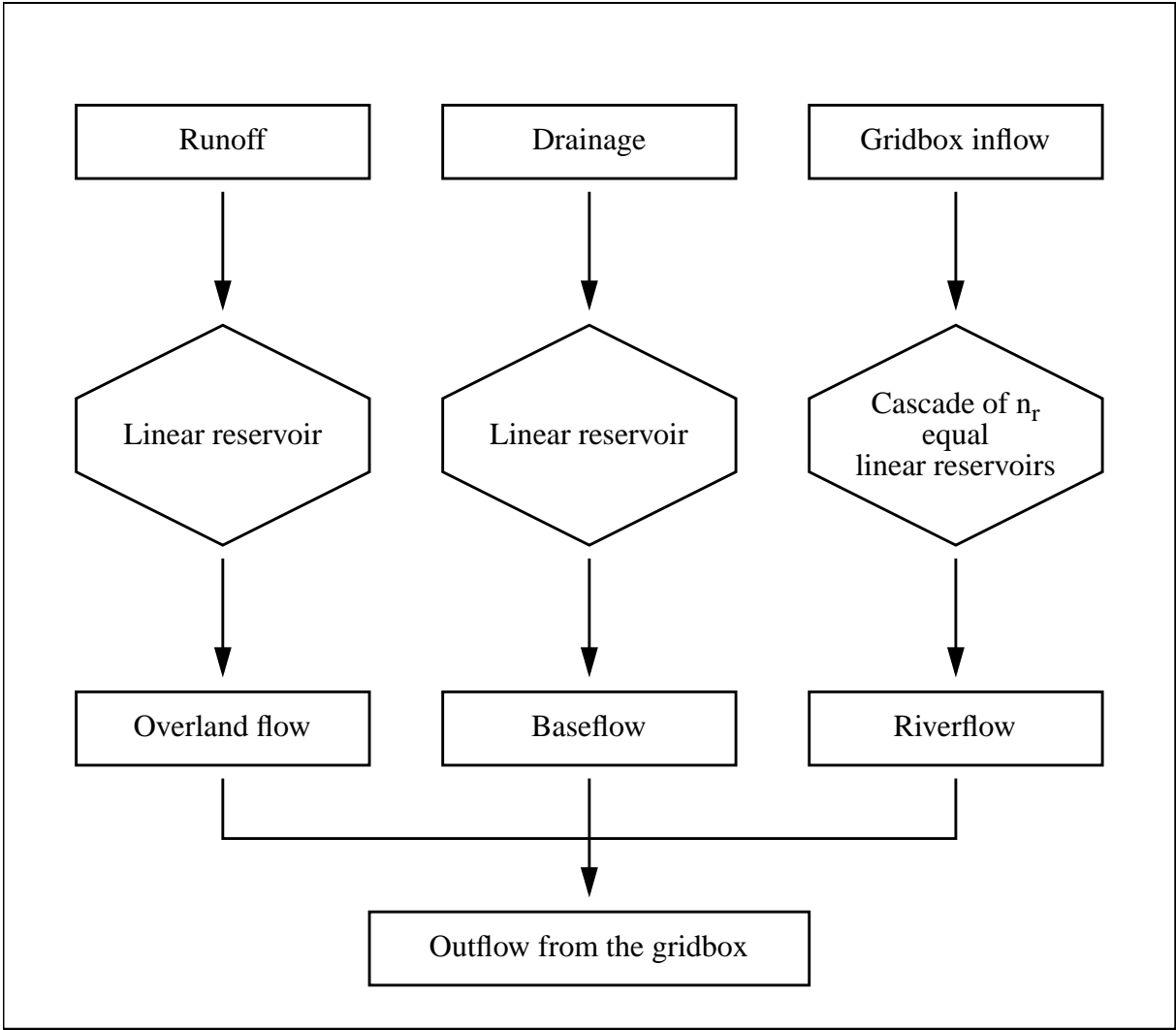
Fig. 1. Structure of the HD model

For each gridbox of the HD model eight possible outflow directions (the four main directions North (N), East (E), South (S) and West (W) and the four diagonal directions NE, SE, SW and NW) are defined. But for a specific gridbox only one outflow direction is permitted. The outflow from this gridbox enters the neighbouring gridbox which has the lowest topography value of the surrounding gridboxes. On the basis of these definitions a global dataset of outflow directions is derived. This global dataset will be referred to as the *river direction file* (RDF). Its structure is shown in Fig. 2.

The skill of the discharge simulation depends not only on the formulation of the model

equations, but also on the precise definition of the boundaries of the model catchments. The sizes and the positioning of the model catchments on the globe were defined by using a modified topography dataset. Since the definition of flow directions derived from the available 0.5˚ topography dataset[1] is not detailed enough for an accurate global discharge simulation, a model topography had to be created. Several mathematical methods (cf. *Hagemann and Dümenil*, 1998) were developed to include important aspects of the real topography such as river flow paths and catchment borders which cause a good agreement between the model catchments and the real catchments. This agreement is generally not perfect because the methods used are automated to a certain degree. Pure manual methods would give a perfect agreement globally but they are too expensive and will not considerably improve the simulation of discharge.
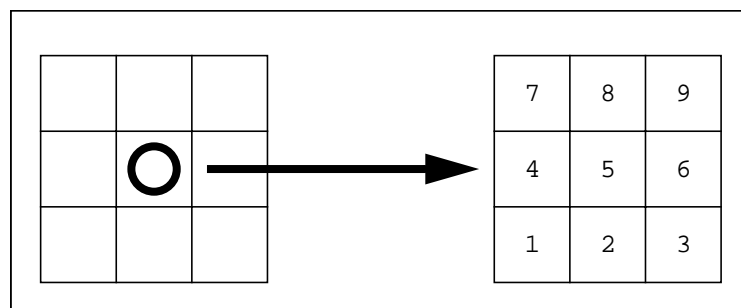

Fig. 2. Structure of river direction file (RDF)
The RDF value of a specific grid box specifies the target grid box for its outflow. 0 specifies a sea point.

The HD model was developed for the coupled application with the climate models used at MPI. In this study it was applied in conjunction with the GCM ECHAM4 (*Roeckner et al.*, 1996). In ECHAM4, the soil is represented by a single soil layer. Time series of runoff and drainage are calculated according to the scheme of *Dümenil and Todini* (1992). Here, runoff is computed as infiltration excess from a bucket which takes the subgrid variability of the soil saturation within a GCM gridbox into account. This is done by defining a statistical distribution of soil water capacities in the gridbox. Drainage is the amount of water that is allowed to percolate downwards from the bucket. A specific parameterization of the influence of wetlands on evaporation is presently not included in the atmospheric model ECHAM4.

3.2. Final parameterizations of the HD model

Detailed investigations in a large Swedish catchment (Torneälven/Kalixälven catchment)

---

1. The available global 0.5˚ topography dataset was derived from the global five minute topography dataset of the National Geographic Data Centre (*Edwards*, 1989) by area weighted averaging.

which comprises of 51 0.5°-gridboxes were used to improve the model formulations and the model constants (cf. *Hagemann*, 1998). Hence, overland flow is a function of the average slope φ within a gridbox instead of the topography gradient $\Delta h/\Delta x$ used in the first parameterization approach (see *Hagemann and Dümenil*, 1998) where overland flow was represented by a cascade of linear reservoirs. The investigations mentioned above as well as global applications of the HD model yielded that for overland flow, the representation of a single linear reservoir is sufficient. Within a 0.5°-gridbox, it is important for the quality of the simulated discharge that overland flow is represented by a two-parameter model (*Hagemann and Dümenil*, 1998). But sensitivity tests have shown that for catchments which comprise of many 0.5°-gridboxes, the simulated discharge shows no differences between simulations where overland flow is represented by a one- and a two-parameter model using the same lag time in both cases (In the source code, overland flow is still coded as a reservoir cascade to account for possible future developments on different spatial scales, so that overland flow is represented by a reservoir cascade with only one reservoir).

Thus, the following equations are formulated for the parameter of overland flow and riverflow:

$$k_o = 17.87 \cdot 10^{-2} \cdot \frac{\Delta x}{\varphi^{0.1}} \qquad \text{und} \qquad n_o = 1 \qquad (12)$$

$$k_r = 9.92 \cdot 10^{-4} \cdot \frac{\Delta x}{\left(\dfrac{\Delta h}{\Delta x}\right)^{0.1}} \qquad \text{und} \qquad n_r = 5 \qquad (13)$$

Here, the gridbox length Δx has to be expressed in km and the retention coefficients k_o and k_r are given in days. In cases, where the average slope φ within a gridbox is Zero, φ (in equation 12) is replaced by the topography gradient $\Delta h/\Delta x$ to the next gridbox in flow direction. For certain gridboxes, the parameter of overland flow and riverflow are influenced by the fractional amount of wetlands and lakes within a gridbox. This will be described more in detail in Sect. 3.3. and 3.4. The corresponding flow velocities $v_{0, i}$ of overland flow and riverflow are given by equation 14, with $i \in \{o, r\}$:

$$v_{0, i} = \frac{\Delta x}{n_i \cdot k_i} \qquad (14)$$

For baseflow, the gridbox length Δx was incorporated in its parameterization to make the formulation independent of the scale used. The retention time k_g of the linear baseflow reservoir is now defined as:

$$k_g = t_g \cdot \frac{\Delta x}{d_0} \qquad (15)$$

Here, t_g is the typical retention time of baseflow (300 days) which is used for k_g globally in the first parameterization approach, and d_0 is the typical diameter of a 0.5°-gridbox (50 km). In a global application of the HD model to an ECHAM4 simulation at T42 resolution, the effects of using this formulation instead of $k_g = t_g$ were comparatively small. But it was decided to use equation 15 since the baseflow parameterization becomes independent of resolution by the explicit incorporation of the gridbox length in its formulation.

A trebling of t_g yielded no significant changes in the total discharge. On the contrary a reduction of t_g to a third or a tenth has an impact to the total discharge but this generally causes a worsening of the simulated discharge. Due to the fact that the total discharge is relatively insensitive against the baseflow, it seems that an extended parameterization using other gridbox characteristics will not cause significant improvements of the simulated discharge. (The influence of wetlands and lakes on the discharge was omitted in these investigations, since Sect. 3.3. and 3.4 deal with these issues more in detail.)

The initialization of the linear model reservoirs is a problem because measurements of the reservoir contents do not exists. Since arbitrarily initialized reservoirs differ from reality, a phase of spin-up is required to allow an assimilation of the model reservoirs to reality. An unrealistic initialization may cause deficiencies of the simulated discharges. Their effects in time have about the same values as the lag times of the considered flow process (cf. *Hagemann and Dümenil*, 1998). For multi-annual discharge simulations, these effects are generally negligible for overland flow and riverflow (except for gridboxes with a high coverage of lakes or wetlands). An unrealistic initialization of the baseflow reservoir may cause biases in the simulated discharge for longer periods of time which may raise or lower the average discharge volume by a certain amount. In order to minimize these biases, the following method turned out to be useful to create appropriate initialization datasets.

With given input time series of runoff and drainage, the discharge is simulated for the first year. The contents of the model reservoirs at the end of this year are then used as initialization for the renewed discharge simulation of the same year. This procedure is repeated as often as the differences of the reservoir contents before and after the year of simulation are still large (differences of fewer than 5% for most of the large rivers seem to be appropriate). The reservoir dataset which is generated at last with this procedure will then be used as initialization for the discharge simulation over the whole planned time period.

3.3. Wetlands parameterization of the HD model

Sect. 3.3.1. provides some background about wetlands and the available datasets on the global scale. The wetlands parameterization used in the HD model is presented in Sect. 3.3.2..

3.3.1. Wetlands - definition and global datasets

Wetlands, by definition, are characterized by water saturation in the root zone, at, or above the soil surface, for a certain amount of time during the year. This fluctuation of the water table above the soil surface is unique to each wetland type (e.g. marsh or swamp). Wetlands play a critical role in regulating the movement of water within watersheds as well as in the global water cycle (*Mitsch and Gosselink*, 1993). They may have different functions as a result of their position in the landscape and their dominant water source. Wetlands store precipitation and surface water and then slowly release the water into associated surface water resources, ground water reservoirs, and the atmosphere through evaporation. With regard to the lateral waterflows they serve as an intercepting reservoir, and hence change sharp waterflow peaks to slower discharges over long periods of time. Therefore the inclusion of a wetlands parameterization into a global discharge model is important for an adequate simulation of the discharge in regions where wetlands are a major feature of the land surface (or the gridbox area). They should be also included into the atmospheric models to get a more realistic amount of evaporation in the wetlands regions. Another important aspect of wetlands is their role in the global methane production (*Walter et al.*, 1996).

The inclusion of wetlands into a global discharge model or an atmospheric model requires the use of a global wetlands dataset. At present there are two global wetlands datasets that define the fractional area of wetlands in a gridbox at 1x1 degree resolution. The two datasets differ for certain regions of the world. Results of *Hagemann and Dümenil* (1998b) indicate that the dataset of *Matthews and Fung* (1987) is more suitable for the use in global climate or hydrological modelling than the dataset of *Cogley* (1987, 1991, 1994), so that this dataset will be used until revised datasets become available. The global distribution of wetlands according to this dataset is shown in Fig. 3.

Since an appropriate parameterization which represents the influence of wetlands on the lateral discharge does not exists, a conceptual model was developed which is described in Sect. 3.3.2.

3.3.2. Wetlands parameterization

The reciprocal of the retention time of a linear reservoir is the renewal rate of water which is a general useful concept of wetlands hydrology. The renewal rate is defined as the ratio of throughput to average volume within the system (*Mitsch and Gosselink*, 1993). However, few measurements of renewal rates have been made in wetlands, although it is a frequently used parameter in limnological studies, so that this information is not available for the whole globe. On the other hand, recent evidence suggests that the theoretical retention time of the linear reservoir is often much longer than the actual residence time of water in the wetlands (*Mitsch and Gosselink*, 1993).
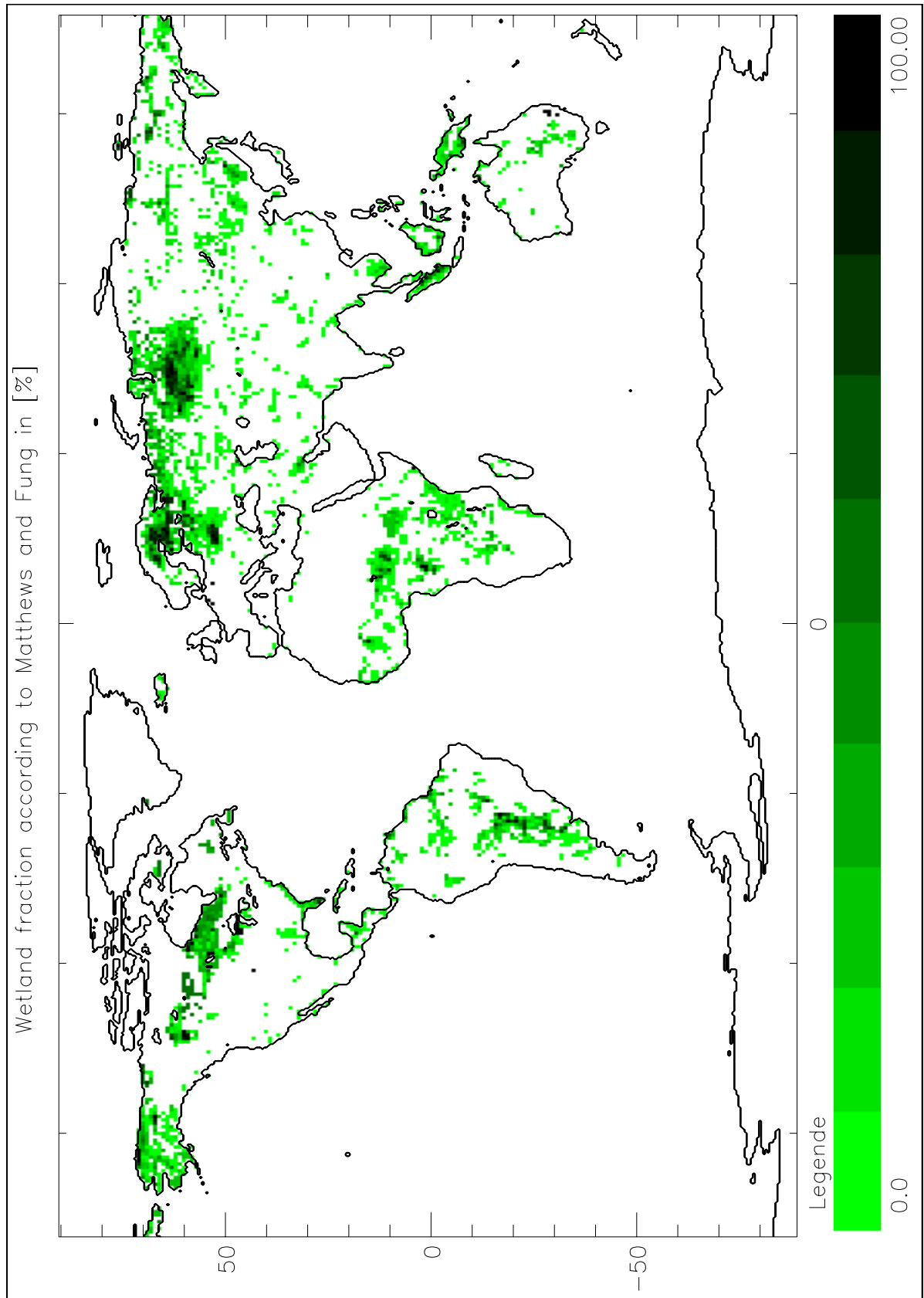
Fig. 3. Wetland fraction according to *Matthews and Fung* (1987).
Scaling in 10% steps

Therefore a conceptual model was developed which represents the influence of wetlands on the lateral flow processes. It is based on principal ideas about how wetlands may affect the specific flow processes. Their influence should not be too large, if only a small fraction of the gridbox is covered by wetlands. On the other hand their influence should be comparatively large if the major part of a gridbox is covered by wetlands. A schematic curve of this behaviour is shown in Fig. 4.
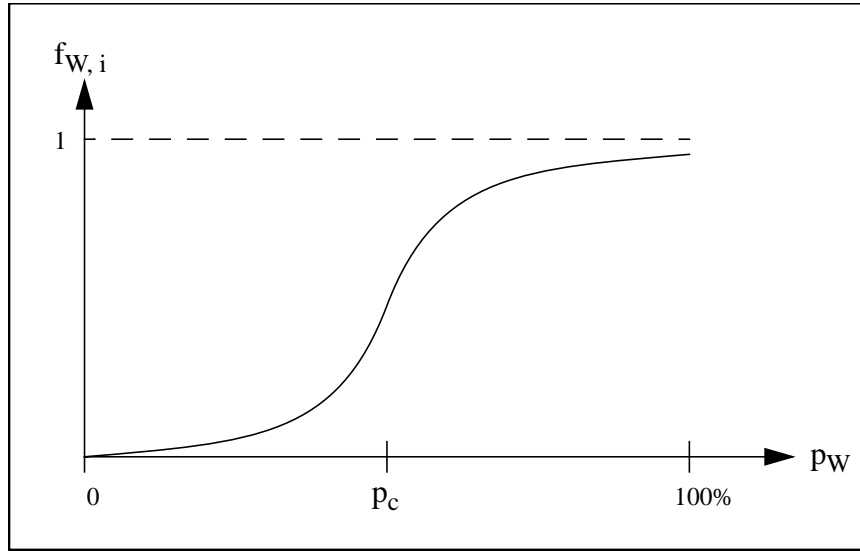
Fig. 4. Schematic curve of the discharge delay factor $f_{W,i}$ as function of the gridbox coverage p_W by wetlands.

If we represent the effect of wetlands by a delay factor $f_{W,i}$ which directly affects the flow velocities v_i of overland flow (index i=o) and riverflow (index i=r), then the following shape is yielded for $f_{W,i}$ as a function of the wetland fraction p_W within a gridbox: $f_{W,i}$ is close to one for small values of p_W. For larger p_W, the curve of $f_{W,i}$ declines steadily and finally changes its behaviour into an asymptotic decrease to the minimum value of $f_{W,i}$ for large values of p_W. This minimum value of $f_{W,i}$ corresponds to the maximum delay for $p_W = 100\%$. This general shape (see Fig. 4) can be described by the tanh-function, so that this function is used in the conceptual approach shown in equation 16.

$$f_{W,i} = 1 - \frac{1}{2}\left(1 - \frac{v_{W,i}}{v_{0,i}}\right) \cdot (\tanh(4\pi \cdot (p_W - p_c)) + 1) \tag{16}$$

Here, $v_{0,i}$ corresponds to the flow velocity (according to equation 14) used in the HD model parameterizations without the inclusion of wetlands. $v_{W,i}$ is the flow velocity for a 100% coverage. p_c is a treshold value of p_W which characterizes the decline of $f_{W,i}$ with increasing p_W. The parameters of the cascades of linear reservoirs of overland flow and riverflow are modified according to equation 17:

14

$$k_i = \frac{\Delta x}{f_{W,\,i} \cdot v_{0,\,i}} \qquad\qquad n_i = 1 \qquad\qquad (17)$$

As a simplification we set $v_{W,\,o}$ to a tenth of $v_{W,\,r}$ and we define $v_W = v_{W,\,r}$.

3.4. Parameterization for the fractional coverage of lake area

Recently increased interest was focused on large lakes not only due to their role as a water resource but also due to their qualification as indicator for climate change (*Birkett and Mason*, 1995). For several large lakes, the residence times of water within the lake are known. In a discharge model, the representation of every lake by a linear reservoir would be a proper physical approach but a global dataset of residence times is not available for all lakes.

At the MPI, a global lake dataset is available with a resolution of 1x1° which describes the fractional coverage of lake area for every gridbox. The dataset is based on data of *Cogley* (1987, 1991, 1994) which contains some inconsistencies. Therefore the dataset was modified using several land surface datasets of the 'International Satellite Land Surface Climatology Project' (ISLSCP; *Meeson et al.*, 1995) and lake data of the *CIA* (1992). The modified global lake area dataset is shown in Fig. 5.

As for wetlands no widely accepted parameterization exists which may represent the influence of fractional lake area on the different flow processes. In *Charbonneau et al.* (1977), a discharge transfer coefficient depends exponentially on the reciprocal lake area fraction. However, several tests with a similar delay factor did not yield satisfactory results.

For the influence of lakes within a gridbox on the discharge, principally the same considerations are valid as for the effect of wetlands (cf. Sect. 3.3.2.). Therefore the same conceptual model was chosen, so that the equations 23 and 24 are applied:

$$f_{L,\,i} = 1 - \frac{1}{2}\left(1 - \frac{v_{L,\,i}}{v_{0,\,i}}\right) \cdot \left(\tanh(4\pi \cdot (p_L - p_c)) + 1\right) \qquad\qquad (18)$$

and

$$k_i = \frac{\Delta x}{f_{L,\,i} \cdot v_{0,\,i}} \qquad\qquad n_i = 1 \qquad\qquad (19)$$
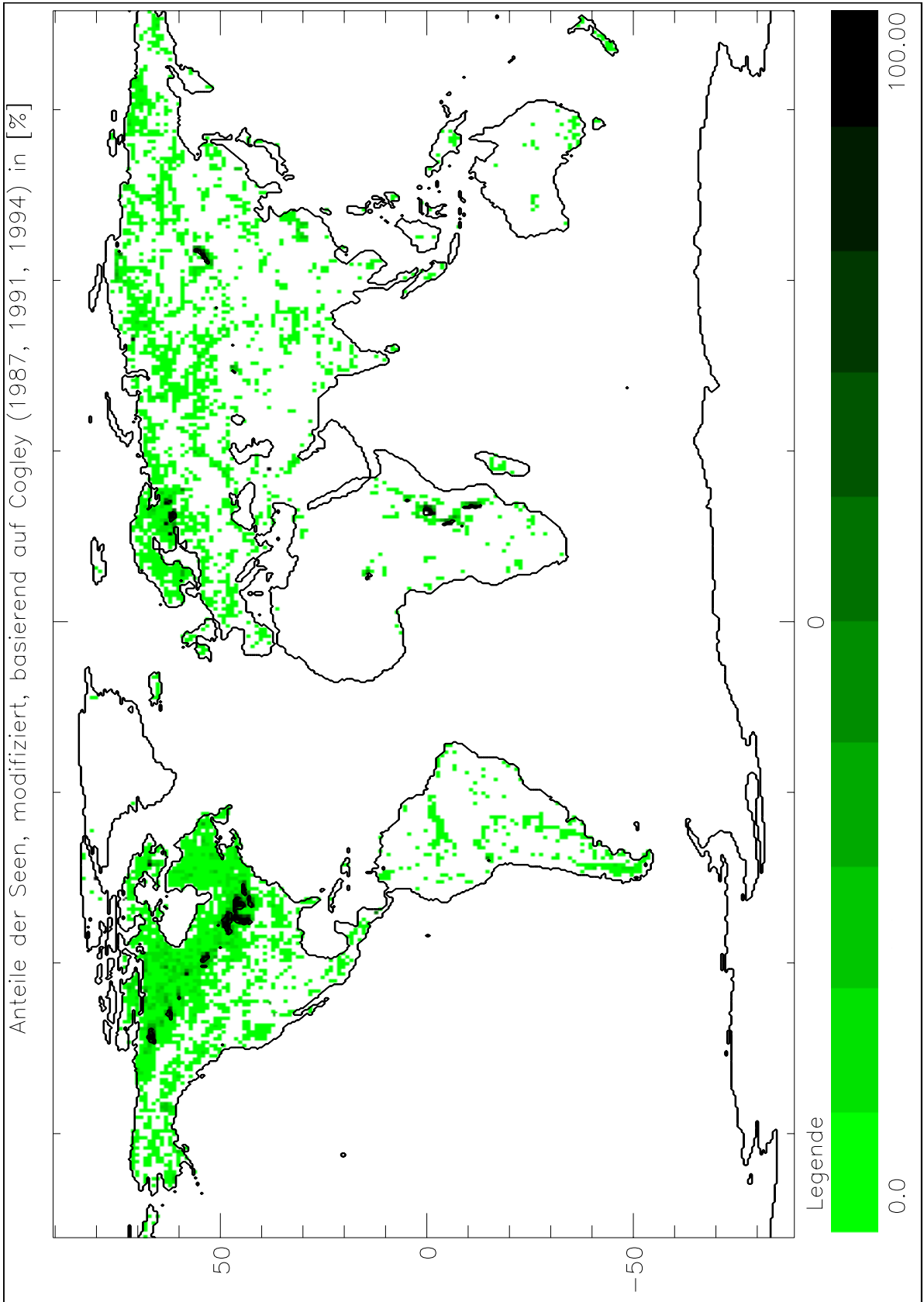
Fig. 5. Modified global distribution of fractional lake area based on *Cogley* (1987, 1991, 1994).
Scaling in 10% steps

The variable designations of the equations 18 and 19 correspond to the designations of equation 16 and 17, except for the fact that the fractional lake area (index L) is used instead of the wetland fraction (index W) within a gridbox. Again, $v_{L,\,o}$ was set to a tenth of $v_{L,\,r}$ as a simplification, and $v_L = v_{L,\,r}$ is defined.

As for Sect. 3.3., $p_c = 50\%$ yields a realistic decline of $f_{L,\,i}$ with increasing p_L. Therefore only v_L had to be calibrated. This was done in the catchments of the St. Lawrence river and of the Gulf of Finland. The discharge from both catchments is significantly dominated by the influence of large lakes, i. e. the Great Lakes (the lakes Superior, Michigan, Huron, Erie and Ontario) for the St. Lawrence and primarily the Lake Ladoga for the Gulf of Finland. For the calibration, we have chosen runoff and drainage fields based on simulated precipitation since no observed data were available. Due to the large dampening effect of the lakes on the variability of discharge coming from the lakes, the model is insensitive to day-to-day variations of precipitation. Only the total amount of precipitation has to be simulated fairly well to allow an adequate discharge simulation for both catchments.

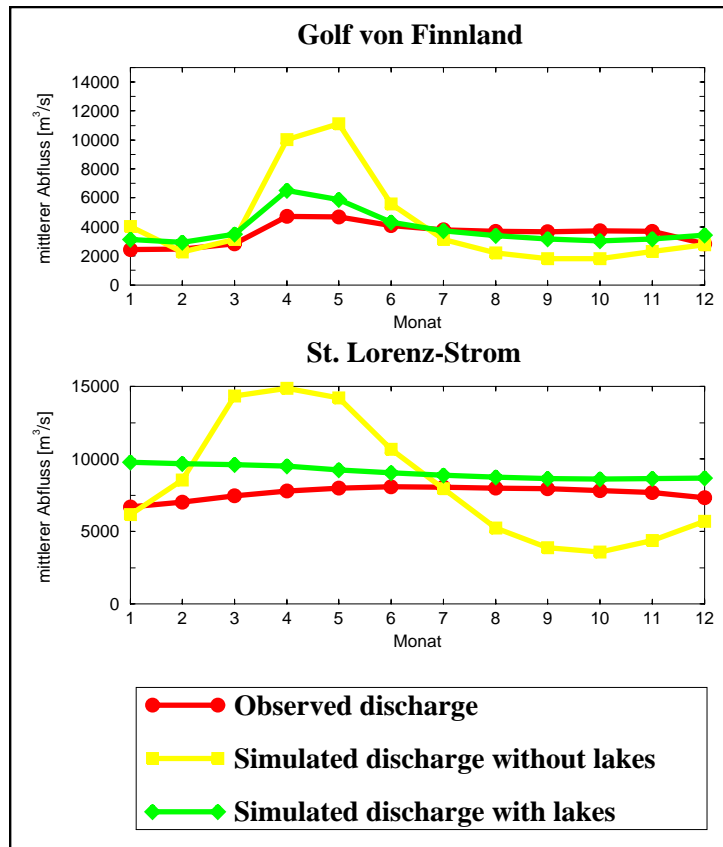**Fig. 6.** Observed and simulated discharge with and without a parameterization of lakes in both catchments that are used for the calibration of v_L..

For the calibration, the runoff and drainage fields of an ECHAM4 control simulation were used as input for the HD model. The calibration was performed manually and yielded a value 0.01 m/s for v_L. Fig. 6 shows a comparison of simulated discharges with and without the usage of

the lake parameterization for the St. Lawrence river and the Gulf of Finland. For both catchments, the simulated discharge is considerably improved by the lake parameterization. The negative trend in the simulated discharge of the St. Lawrence river is caused by larger differences in the content of the linear model reservoirs before and after the five years of the discharge simulation which indicates that the model reservoirs in this regions are not realistically initialized.

The generation procedure of the initialization fields of the linear reservoirs (cf. Sect. 3.2.) converges satisfactorily for most of the large rivers after a few iterations. Only rivers with strong retention characteristics within the catchment (e. g. due to the influence of wetlands or lakes) require more iterations. The correct initialization of the reservoirs is especially difficult for the St. Lawrence, because the extremely large retention capacity of the Great Lakes causes very large retention times of the model reservoirs. Therefore these reservoirs need a lot of time to spin-up with this generation procedure.

As for wetlands, the resolution of $1°$ of the global lake dataset may be too coarse in some regions to yield accurate discharge simulations. Therefore the generation of a global $0.5°$ lake area dataset would be desirable.

4. Technical details of the HD model

In Sect. 4.1. the basic characteristics of the different source code routines are described and their main variables are listed. Sect. 4.2. presents a typical job script to run the HD model on the Cray J90 (LAKE). Furthermore, the file *hdmodel.inp* is listed as an example for the file *global.inp* that must be used to set some paths and switches in the model.

4.1. Source code characteristics

4.1.1. Main program

```
PROGRAM GLOBSIM
```

The following routines are necessary to compile and link the HD model.

```
globsim.f              (latest version: globsim9.f)
globuse.f
echread.f              (version corresponding to model input data)
infread.f
pcom05.for             (include file pcom.for used in infread.f)
```

The source code file *globsim9.f* contains the main program routines of the HD model. Some switches and parameters of the HD model are set in the file *global.inp* (cf. Sect. 4.2.1.). Herein, the names of all model input files have to be specified, too.

Several output files are generated during a discharge simulation. The simulated discharge is written into the binary direct access file *meanflowbin.dat*. The daily discharges into the Bothnian Bay (first column) and Bothnian Sea (second column) are written into the ascii file *iso.dat* as a control output. The daily outflow from 3 gridboxes is written into the ascii file *global.log* as a further control output. Their coordinates were chosen for testing reasons and may be freely changed in the source code on user's demand.

For ISWRIT ≠ 0, the reservoir contents of the different flow processes are written after ISWRIT time steps. This writing is repeated every ISWRIT time steps as long as the simulation continues. Typical ISWRIT is set to NSTEP, so that the discharge simulation may be continued with a further run of the program. Then these reservoir contents can be used as initialization values if the corresponding file names are specified in *global.inp* and ISREAD is set to 1 (cf. Sect. 4.2.1.). The reservoir contents of overland flow and riverflow are written into the file *<IStep>reservoir_lr.dat* where *<IStep>* designates the actual time step. The content of the linear baseflow reservoir is separately written into the file *<IStep>reservoir_base.dat*. In addition to the reservoir contents, the actual global discharge array (inflow into a gridbox) is

written into the file *<IStep>inflow.dat*.

```
c ******* List of variables ******************************************
c ***
c ******* Logical units
c ***
c *** LU        General logical unit for file streams to ascii files = 20
c *** LULOG     Logical unit of log file global.log = 11
c *** LURUN     Logical unit of binary runoff input file = 50
c *** LUBAS     Logical unit of binary drainage input file = 51
c *** LURES     Logical unit of binary reservoir in- and output files = 52
c *** LUBIN     General logical unit for file streams to binary files = 53
c *** LUOUT     Logical unit of binary discharge output file = 60
c ***
c ******* Switches
c ***
c *** IFORM     Type of file format for global input data arrays used in
c ***           subroutine GLREAD
c ***           1 = Service format with origin at Greenwich
c ***           2 = as 4 without header
c ***           3 = ascii file in Waveiso2-Format = '(8F8.2)' per data row
c ***           4 = Service format with origin at date line
c ***
c *** IFLOW     Type of flow
c ***           1 = Overland flow
c ***           2 = Riverflow
c ***
c *** IOUT      Type of output which specifies the size of the time step the
c ***           output is averaged and written.
c ***           1 = 30-Day Averages
c ***           2 = Decadal Averages
c ***           3 = Weekly Averages
c ***           4 = Monthly Averages (without leap years)
c ***
c *** IBASE     Switch for baseflow computation (ON = 1, Out = 0)
c *** ISREAD    Switch for reservoir initialization (ON = 1, Out = 0)
c *** ISWRIT    Number of time steps the reservoir contents have to be saved
c ***           (0 = No saving is done)
c *** IQUE      Log variable (0 = No log output to standard out)
c ***
c ******* Numbers and array dimensions
c ***
c *** ISTEP     Counter of the actual time step
c *** NSTEP     Number of time steps of the discharge simulation
c *** NMEMLF    Maximum number of overland flow reservoirs per cascade = 5
c *** NMEMRF    Maximum number of riverflow reservoirs per cascade = 10
c *** NL        Number of longitudes = 720 at 0.5 degree resolution
c *** NB        Number of latitudes = 360 at 0.5 degree resolution
c *** NBYTE     Number of Bytes per global data array + header. This is
c ***           necessary for the direct access output (8 Bytes = 1 word)
c *** NMAX      Maximum number time steps = 5000 that may be used for NSTEP
c ***
c *** MM        Number of sub time steps for the computation of riverflow
c ***           (typical 4 which equals a time step of 6 hours)
```

```
c ***
c ******* Global real arrays with dimension NL*NB:
c ***
c *** FINP     Local input or dummy array at time step ISTEP
c *** FDATA    Local output array at time step ISTEP
c *** FINFL    Inflow into a gridbox from other gridboxes at time step ISTEP
c *** FDIR     River direction file
c *** FLAG     Land sea mask
c *** ALF_K    Retention times k of overland flow reservoirs in days
c *** ALF_N    Number of reservoirs in the overland flow reservoir cascade
c *** ARF_K    Retention times k of riverflow reservoirs in days
c *** ARF_N    Number of reservoirs in the riverflow reservoir cascade
c *** AGF_K    Retention times k of baseflow reservoirs in days
c *** FGMEM    Baseflow reservoir in [m^3/s * days]
c *** FRIV     Mean inflow per gridbox in m^3/s
c ***
c *** FINMEM   Content of linear reservoirs of the riverflow reservoir
c              cascade. Array has dimension NL * NB * NMEMRF. In order to
c              allow an easier computation the content unit is m^3/s * day.
c *** FRUMEM   Content of linear reservoirs of the overland flow reservoir
c              cascade. Array has dimension NL * NB * NMEMLF. In order to
c              allow an easier computation the content unit is m^3/s * day.
c ***
c ******* Other arrays
c ***
c *** IHEAD    Header array used in the service format: Integer array with a
c              dimension of 8
c *** F1       Outflow at coordinate 1 for log file iso.dat
c *** F2       Outflow at coordinate 2 for log file iso.dat
c *** AREA     Real array of gridbox area per latitude in m^2, dimension NB
c *** DELB     Constant latitudinal gridbox length in m
c *** DELTAL   Array of longitudinal gridbox lengths per latitude in m
c ***
c ***
c ******* File name strings
c ***
c *** DNRUN    Binary global runoff input array
c *** DNBAS    Binary global drainage input array
c *** DNRES    global initial reservoir array for overland and riverflow
c *** DNDIR    River direction file
c *** DNMAS    Land sea mask
c *** DNINP    General input file
c *** DNOUT    General output file as well as discharge output file. The
c              latter is the binary direct access file meanflowbin.dat
c *** DNAREA   Global gridbox length and area arrays
c *** DNGINP   Input file for parameter initialization = global.inp
c ***
c ******* Indices
c ***
c *** JL       Longitudinal index
c *** JB       Latitudinal index
c *** IL       relative change in JL for the routing to the next gridbox
c *** IB       relative change in JB for the routing to the next gridbox
c *** JLNEW    JL+IL
c *** JBNEW    JB+IB
```

```
c *** IRE       Internal counter for the saving of the reservoirs
c ***



      SUBROUTINE KASGLOB(LUINP, II, FINP, YMOD, A_K, A_N,
     &                   AREA, IFLOW, NL, NB, FMEM, MM, IQUE)
```

Routine that performs the simulation of overland flow and riverflow using the hydrological model approach of the cascade of n equal linear reservoirs (cf. Sect. 2.). Depending on IFLOW either overland flow (1) or riverflow (2) is considered. For overland flow, the input array is read from the file stream with the logical unit LUINP using the routine ECHREAD, while for riverflow the input array has to be delivered by the calling routine.

The general time step is one day but riverflow is usually computed with a shorter time step to pay regard to the fact that the chosen time step is the minimum travel time through a 0.5 degree gridbox. Therefore the 1 day time step is separated into MM (with a typical value of 4) sub time steps for the computation of riverflow.

```
c ******* List of variables
c ***
c *** LUINP    Logical unit of binary input file for overland flow
c *** II       Number of the actual time step
c *** FINP     Input array for the considered flow process. Real array with
                dimension NL*NB
c *** YMOD     Output array of the simulated flow process. Real array with
                dimension NL*NB
c *** A_K      Retention times k of the reservoirs in days of the considered
                flow process. Real array with dimension NL*NB
c *** A_N      Number of reservoirs in the reservoir cascade of the
                considered flow process. Real array with dimension NL*NB
c *** AREA     Real array of gridbox area per latitude in m², dimension NB
c *** IFLOW    Specification of the considered flow process
c ***          1 = Overland flow
c ***          2 = Riverflow
c *** NL       Number of longitudes
c *** NB       Number of latitudes
c *** FMEM     Content of linear reservoirs of the considered cascade
                dependent on the flow process. The real array has the
                dimension NL * NB * Var. The third dimension Var has to be
                specified in the calling routine. In order to allow an easier
                computation the content unit is m³/s * day.
c *** MM       Number of sub time steps for the computation of riverflow
                (typical 4 which equals a time step of 6 hours)
c *** IQUE     Log variable (0 = No log output to standard out)
c ***


      SUBROUTINE GMITWRI(LUOUT, IHEAD, FOUT, ISTEP, IOUT, IQUE)
```

Routine that averages the discharge over NSTEP time steps and writes the averaged global discharge arrays into a binary direct access file. The stream using the corresponding logical unit LUOUT has to be opened in the calling routine. The writing is done using the routine INFWRITE.

```
c ******* List of variables
c ***
c *** LUOUT    Logical unit that is connected to a binary direct access
                output file which has to opened in the calling routine. Except
                for the direct access, the file is written as a service format
                file.
c *** IHEAD    Header array used in the service format: Integer array with a
                dimension of 8
c *** FOUT     Data array at 0.5 degree resolution: Real with a dimension of
                NL*NB
c *** ISTEP    Actual time step of output data array FOUT
c *** NL       Number of longitudes in data array = 720
c *** NB       Number of latitudes in data array = 360
c *** IOUT     Type of output which specifies the size of the time step the
                output is averaged and written.
c ***           1 = 30-Day Averages
c ***           2 = Decadal Averages
c ***           3 = Weekly Averages
c ***           4 = Monthly Averages (without leap years)
c ***          -1 = Averaging over the remaining time steps after the last
                     writing of averages and return
c *** IQUE     Log variable (0 = No log output to standard out)
c ***
c ******* Local variables
c ***
c *** IREC     Number of the actual writing time step whose size is specified
                by IOUT.
c *** IMON     Number of the month and new record switch (Yes/No)
c ***          -1 ==> Start of a new record
c *** IDAY     Local time step counter of record IREC+1
c ***
c ***
c
```

SUBROUTINE CHIMASK(LU, DNACAT, LMASK, NL, NB, IQUE)

This routine reads 0.5 degree grid coordinates from the ascii file DNACAT and sets the corresponding global mask as a Boollean array LMASK(NL,NB). The routine was used only for test reasons and is not called in the current code version. The routine is stored in the main program file (*globsim9.f*).

4.1.2. Subroutines

SUBROUTINE ECHREAD(LUINP, IHEAD, TOCODE, ISTEP, IQUE)

This routine reads the input arrays of runoff or drainage (at time step ISTEP) in the specified resolution, e.g. T42, from the data stream with the logical unit LUINP. It interpolates the array to the 0.5 degree resolution. Several files exists that contains this routine to account for different resolutions:

echread.f	T42 using unblocked service format files as input
echT42.f	T42 using service format files as input
echT63n.f	T63 using service format files as input, e.g. NCEP reanalysis data
echam106.f	T106 using service format files as input
echT106.f	T106 using service format files as input that are mirrored from North to South
hirread.f	HIRHAM rotated 0.5 degree grid over Europe using service format files
remoread.f	REMO rotated Europamodell grid at 0.5 degree resolution using service format files

```
c ******* List of variables
c ***
c *** LUINP   Logical unit that is connected to a binary input file which
                has to opened in the calling routine. The binary input file is
                usually in the service format.
c *** IHEAD   Header array used in the service format: Integer array with a
                dimension of 8
c *** TOCODE  Interpolated data array: Real array with a dimension of 720*360
c *** ISTEP   Chosen time step
c *** IQUE    Log variable (0 = No log output to standard out)
c ***



SUBROUTINE GLREAD(LU, DNAM, IFORM, FWERT, NL, NB, IQUE)
```

This routine reads a single global data array with a dimension of NL*NB. The coordinate with the indices (1, 1) is usually (IFORM=4) assigned to the gridbox which has the crossing of the North pole and the date line in the North-West corner. At 0.5 degree resolution this is the gridbox with the centre coordinates of 89.75 N and -179.75 W. The routine is stored in the file *globuse.f.*

```
c ******** List of variables
c ***
c *** LU      Logical unit of file stream to DNAM
c *** DNAM    Name of data file
c *** FWERT   Data array: Real with a dimension of NL*NB
c *** IFORM   Type of file format
c ***         1 = Service format with origin at Greenwich
c ***         2 = Just the global array in a simple binary file, e.g. on REGEN
c ***         3 = ascii file in Waveiso2-Format = '(8F8.2)' per data row
c ***         4 = Service format with origin at date line
c *** NL      Number of longitudes in data array
c *** NB      Number of latitudes in data array
c *** IQUE    Log variable (0 = No log output to standard out)
c
c
SUBROUTINE GLWRITE(LU, DNAM, IFORM, FWERT, NL, NB, IQUE)
```

This routine writes a single global data array with a dimension of NL*NB. The coordinate with the indices (1, 1) is usually (IFORM=4) assigned to the gridbox which has the crossing of the North pole and the date line in the North-West corner. At 0.5 degree resolution this is the

gridbox with the centre coordinates of 89.75 N and -179.75 W. The routine is stored in the file *globuse.f*. Its variables correspond directly to the variables used in the routine GLREAD.

```
SUBROUTINE AREAREAD(LU, DNAM, AREA, DELTAL, DELB, JBMAX, IQUE)
```

This routine reads the global gridbox length and area arrays that belong to the runoff and drainage input arrays. The routine is stored in the file *globuse.f*.

```
c ******** List of variables
c ***
c *** LU       Logical unit of input file with input data array
c *** DNAM     Name of data file which has to be specified in global.inp
c *** AREA     Real array of gridbox area per latitude in m², dimension JBMAX
c *** DELB     Constant latitudinal gridbox length in m
c *** DELTAL   Array of longitudinal gridbox lengths per latitude in m
c *** JBMAX    Number of latitudes = dimension of arrays AREA and DELTAL
c *** IQUE     Log variable (0 = No log output to standard out)
c
```

```
SUBROUTINE ISOWRITE(LU, DNAM, F1, F2, NMES)
```

Subroutine for the writing simple ascii file with two columns of data. The the two columns contain the data values of the arrays F1 and F2, respectively. The routine is stored in the file *globuse.f*

```
c ******** List of variables
c ***
c *** LU       Logical unit of output file
c *** DNAM     Name of data file
c *** F1       Data array for first column, dimension of NMES
c *** F2       Data array for second column, dimension of NMES
c *** NMES     Number of rows = dimension of the two data arrays
c
```

```
SUBROUTINE GLOBINP(LU, DNAM, CINI, FINI, ZEILE, IQUE)
```

This routine is used in the initialization part of the main programs. It reads the specified input parameter from the input file DNAM which is set to *global.inp* in the main program. The routine is stored in the file *globuse.f*

```
c ******** List of variables
c ***
c *** LU       Logical unit for the file stream to DNAM
c *** DNAM     Name of the input file. Although this is set in the main program
c              to global.inp it is used as a routine variable since this
c              routine can be used by different programs that read its
c              parameters from input files.
c *** CINI     Search string that specifies the parameter name. It must have
```

```
              6 characters (Spaces are allowed). If CINI(1:1) = "T", a string
              variable is searched and returned in ZEILE. Otherwise a real
              value is returned in FINI.
c *** FINI     Parameter value if real value is requested
c *** ZEILE    Parameter string if string variable is requested
c *** IQUE     Log variable (0 = No log output to standard out)
c
c ********* Typical file structure of input file DNAM
c ***
c *** CINI1: Comment for FINI1
c *** FINI1
c *** CINI2: Comment for FINI2
c *** FINI2
c ***   :              :
c ***   :              :
c
```

```
SUBROUTINE INFWRITE(LUINP, IHEAD, FDATA, ISTEP, NSM, IQUE)
```

Routine that writes the data array FDATA into a binary direct access output file that has to be opened in the calling routine. The routine may be used for a cyclic writing into a direct access file with a fixed size of NSM datasets. The following table shows an example for the cyclic writing.

Dataset No.		Record No.
1	-->	1
2	-->	2
NSM	-->	NSM
NSM+1	-->	1
NSM+2	-->	2

In order to allow normal sequential writing NSM has to be set to ISTEP in the calling routine. The routine is stored in the file *infread.f*.

```
c ******* List of variables
c ***
c *** LUINP    Logical unit that is connected to a binary direct access
              output file which has to be opened in the calling routine.
              Except for the direct access, the file is written as a service
              format file.
c *** IHEAD    Header array used in the service format: Integer array with a
              dimension of 8
c *** FDATA    Data array: Real with a dimension of NL*NB
c *** ISTEP    Number of actual writing time step
c *** NSM      Maximum number of datasets in output file
c *** IQUE     Log variable (0 = No log output to standard out)
c ***
c ******* Include of parameter NL, NB using the include file pcom.for
c ***
c *** NL       Number of longitudes in data array
c *** NB       Number of latitudes in data array
c *** FLORG    Longitude of grid origin (not used)
c *** FBORG    Latitude of grid origin (not used)
```

```
c *** FSCAL    Scaling factor (not used)
c ***
c ******* Local variable
c ***
c *** IREC    Actual record number in output file
```

4.2. Running the HD model software

In this section, a typical job script to run the HD model on the CRAY J90 computer system is presented. Here, the HD model is applied to the input fields of an atmospheric simulation using ECHAM4-T42 so that the routine *echread.f* has to be used. The user of the HD model has to change the paths of the `<source code directory>` and the `<output directory>` to fit into his own environment. The source code, two examples for the input parameter file *global.inp* (cf. Sect. 4.2.1.) and several examples for the following script (such as *fgl_t42*, *fgl_t106*, etc.) to run the HD model are stored in the file *HDmodel_source.tar*.

```
# QSUB-lT 16000                                    # 5 year run with T42 input = ca. 13000 s of CPU
# QSUB-lM 8Mw                                      # 5 year run with T106 input = ca. 22000 CPU s
# QSUB-r fglob9                                    # Job title
# QSUB-o <output directory>                        # Directory for job log file output
#
#    Execution of the program GLOBSIM Vs. 3.1 in F90
#
#
        echo „GLOBSIM Version 3.1 on the LAKE (with globsim9.f)“
        setenv NCPUS 4                             # max. avg. CPU usage: 2.49
#
        cd <output directory>                      # directory for output files
        cp -p <source code directory>/globsim9.f . # Main program file
        cp -p <source code directory>/globuse.f .  # Read parameter fields
        cp -p <source code directory>/infread.f .  # Write output routine
        cp -p <source code directory>/echread.f .  # Read input time series
#
        cp -p <source code directory>/pcom05.for ./pcom.for    # Include file for infread.f
        cp -p <source code directory>/hdmodel.inp ./global.inp # Input parameter file: global.inp
#
        f90 -Otask3,vector3,scalar3 -r2 -o GLOBSIM globsim9.f globuse.f echread.f infread.f


#
assign -R
###     assign -s unblocked u:50       # binary runoff input file - unblocked file
assign -N ieee -F f77 u:50             # binary runoff input file - ieee file
###     assign -s unblocked u:51       # binary drainage input file - unblocked file
assign -N ieee -F f77 u:51             # binary drainage input file - ieee file
assign -s unblocked u:52               # binary reservoir file
assign -s unblocked u:53               # binary unspecified file, i.e. for global parameter fields
assign -s unblocked u:60               # binary discharge output file - direct access file
#
```

27

```
ja
GLOBSIM
ja -st
qstat -a
```

4.2.1. Input file *global.inp*

The file *hdmodel.inp* is listed as an example for the file *global.inp* that must be used to set some paths and switches in the model. The user of the HD model has to change the path of the `<data directory>` to fit into his own environment.

Initialization file for program GLOBSIM, i.e. for routine GLOBINP on the LAKE:

```
IQUE  : Comment Switch ( 0 = No Comment )
0
IOUT  : Averaging Switch for Program output: 1=30d, 2=10d, 3=7d,4=Monthly
1
IBASE : Baseflow Switch ON (1) or Out (0)
1
ISREAD: Initialization Switch for Reservoir fields: 0 = Out, otherwise ON
1
ISWRIT: Saving of Reservoir fields: 0 = Out, otherwise time step of saving
1800
NSTEP : Number of Time Steps of the discharge simulation: 1 to NSTEP
1800
NSMLF : max. number of overland flow reservoirs = dummy relict for testing
5
NSMRF : max. number of riverflow reservoirs = dummy relict for testing
10
MM    : Number of Sub-time steps for riverflow: 1 = 1 day, 4 = 6 hours
4
ISOLOG: Log-output in 2-column ascii file (= iso file), only in regional model version (0=No, 1=Both.)
1
TDNRUN: File name of binary global runoff input array
<data directory>/runoff_2639024_6-10.dat
TDNBAS: File name of binary global drainage input array
<data directory>/drain_2739024_6-10.dat
TDNMAS: File name of land sea mask (0.5 degree)
<data directory>/slm-lakes
TDNRES: File name of global initial reservoir array for overland and riverflow
<data directory>/reservoir.dat
TDNDIR: File name of river direction file
<data directory>/rivdir.dat
TDNLFK: File name of overland flow k-Parameter array
<data directory>/over_k.dat
TDNLFN: File name of overland flow n-Parameter array (usually constant = 1)
<data directory>/over_n.dat
TDNRFK: File name of riverflow k-Parameter array
<data directory>/riv_k.dat
```

TDNRFN: File name of riverflow n-Parameter array
<data directory>/riv_n.dat
TDNGFK: File name of baseflow k-Parameter array
<data directory>/bas_k.dat
TDNGSP: File name of linear baseflow-reservoir for initialization
<data directory>/bas_reservoir.dat
TDNARE: File name of global gridbox length and area arrays
<data directory>/fl_dp_dl.dat
The End

4.2.2. Check list for running the HD model

1. Put the source code files (stored in *HDmodel_source.tar*) into the `<source code directory>`.

2. Put the input datasets (stored in *HDmodel_inputdata.tar*) into the `<data directory>`. Be aware that all global input data files are in an unblocked CRAY format.

3. Put the input time series of runoff and drainage into the `<data directory>`. Be aware that the corresponding files must be in the appropriate file format specified in Sect. 4.1.2. for the subroutine ECHREAD. Check which file format your input files have, e. g. the unblocked CRAY format or the *ieee* format (cf. the assign statements in the job script in Sect. 4.2.).

4. Create the job script that runs the HD model according to Sect. 4.2. by changing one of the example scripts or by generating a new one.

   - Choose the appropriate interpolation routine ECHREAD (see Sect. 4.1.2.) that transforms the input time series to the 0.5 degree resolution.

   - Check all paths in the job script

5. Create the file *global.inp* according to Sect. 4.2.1. and check all paths within this file. An example file is included in *HDmodel_source.tar*.

6. Check if there is enough space on the chosen hard disk on which you have specified the `<output directory>`. As a remark, a five year simulation of the HD model will generate about 400 MB of output.

7. Run the HD model by submitting the job script with the unix command *qsub*.

5. Evaluation software

This section deals with some software routines that may be helpful in evaluating the simulated discharges as well as some of the atmospheric input fields. In the following, the user of a particular job script has to change the paths of the `<source code directory>`, `<data directory>` and the `<output directory>` to fit into his own environment. The script and some files used in the scripts are stored in *HDmodel_eval.tar*. The catchment files used in the script *FQAREA* (Sect. 5.3.) are stored *hdmodel_inputdata.tar*.

5.1. Computation of global monthly mean climatologies - Script *FMONMIT*

The script ***FMONMIT*** computes global monthly mean climatologies.

```
#!/bin/sh
# QSUB-r fklmomi
# QSUB-q M4
# QSUB-o <output directory>              # Directory for job log file output
set -ex
#
#   ******** Computation of global monthly mean climatologies - Script for the CRAY J90
#
#      Input: Binary direct access file DNAM with monthly mean values, as computed by program GLOBSIM.
#      Output: Sequential global binary files with climatological values for the 12 months
#          Files:
#          mean.dat            climatological monthly mean values
#          sigma.dat           standard deviation of the climatological monthly means
#
#      File mean.dat may be used as input file for script FQDAT
#
#      Version 1.1. -- by Stefan Hagemann -- Based on a program of Uwe Schulzweida
#
cd $TMPDIR
EXPNO=70000                             # Experiment number - just a dummy
NYF=6                                   # First year of input time series
NYL=10                                  # Last year of input time series
#
DATA=<data and output directory>        # Working directory with input file and target directory for output files
#
#   ******** Input file has to be specified here
DNAM=$DATA/meanflowbin.dat
echo „ Input-Datei = $DNAM“
#
RES=30                                  # Resolution constant = number of latitudes of the used grid
#
if [ $RES -eq 30 ]   ; then   LAT=360; fi   # Resolution of 0.5˚ degrees
if [ $RES -eq 21 ]   ; then   LAT=32 ; fi
if [ $RES -eq 42 ]   ; then   LAT=64 ; fi
if [ $RES -eq 63 ]   ; then   LAT=96 ; fi
```

```
if [ $RES -eq 106 ]  ; then   LAT=160 ; fi
#
#------------------------------------------------
NYEARS=`expr $NYL - $NYF + 1`
NR=$NYF
#
assign -R
#
# ***
#
assign -s unblocked u:50                    # assign for input file
assign -s unblocked u:60                    # assign for output file mean.dat
assign -s unblocked u:61                    # assign for output file sigma.dat

cat >pmomi.f << END_OF_FILE
<Here, the FORTRAN code is included which does the work described above>
END_OF_FILE

cp -p  <source code directory>/infread.f .
cp -p  <source code directory>/pcom05.for ./pcom.for
f90 -r2 -o MONMIT pmomi.f infread.f
ja
MONMIT
ja -st
#
cp fort.60 $DATA/mean.dat
cp fort.61 $DATA/sigma.dat
#
ls -l
exit
```

5.2. Computation of climatological monthly mean discharges- Script *FQDAT*

The script **FQDAT** computes climatological monthly mean discharges for some of the largest
rivers of the world as well as for some smaller catchments of interest.

```
#!/bin/sh
# QSUB-lT 200
# QSUB-lM 4Mw
# QSUB-o <output directory>                # Directory for job log file output
set -ex
# ******** Computation of climatological monthly mean discharge files for some of the largest rivers of the
#             world as well as for some smaller rivers of interest - Script for the CRAY J90
#
#      Input: Sequential global binary file with climatological values for the 12 months, as computed by the
#             Script FMONMIT, e.g. file mean.dat
#             tracer.inp     Input file with discharge coordinates of the different rivers
#             modflow        Input file with river abbreviations
#
#      Output: ASCII files with 12 values for the 12 months
#             Files are readable by the script FPLOT which generates XVGR plot files
```

```
#
#   ******** Version 2.1 -- Oct. 1996  --  by Stefan Hagemann -- Based on a program of Uwe Schulzweida
#
#
cd $TMPDIR
#
#   ******** Insert experiment number of discharge simulation, z.B.: 2522032
EXPNO=6823236                    # Experiment number of discharge simulation
NYF=02                           # First year of input time series
NYL=06                           # Last year of input time series
#
DATA=<data and output directory>         # Working directory with input data file
cp $DATA/mean.dat fort.40                         # Input file with climatological monthly means
cp <source code directory>/tracer.inp .           # Input file with discharge coordinates
###cp <source code directory>/acsys05.inp ./tracer.inp   # Input file with ACSYS discharge coordinates
cp <source code directory>/modflow .              # Input file with river abbreviations
#
assign -R
assign -s unblocked fort.40               # Binary input file with climatological monthly means
#
cat > ppunkte.f << EOR
<Here, the FORTRAN code is included which does work described above>
EOR
#
f90 ppunkte.f -o ppunkte.x
ppunkte.x
#
#   ******* Copy output files
#
cat > KOPIEREN << EOR
<Here FORTRAN code is included which is used to copy the output files into the working directory.>
EOR
#
cft77 KOPIEREN
segldr KOPIEREN.o -o KOPIEREN.x
KOPIEREN.x
#
cp ppunkte.f $MFHOME
#
#
ls -l
exit
```

5.3. Computation of climatological monthly mean area integrals- Script *FQAREA*

The script **FQAREA** computes climatological monthly mean values for different variables integrated or averaged over the catchment area for some of the largest rivers of the world as well as for some smaller catchments of interest.

```
#!/bin/sh
# QSUB-lT 200
```

```
# QSUB-lM 2Mw
# QSUB-o <output directory>                 # Directory for job log file output
#
set -ex
##
# ******** Computation of climatological monthly mean area integrated files (different variables) for some of
#               the largest catchments of the world as well as for some smaller catchments of interest.
#               The different variables can be integrated or averaged over the catchments - Script for the CRAY J90
#               The program is similarly structured as FQDAT.
#
#      Input: Sequential global binary file with climatological values for the 12 months, as computed by the
#               Script FMONMIT, e.g. file mean.dat

#               Chiiso files[1] with model catchment coordinates in a directory, the path of which has to be specified
#               in the code section of this script below.
#               File names: <river name>.dth
#               (Remark: The river mouth gridbox (number 0) are excluded from the model catchment for the
#               integration.)
#      or
#               Chiiso files with real catchment coordinates in a directory, the path which has to be specified in the
#               code section of this script below.
#               File names: <river name>.dca
#
#      catchnr.txt            Input file which aligns river names and catchment numbers
#      modflow                Input file with river abbreviations
#      fl_dp_dl.dat           Input files with latitudinal length and area distribution
#      (maskiso.dat)          If the switch REG is set (see below) this input chiiso file is needed which
#                             masks a particular region such as a specific catchment.
#
#      Output: ASCII files with 12 values for the 12 months
#               Files are readable by the script FPLOT which generates XVGR plot files
#
#   ******** Version 1.3  --  Sep. 1996  --  by Stefan Hagemann
#
#
#   ******** Nomenclature for the experiment number EXPNO:

# There is a 2-digit number at the beginning:
#
#      10     Temperature
#      11     Precipitation
#      12     Snowmelt
#      13     accumulated snow
#      14     Standard deviation of temperature
#      15     Evapotranspiration
#      16     Soil moisture
#      17     Incoming solar radiation (Brutto without albedo)
#      25     Sum of runoff and drainage
#      26     Runoff
#      27     Drainage
#
#      40++   HD model discharge for different model versions and model topographies (40-68).
```

---

1. A chiiso file is an ASCII file with three columns: latitude, longitude and data value. The specified coordinate designates the North-West corner of a 0.5°-gridbox.

```
#
# The 2-digit number is followed by a 5-digit number (except for one case with a 3-digit number)
#      10001          Snow depth climatology of *Foster and Davy* (1988), (year 1-1)
#      10002          SMMR snow data, (year 1-1)
#      10010          ISCCP incoming solar radiation (year 1-1)
#      22590          ECHAM4 - T106 AMIP control simulation
#      23236          ECHAM4 - control simulation using climatological SST
#      23241          ECHAM4 - control simulation ???
#      239++          ECHAM4 - Bettina's simulations
#      390++          ECHAM4 - Axel's simulations
#      46001          ECHAM4 - simulation of Bettina
#      47001          ECHAM4 - simulation of Axel
#       609           ECHAM4 - control simulation with different land sea mask
#      500++          ECMWF - reanalysis data (00=year 79-83 or year 79-93, 01=year 81-85)
#      50100          Precipitation climatology of GPCP (*Rudolf et al.*, 1996), (year 86-94)
#      5100+          SL scheme output derived from ECMWF reanalysis data
#      51010          SL scheme output derived from ECHAM4-T106 (using climatological SST)
#      5110+          SL-ARNO scheme output derived from ECHAM4-T106 (using climatological SST)
#      52000          SL scheme output derived from regional (Torneaelven) measurements
#      53000          NCEP - reanalysis data (year 81-85)
#      53100          SL scheme output derived from NCEP - reanalysis data
#      53200          SL scheme output derived from NCEP - reanalysis data using NCEP evapotranspiration
#      60000          REMO - simulation using DWD physics
#      70000          HIRHAM4 - simulation at 50 km resolution for Europe
#
# There are a few exceptions with a alphanumeric experiment number
#      LEG            Precipitation climatology of *Legates and Wilmott* (1990a) (year 1-1)
#      LEGT           Temperature climatology of *Legates and Wilmott* (1990b) (year 1-1)
#
cd $TMPDIR
#
#   ******** Insert experiment number, z.B.: 2522032
EXPNO=1150100                         # Experiment number
NYF=06                                # First year of experiment
NYL=10                                # Last year of experiment
#EXPNO=LEGT                           # Experiment number
#NYF=1                                # First year of experiment, here 1 for climatology
#NYL=1                                # Last year of experiment, here 1 for climatology
#
# ******** Type of catchment for integration: EXT
#      1 =    model catchment      --> Extension .dth
#      2 =    real catchment       --> Extension .dca
EXT=2
#
if [ $EXT -eq 1 ]   ; then   ECO=ar ; fi
if [ $EXT -eq 2 ]   ; then   ECO=ca ; fi
#
# ******** typical file with monthly means: mean.dat or legatesmean.dat
DATA=<data and output directory>                  # Working directory with input data file
cp $DATA/mean.dat fort.40                          # Input file with climatological values for the 12 months
cp <source code directory>/catchnr.txt .           # Input file which aligns river names and catchment
                                                                numbers
cp <source code directory>/modflow .               # Input file with river abbreviations
#
#   ********* Area weighted averages? (and/or unit transformation?)
#   ********* 0 = No, just integrate over the catchment gridboxes (Q, older mean R&D)
```

```
#   ********* 1 = Yes              (e.g. for Sigma-Temp., Radiation)
#   ********* 2 = Yes + Kelvin-transformation to degree Celsius (T)
#   ********* 3 = No, but multiplication with gridbox areas, e.g. m --> m^3 (P, R)
#   ********* 4 = No, but unit transformation with factor UNIT
#   ********* 5 = Yes + unit transformation with factor UNIT (e.g. snowmelt)
#   ********* 6 = No, but division by gridbox areas, e.g. m^3 --> m
#   ********* 7 = No, but division by gridbox areas and multiplication with unit factor UNIT
#   ********* 8 = No, but multiplication with gridbox areas and unit factor UNIT, m-->m^3 (E with -1)
#
FLW=3.
#
#   ******* Unit factor, e.g. m/s --> mm/day * 1000. * 86400.
UNIT=1.
###UNIT=`expr 1 / $UNIT`
#
#
# *******    Reading and applying a regional mask from file maskiso.dat? 0 = No, 1= Yes
#            An application of the masks means that only grid boxes are considered where the mask allocates
#            values not equal to 0.
REG=0
#
### cp <data directory>/maskiso.dat .
cp <source code directory>/fl_dp_dl.dat .
#
assign -R
assign -s unblocked fort.40
###assign -N ieee -F f77 fort.40    # Binary ieee file (e.g. on REGEN)
#
cat > ppunkte << EOR
<Here, the FORTRAN code is included which does the work described above. The user has to
change the following source line
NC=20
DNAFL(IFLOW)(1:NC) = "/mf/m/m214046/river/"
into
NC= <number of characters the path <directory with catchment data files> consists of>
DNAFL(IFLOW)(1:NC) = "<directory with catchment data files>"
to fit the program into his own environment.>
EOR
#
cft77 ppunkte
segldr ppunkte.o -o ppunkte.x
ppunkte.x
#
#   ******* Copy output files
#
cat > KOPIEREN << EOR
<Here FORTRAN code is included which is used to copy the output files into the working directory.>
EOR
#
cft77 KOPIEREN
segldr KOPIEREN.o -o KOPIEREN.x
KOPIEREN.x
#
###cp ppunkte $MFHOME
#
```

```
#
ls -l
exit
```


5.4. Computation of *xvgr* input files - Script *FPLOT*

The script **FPLOT** is used to generate a file that can be read by the plotting software *xvgr* that is suitable for the graphic presentation of climatological monthly mean values. This script needs the ASCII-Files *modflow* and *rivers* for input as well as the source file *PLOTFLOW_MOD.f*. All files have to be located in the directory $HOME/RO.

The scripts handles up to four data files that contain monthly mean values for the corresponding catchments as they are generated with the scripts FQDAT and FQAREA. These data files must be located in the subdirectory <experiment shortcut>. Usually the experiment shortcut comprises of a 7-digit experiment number (see Sect. 5.3.) and a 2-character shortcut that characterizes the type of area the data are assigned.

ar = model catchment area
ca = real catchment area taken from a 0.5 degree dataset
ro = discharge representing a specific measuring station or the river mouth

Therefore, e.g., the discharge data of the simulation 6823236 have to be located in the directory $HOME/RO/23236ro. Further examples are given in the comment for the script variable EXPNi below.

Operating procedure: 1. Edit *FPLOT*
                       2. Run *FPLOT* ==> file *plot.dat* is generated
                       3. Run *xvgr* with: *xvgr plot.dat*

```
#!/bin/sh
#
# ************************************************
# **** Shell script to plot monthly mean data with XVGR V. 2.09
# **** Written by: Uwe Schulzweida, Hans Stefan Bauer
# **** Max Planck Institut fuer Meteorologie 1993/1994
# ************************************************
# **** modified and improved by Stefan Hagemann 1996
# ************************************************
#
# VARIABLES:
#
# CHOICE          River numbers
# EXPN1 ..6       Experiment numbers
# EXPANZ          Number of experiments
# FPATH           Path to the model data
```

```
# FNAME           Name of model data file
# SOURCE          Name of the program which should be manipulated
# ICC             Contains the processed river number
# IX              Counter for the experiment numbers
# IY              Counter for the frames per plot
# IZ              Counter for the plot numbers
# PR              Contains the name of the river
# PRE             Contains the river shortcut
# TYPS            Contains the river numbers
# ANZ             Number of rivers per plot
# SANZ            Number of rivers
#
# source.f        Name of the program, which produces the plotfile
#
# ***********************************************************
#
#
#*********   Example lines for the plotting of data for several catchments
#
#CHOICE="03 05 01 15"        # Bothn. Bay, Bothn. Sea, Baltic Sea, Baltic ex. K.
#CHOICE="07 09 11 13"        # Gulf Finland, Gulf Riga, Baltic Proper, Kattegat
#CHOICE="02 04 06 34"        # Vouksi, Goeta-Vanern, Vistula, Neman
#CHOICE="08 12 14 38"        # Rhein, Elbe, Donau, Don
#CHOICE="30 32 36 44"        # Northern Dvina, Wolga, Ural, Kolyma
#CHOICE="40 42 43 46"        # Lena, Jenissei, Selenga, Ob
#CHOICE="48 51 52 53"        # Amur, Huang Ho, Yangtze, Xun Jiang
#CHOICE="50 54 56 58"        # Indus, Ganges, Godavari, Mekong
#CHOICE="60 64 68 70"        # Nil, Niger, Congo, Sambesi
#CHOICE="66 72 74 62"        # Chari, Limpopo, Oranje, Senegal
#CHOICE="76 78 80 82"        # Mackenzie, Yukon, Nelson, St. Lawrence
#CHOICE="86 85 87 83"        # Mississippi, Ohio, Missouri, St. Lorenz
#CHOICE="84 88 90 81"        # Columbia, Susquehanna, Colorado, Churchill
#CHOICE="94 92 96"           # Amazonas, Orinoco, Sao Francisco
#CHOICE="97 98 99 100"       # Paraguay, Parana, Uruguay, Murray
#CHOICE="10 110 114"         # Oder, Torneaelven, Luleaelven
#CHOICE="150 140"            # ACSYS region, 6 largest ACSYS rivers
#CHOICE="17 19 20 24"        # Upper Seine, Upper Rhone, Loire, Garonne
#CHOICE="22 26 28"           # Po, Ebro, Guadalquivir
#
EXPANZ=3
EXPN1=obs_act               # Observed discharges
###EXPN1=obsnat             # Naturalized flows
###EXPN1=LEGca_1-1          # Precipitation climatology of Legates and Wilmott (1990a)
###EXPN1=1150100ca_86-94    # Precipitation climatology of GPCP (Rudolf et al., 1996)
###EXPN1=1310001ca_1-1      # Snow depth climatology of Foster and Davy (1988)
###EXPN1=1710010ca_1-1      # Incoming solar radiation climatology
###EXPN1=1550001ca_81-85    # Evapotranspiration of ECMWF reanalysis (years 1981-85)
EXPN2=2523236ca_02-06       # Sum of surface runoff and drainage of ECHAM4-T42 control simulation
EXPN3=6823236ro_02-06       # Simulated discharge computed from the control run
EXPN4=
EXPN5=
EXPN6=
AUTOSCALE=ON
FPATH=$HOME/RO
SOURCE=$HOME/RO/PLOTFLOW_MOD.f
TYPS=",,
```

```
ANZ=",,
XFLOWA=",,
XFLOWE=",,
SANZ=0
cp $SOURCE source.f
#
# *** Language (0 = Englisch, 1 = Deutsch)
ILANG=0

# *** Unit variable (Default=Discharge in m^3/s, 10=T, 11=prec.,...
IUNIT=16
#
# *** Text description for Y-Axis
if [ $IUNIT -eq 10 ] ; then UTEXT='monthly average temp. [^C]'
elif [ $IUNIT -eq 11 ] ; then UTEXT='monthly average prec. [m\\\\S3\\\\N/s]'
elif [ $IUNIT -eq 12 ] ; then UTEXT='monthly average snowmelt [m\\\\S3\\\\N/s]'
elif [ $IUNIT -eq 13 ] ; then UTEXT='monthly average snowpack [mm H\\\\s2\\\\NO]'
elif [ $IUNIT -eq 15 ] ; then UTEXT='monthly average evap. [m\\\\S3\\\\N/s]'
elif [ $IUNIT -eq 16 ] ; then UTEXT='monthly average soil water [mm]'
elif [ $IUNIT -eq 17 ] ; then UTEXT='monthly average rad. [W/m\\\\S2\\\\N]'
elif [ $IUNIT -eq 25 ] ; then UTEXT='monthly average R+D [m\\\\S3\\\\N/s]'
else UTEXT='monthly average discharge [m\\\\S3\\\\N/s]' ; fi
#
if [ $ILANG -eq 1 ] ; then
  if [ $IUNIT -eq 10 ] ; then UTEXT='mittlere Temperatur [^C]'
  elif [ $IUNIT -eq 11 ] ; then UTEXT='mittlerer Niederschlag [m\\\\S3\\\\N/s]'
  elif [ $IUNIT -eq 12 ] ; then UTEXT='mittlere Schneeschmelze [m\\\\S3\\\\N/s]'
  elif [ $IUNIT -eq 13 ] ; then UTEXT='mittlere Schneehoehe [mm H\\\\s2\\\\NO]'
  elif [ $IUNIT -eq 15 ] ; then UTEXT='mittlere Evap. [m\\\\S3\\\\N/s]'
  elif [ $IUNIT -eq 16 ] ; then UTEXT='mittlere Bodenfeuchte [mm]'
  elif [ $IUNIT -eq 17 ] ; then UTEXT='mittlere Einstrahlung [W/m\\\\S2\\\\N]'
  elif [ $IUNIT -eq 25 ] ; then UTEXT='mittlerer R+D [m\\\\S3\\\\N/s]'
  else UTEXT='mittlerer Abfluss [m\\\\S3\\\\N/s]' ; fi
fi
#
echo ,,UTEXT= ,, $UTEXT
#
# *** Einheitentext der X-Achse
if [ $ILANG -eq 0 ] ; then XTEXT='month'
elif [ $ILANG -eq 1 ] ; then XTEXT='Monat' ; fi
echo ,,XTEXT= ,, $XTEXT
#
# ************* place Experimentnumbers into source.f *************
#
IX=1
#
for EXPNO in $EXPN1 $EXPN2 $EXPN3 $EXPN4 $EXPN5 $EXPN6
do
echo $EXPNO
cat source.f | sed -e ,,s(EXPNO$IX($EXPNO(,, > tmp
mv tmp source.f
IX=`expr $IX + 1`
done
#
# *************************************************************
#
```

```
IY=1
IZ=1
#
for IC in $CHOICE
do
if [ $IZ -gt 1 ] ; then  TYPS="${TYPS}," ; ANZ="${ANZ}," ; fi
ICC=`echo $IC | cut -d „„" -f1`

if [ $IC -eq 10 ]; then PRE=oder
elif [ $IC -eq 11 ]; then PRE=b_BP
elif [ $IC -eq 12 ]; then PRE=elbe
elif [ $IC -eq 13 ]; then PRE=b_K
elif [ $IC -eq 14 ]; then PRE=donau
elif [ $IC -eq 15 ]; then PRE=b_ex
elif [ $IC -eq 40 ]; then PRE=lena
elif [ $IC -eq 50 ]; then PRE=indus
else PRE=`grep $ICC $FPATH/modflow | cut -d „ „ -f2` ; fi

###PRE=`grep $ICC modflow | cut -d „ „ -f2`


echo $ICC
TYPS=${TYPS}${IC}
NAM=`echo $IC | sed -e „s(,( (g"`
IANZ=0
for i in $NAM ; do  IANZ=`expr $IANZ + 1` ; done
ANZ=$ANZ$EXPANZ
SANZ=`expr $SANZ + $IANZ`
#
# ***************** place river name into source.f *****************
#
if [ $IC -eq 10 ]; then RIVER=Oder
elif [ $IC -eq 11 ]; then RIVER=Baltic_Proper
elif [ $IC -eq 12 ]; then RIVER=Elbe
elif [ $IC -eq 13 ]; then RIVER=Kattegat
elif [ $IC -eq 14 ]; then RIVER=Donau
elif [ $IC -eq 15 ]; then RIVER=Baltic_excl_K
elif [ $IC -eq 40 ]; then RIVER=Lena
elif [ $IC -eq 50 ]; then RIVER=Indus
else RIVER=`grep $IC $FPATH/rivers | cut -d „ „ -f2` ; fi
#
cat source.f | sed -e „s(TITLE$IZ($RIVER(„ > tmp
mv tmp source.f
echo $RIVER
#
# ***************** place filename into source.f *****************
#
for EXPNO in $EXPN1 $EXPN2 $EXPN3 $EXPN4 $EXPN5 $EXPN6
do
EXPNR=`echo $EXPNO | cut -f1 -d"_"`
ZEITRAUM=`echo $EXPNO | cut -d „_" -f2`
FNAME=$FPATH/${EXPNR}/${EXPNR}_${PRE}_${ZEITRAUM}

echo $EXPNO
if [ $EXPNO = ,OBS' ] ; then FNAME=$FPATH/OBS/${PRE}_obs ; fi
if [ $EXPNO = ,obsnew' ] ; then FNAME=$FPATH/obsnew/${PRE}_obs ; fi
```

```
if [ $EXPNO = ,obs_4' ] ; then FNAME=$FPATH/obs_4/${PRE}_obs ; fi
if [ $EXPNO = ,obsislscp' ] ; then FNAME=$FPATH/obsislscp/${PRE}_obs ; fi
if [ $EXPNO = ,obsnat' ] ; then FNAME=$FPATH/obsnat/${PRE}_obsnat ; fi
if [ $EXPNO = ,obs_act' ] ; then FNAME=$FPATH/obs_act/${PRE}_obs ; fi
if [ $EXPNO = ,obs_79-83' ] ; then FNAME=$FPATH/obs_79-83/${PRE}_obs ; fi
if [ $EXPNO = ,obs_81-85' ] ; then FNAME=$FPATH/obs_81-85/${PRE}_obs ; fi
if [ $EXPNO = ,obsnat_81-85' ] ; then FNAME=$FPATH/obsnat_81-85/${PRE}_obs ; fi
if [ $EXPNO = ,LEGca' ] ; then FNAME=$FPATH/LEGca/LEGca_${PRE}_1-1 ; fi
if [ $EXPNO = ,LEGar' ] ; then FNAME=$FPATH/LEGar/LEGar_${PRE}_1-1 ; fi
FELDELEMENT=$IZ,$IY
echo $FELDELEMENT
echo $FNAME
cat source.f | sed -e ,,s(file$IZ$IY($FNAME(,, > tmp
mv tmp source.f
#
IY=`expr $IY + 1`
#
done
#
IY=1
#
# ***************************************************************
#
IZ=`expr $IZ + 1`
done
#
# ************ place other parameters into source.f ***************
#
echo $AUTOSCALE
#
cat source.f | sed -e ,,s(TYPS(${TYPS}(,, > tmp
mv tmp source.f
cat source.f | sed -e ,,s(MAXWERT1($SANZ(,, > tmp
mv tmp source.f
IZ=`expr $IZ - 1`
cat source.f | sed -e ,,s(EXPANZ($EXPANZ(,, > tmp
mv tmp source.f
cat source.f | sed -e ,,s(A_SCALE($AUTOSCALE(,, > tmp
mv tmp source.f
#
# ************ place Y-Achsen Einheit in source.f
cat source.f | sed -e ,,s(unittext($UTEXT(,, > tmp
mv tmp source.f
#
# ************ place X-Achsen Einheit in source.f
cat source.f | sed -e ,,s(xunit($XTEXT(,, > tmp
mv tmp source.f
#
# ***************************************************************
#
# ********************** run source.f **********************
f77 source.f
a.out
rm a.out
rm -f file??
```

References

Birkett, C.M, and I.M. Mason, 1995
    A new global lakes database for a remote sensing program studying climatically
    sensitive large lakes
    J. Great Lakes Res. **21** (3), 307-318
Charbonneau, R., J.-P. Fortin and G. Morin, 1977
    The CEQEAU model: description and examples of its use in problems related to water
    resource management
    Hydrol. Sci. Bull. **22**, 193-202
CIA, 1992
    CIA World Data Bank II
    Public domain dataset
Coe, M.T., and G.B. Bonan, 1997
    Feedbacks between climate and surface water in Northern Africa during the middle
    holocene
    J. Geophys. Res., accepted for publication
J.G. Cogley, 1987, 1991, 1994
    GGHYDRO - Global Hydrographic Data, Release 2.1
    Department of Geography, Trent University, Peterborough, Ontario, Canada
Dooge, J.C.I., 1977
    Problems and methods of rainfall - runoff modelling
    Mathematical Models for Surface Water Hydrology, Proceedings of the Workshop held
    at the IBM Scientific Center, Pisa, Italy, John Wiley & Sons
Dümenil, L., and E. Todini, 1992
    A rainfall-runoff scheme for use in the Hamburg climate model
    Ed.: J.P. Kane: *Advances in Theoretical Hydrology - a Tribute to James Dooge*
    Elsevier Science Publishers, 129-157
Edwards, M.O., 1989
    Global gridded elevation and bathymetry (ETOPO5). Digital raster data on a 5-minute
    geographic (lat/long) 2160*4320 (centroid-registered) grid.
    NOAA National Geophysical Data Center, Boulder, Colorado
Foster, D.J., and R.D. Davy, 1988
    Global Snow Depth Climatology
    USAFETAC/TN-88/006, Scott Air Force Base, Illinois
Hagemann, S., 1998
    Entwicklung einer Parameterisierung des lateralen Abflusses für Landflächen auf der
    globalen Skala
    Max-Planck-Institute for Meteorology, Examensarbeit **52**, Hamburg, Germany
Hagemann, S., and L. Dümenil, 1998
    A parameterization of the lateral waterflow for the global scale
    Clim. Dyn., **14**, 17-31
Hagemann, S., and L. Dümenil, 1998b
    Comparison of two global wetlands datasets using a global hydrological discharge
    model.
    Submitted to *Hydrology and Earth System Sciences*
Legates, D.R., and C.J. Willmott, 1990a

Mean seasonal and spatial variability in gauge-corrected, global precipitation
Int. J. Climatol. 10, 111-127

Legates, D.R., und C.J. Willmott, 1990b
Mean seasonal and spatial variability in global surface air temperature
Theor. Appl. Climatol. 41, 11-21

Matthews, E., and I. Fung, 1987
Methane emissions from natural wetlands: Global distribution, area, and environmental characteristics of sources
Global Biogeochem. Cycles **1**, 61-86

Meeson, B.W., F.E. Corprew, J.M.P. McManus, D.M. Myers, J.W. Closs, K.-J. Sun, D.J. Sunday and P.J. Sellers, 1995
ISLSCP initiative I - global data sets for land-atmosphere models, 1987-1988
Volumes 1-5. Published on CD by NASA

Mitsch, W.J., and J.G. Gosselink, 1993
Wetlands. Second edition
Van Nostrand Reinhold, New York

Roeckner, E., K. Arpe, L. Bengtsson, M. Christoph, M. Claussen, L. Dümenil, M. Esch, M. Giorgetta, U. Schlese and U. Schulzweida, 1996
The atmospheric general circulation model ECHAM-4: model description and simulation of present-day climate
Max-Planck-Institute for Meteorology, Report **218**, Hamburg, Germany

Rudolf, B., H. Hauschild, W. Rüth und U. Schneider, 1996
Comparison of raingauge analyses, satellite-based precipitation estimates and forecast model results
Adv. Space. Res. 7, 53-62

Singh, V.P., 1988
Hydrologic systems vol. 1: Rainfall-runoff modeling
Prentice Hall

Walter, B.P., M. Heimann, R.D. Shannon and J.R. White, 1996
A process-based model to derive methane emissions from natural wetlands
Geophys. Res. Lett. **23**, 3731-3734