WaterGAPLite

Content

1 Introduction

To improve the global model WaterGAP3 a smaller, more flexible model is created called *WaterGAPLite* which uses Rstudio as interface to run model equations in c++. Because the model WaterGAP3 has a lot of model options that have their roots in previous dissertations and that are not useful in most cases the created model has less options that are confined to the options that are found to be most important. These characteristics lead also to the model name *WaterGAPLite*. The model settings are explained in more detail in chapter 3. To get a first impression of the potential of WaterGAPLite the simulation routine and the model options are explained briefly:

- **Climate** Input: EWEMBI (worldwide) or continental, basinwide climate data
- Calculation of **Potential Evapotranspiration**: Priestley-Taylor
- **Permafrost**: not calculated (avoid frost numbers)
- **Arid/Humid** areas: not calculated (avoid Köppen areas)
- **Interception**: is simulated with temporal changing interception storage size
- **Snow**: using approach form Verzano and sub-grid
- **Soil**: standard model equation based on HBV model is used
- **Sealed areas**: Assumption that 50 % of precipitation above sealed area contributes to fast (surface) runoff
- **Groundwater/Surface Water**: Splitting from daily runoff into surface and groundwater runoff is based on various information from grid cell – also this factor can be set manually for the whole basin
- **River Routing**: constant and variable flow velocity (s. Verzano) is possible
- **Reservoir**: option to simulate reservoirs as global lakes or with Hanasaki algorithm is possible
- **Water Use**: water use options are: on/off and temporal and/or spatial allocation
- **Gap Year**: on/off (with 29.02 or without 29.02)

WaterGAPLite can be driven by the original model input from WaterGAP3 which makes a model-inter comparison and therefore a model improvement based on findings from *WaterGAPLite* simulation more reliable. The usage of so-called correction factors is avoided so the water balance of all simulated basins is closed.

2 Simulation Preparation

To run WaterGAPLite there are some pre-requisites required. This are among others an actual R version (R >= 3.6.0) and Rstudio. As well as some R packages:

- raster,
- sp,
- sf,
- Rcpp (a c++ compiler may be needed),
- ncdf4,
- fasterize,
- rgeos,
- rgdal,
- binaryLogic,
- ggplot2,
- dismo,
- data.table.

Not all the packages may be needed to run WaterGAPLite. However, an installation may be recommendable to deal with model output.

2.1 Required folder structure

The modell WaterGAPLite requires a pre-defined model structure so that the model interface Rstudio can find all required model data.

- **source_code**
    - MainCode
    - SubModules
- **data**
    - basinInfo/[cont]
    - changingFiles/[cont]
    - calibration/[cont]
    - climate/global | continental | basin
    - routing/[cont]
    - waterUse/[cont]
- **output**/[cont]

**the subfolders of climate which is needed depends on the type of climate input*

2.2 Required Input

To run WaterGAPLite original model input from WaterGAP3 is needed. In the following the required information are briefly described. Because the model input stems from the original model WaterGAP3 most of the input is written in a special binary format called 'UNF-Files'.

Alternatively, the data can be prepared without the package tools and non-dependent on any UNF-Files. To do so, vectorized input data must be defined per basin and classes as defined in init.classes.r must be created. Then, the defined objects can be passed to basin.prepareRun to create the list to past to the runModel function.

Info: UNF-Files

To make dealing with binary files more transparent each file has a specific ending indicating how to read binaries into the system. These specific endings are shown in Table 2.1. Depending on the type there are maximal numbers that can be stored in this type.

Table 2.1: Information about ending

Name	Type	nBytes (Size)	maxValue*
.UNF0 .UUF0	Float	4	Signed: -128; 128 Unsigned: 0; 255
.UNF1 .UUF1	Integer	1	Signed: -128; 128 Unsigned: 0; 255
.UNF2 .UUF2	Integer	2	Signed: - −32.768; 32.767 Unsigned: 0; 65.535
.UNF4 .UUF4	Integer	4	Signed: - −2.147.483.648; 2.147.483.647 Unsigned: 0; 4.294.967.295

**see: https://de.wikipedia.org/wiki/Integer_(Datentyp)*

Also, it should be noted that multi-layer UNF-Files are built in a specific manner. First, to define how many layers exists in the binary file, there exists a number before the ending *.UNF. Inside the binary file (e.g. with n cells and k layers) the order is the following: cell1.layer1, cell1.layer2, ⋯, cell1.layerk, cell2.layer1, cell2.layer2, ⋯, cell2.layerk, ⋯, celln.layerk. To get a raster from the UNF-Files the information in GC and GR can be used to build a grid.

2.2.1 basinInfo

The folder *basinInfo* contains almost all information regarding the simulated basin. Because the original model input for WaterGAP3 persist of continentwide data there is only one file per continent needed. So, when simulating various basins in the same continent there is only one set of files needed.

2.2.1.1 General Information about Basin:

- G_ALTITUDE.UNF2
- G_AQ_FACTOR.UNF1
- G_ARID_HUMID.UNF2
- G_BATJES.UNF2
- G_ELEV_RANGE.26
- G_GAMMA_HBV.UNF0
- G_PERMAGLAC.UNF1
- G_SLOPE_CLASS.UNF1
- G_TEXTURE.UNF1
- GAREA.UNF0
- GBUILTUP.UNF0
- GC.UNF2
- GCRC.UNF4
- GLCT.UNF1
- GR.UNF2
- LAI_31.DAT
- LCT_31.DAT

2.2.1.2 Info to use global climate data:

- G_WG3_WG2WITH5MIN.UNF4
- G_WG3_WATCH.UNF4

2.2.1.3 Info about water bodies

- G_GLOLAK.UNF1
- G_GLOWET.UNF1
- G_LAKAREA.UNF4
- G_LOCLAK.UNF1
- G_LOCWET.UNF1
- G_RES_TYPE.UNF1
- G_RESAREA.UNF4

- G_STORAGE_CAPACITY.UNF0

Note: At the moment it is assumed that land cells are 100% non-ocean

2.2.2 changingFiles

At the moment these files are only read, but should change with calibration because they are products of the model itself (during 1st 'natural' calibration run) and needed for the reservoir management scheme and the variable flow velocity:

- G_MEAN_INFLOW.12.UNF0
- G_MEAN_INFLOW.UNF0
- G_START_MONTH.UNF1
- G_ALLOC_COEFF.20.UNF0

for reservoir algorithm

- G_BANKFULL.UNF0 (for variable flow velocity)

Some of this data is calculated in WaterGAP3 model intern (like G_start_month and ALLOC_COEFF I think) and some are calculated external ("post-calib"). For WaterGAPLite it is planned to calculate all of these files externally so data pre-processing and simulation are more strictly separated. When reservoir management scheme is omitted and flow velocity is hold constant these files can be ignored.

2.2.3 routing

- G_FLOW_ACC.UNF4
- G_OUTFLC.UNF4
- G_RIVER_LENGTH.UNF0
- G_RIVERSLOPE.UNF0
- G_ROUGHNESS.UNF0

2.2.4 climate

- GLONGWAVE_DOWN_[YEAR]_ [MONTH].31.UNF0
- GPREC _[YEAR]_ [MONTH].31.UNF0
- GSHORTWAVE_[YEAR]_ [MONTH].31.UNF0
- GTEMP_[YEAR]_ [MONTH].31.UNF0

2.2.5 waterUse

- G_NUs_7100.UNF0
- G_NETUSE_GW_HISTAREA_m3_[YEAR].12.UNF0*
- G_NETUSE_SW_HISTAREA_m3_[YEAR].12.UNF0*

Note: All data is based on the water use model

*only when water use is considered (prerequisite for reservoir algorithm)

2.2.6 calibration

- [grdc-number]_Q_Day.Cmd.txt from GRDC Data

3 Model Settings

Here, model settings are described. Because software is a lively project, model settings might change. To have a look at the current settings type into the R console:

```
?WaterGAPLite:: init.settings.
```

Important model information to run the model:

```
Settings = c(WaterUseType,
             WaterUseAllocationType,
             FlowType,
             GapYearType,
             Reservoir Options,
             SplittingFactor,
             CalculateLongwave)
```

In the following these model settings will be explained briefly.

3.1 WaterUseType

0: No water use is considered

1: water use without transport to cities is considered

2: water use including transport to cities is considered

3.2 WaterUseAllocationType

0: spatial and temporal distribution

1: only spatial distribution (neighbouring cell and next 20 downstream cells within basin)

2: only temporal distribution (delayed satisfaction is allowed, set to 0 on 01.01.)

3.3 FlowType

0: const. velocity [1 m/s]

1: variable flow velocity after Verzano

3.4 GapYearType

0: normal time period is used

1: 29.02. is not considered (to compare simulation results with WG3)

3.5 Reservoir Options

0: Hanasaki (for this setting water use needs to be turned on in settings)

1: reservoirs as global lakes

3.6 SplittingFactor

0: calculating splitting factor as defined in WG3

1: setting splitting factor with list (can be used for calibration purpose)

3.7 Calculating Longwave

0: Longwave information is used from LONGWAVE_DOWN (read in)

1: Longwave information is calculated based on shortwave downward radiation and daily Temperature

4 Model Equations

In the following, the integrated model equations of WaterGAPLite are explained. Hereby, the index ***cell*** stands for all cells that are part of the simulation grid. The index ***k*** stands for the subscale grid cells in every cell and ***i*** is the index for the simulation step where 1 belongs to the first simulation step. The called variables are specified in Chapter **Fehler! Verweisquelle konnte nicht gefunden werden.** in form of an overview table.

It should be noted beforehand that in general, a numerical solution of the differential equation is neglected in WaterGAPLite. In contrast, almost all evolving differential equations are solved step by step assuming that one flux (e g., incoming precipitation) is 'first' leaving or arriving at the storage and 'afterward' another flux (e.g., outgoing evaporation) is leaving or arriving and so on. At the moment, this stepwise approach is integrated in the state-of-the-art version of WaterGAP3. Its simplicity is because the usually monthly or annual model output is analyzed where the numerical uncertainty plays a minor role when the model simulations are realized on a daily time step. However, it should be considered that there might be an advantage of integrating more straightforward approaches to solve the differential equation in the future.

4.1 Climate input

After reading all the needed information the potential evapotranspiration is estimated for every day ***i*** and every ***cell***. In the following, it is explained how this estimation is integrated into WaterGAPLite. Also, the estimation of the net longwave radiation is explained briefly which is a model option that can be defined in the settings.

4.1.1 Estimating longwave radiation (optional)

The estimation of net longwave radiation is based on Kaspar (2004) which uses the following information

- T = Temperature [° C],
- R_S = Shortwave Downward Radiation [W/m^2],
- a_c and b_c = radiation coefficients (based on info about humid/arid region) [-],
- a_s and b_s = Angstrom coefficients [-],
- σ = Stefan-Boltzman constant (4.903·10^{-9}) [MJ/(m^{2}K^{4}d)]
- ext_{rad}= extraterrestrial radiation [mm/d]

The following equation are applied to estimate the net longwave radiation [mm/d].

$$R_{l,net,i} = \left(\left(a_c \cdot \left(\frac{\frac{0.0864}{lat_{heat,i}} \cdot R_{S.i}}{solar_{rad,possible,i}}\right) + b_c\right) \cdot \varepsilon_{net,i} \cdot \sigma \cdot T_i^4\right) / lat_{heat,i} \text{ [mm/d]} \tag{4.1}$$

With:

$$a_c = \begin{cases} 1.35 & , arid\ region \\ 1.00 & , humid\ region \end{cases}$$

$$b_c = \begin{cases} -0.35 & , arid\ region \\ 0.00 & , humid\ region \end{cases}$$

$$lat_{heat,i} = \begin{cases} 2.501 - 0.002361 \cdot T_i & , T_i > 0 \\ 2.835 & , otherwise \end{cases}$$

$$\varepsilon_{net,i} = -0.02 + 0.261 \cdot e^{-0.000777 \cdot T^2}$$

$$solar_{rad,possible,i} = (a_s + b_s) \cdot ext_{rad,i} = (0.25 + 0.5) \cdot ext_{rad,i}$$

Where the extraterrestrial radiation is estimated based on the time of the year for which the longwave radiation is estimated and information about the position of the cell:

- θ = position of the cell (latitude) [rad],
- DOY = Day of the year of calculated day (1st January ≙ 1).

The following equation are applied to estimate the extraterrestrial radiation [mm/d] which is described in detail in Duffie und Beckman (2013).

$$ext_{rad,i} = 15.392 \cdot dist_{es,i} \cdot (\sigma_{s,i} \cdot \sin(\theta\ ) \cdot \sin(decl) + \cos\ (\theta\ ) \cdot \cos(decl_i) \cdot \sin(\sigma_{s,i}) \qquad (4.2)$$

with:

$$dist_{es,i} = 1 + 0.033 \cdot \cos\left(DOY_i \cdot \frac{2\pi}{365}\right)$$

$$\begin{aligned} decl_i = \sin^{-1}(0.39795 \cdot \cos(0.2163108 \\ +2 \cdot \tan^{-1}(0.9671396 \cdot \tan(0.0086 \cdot (DOY_i - 186)))))\end{aligned}$$

$$\sigma_{s,i} = \pi\ - \cos^{-1}\left(\frac{\sin(\theta)\sin(decl_i)}{\cos\ (\theta) \cdot \cos\ (decl_i)}\right)$$

4.1.2 Potential Evapotranspiration

Simplified approach of **Priestley Taylor approach** (Priestley und Taylor 1972) that needs following information

- T = Temperature [° C],
- R_S = Shortwave Downward Radiation [W/m²],
- R_l = Longwave Downward Radiation [W/m²] or net Longwave Radiation [mm/d]
- α = Albedo/SnowAlbedo (based on landcover info, for water a value of 0.08 is used) [-]
- ε = Emissivity (based on Land cover info) [-],
- α_{PT} = alphaPT (based on info about humid/arid region) [-],
- γ_{pr} = Psychrometer constant is assumed to be 0,65 hPa/K after (Maniak 2016).

First, net radiation is calculated based on the measured shortwave and longwave downward radiation with information regarding albedo and emissivity and using the Stefan-Boltzmann constant (σ = 5.67×10^{-8} [1/(Wm^2 K^4)] = 4.903×10^{-9} [MJ/(m^{2}K^4 d)]). To read more about this equation see Shuttleworth (1993). When longwave downward radiation is not available, net longwave radiation can be estimated using the formulas in chapter 0. The transformation from W/m² into mm/d is done by estimating the latent heat with information of the temperature. In contrast to Shuttleworth (1993) emissivity is not used as a time-variable value dependent on humidity or temperature but as a constant dependent on the landcover. These values are literature-based and are obtained from Wilber et al. (1999) and correspondents to the values for the category '*Broadband'* which is the band-aveage emissivity of 12 different spectral bands using a Planck function for weighting the energy distribution.

$$R_{n,i} = (1-\alpha_i)R_{s,i} + R_{l,net,i} \text{ [W/m²]} \tag{4.3}$$

$$R_n = \frac{0.0864}{lat_{heat,i}} \cdot \left(R_{s,i} \cdot (1-\alpha_i)\right) + R_{l,net,i} \left[\frac{mm}{d}\right]$$

with: $\alpha_i = \begin{cases} \alpha_{normal} & , S_{sn_{cell,i}} \leq 3\ mm \\ \alpha_{snow} & , otherwise \end{cases}$

$lat_{heat,i} = \begin{cases} 2.501 - 0.002361 \cdot T_i & , T_i > 0 \\ 2.835 & , otherwise \end{cases}$

and: $R_{l,net,i} = \frac{0.0864}{lat_{heat,i}} \cdot R_{l,i} - \frac{\varepsilon \cdot \sigma \cdot T_i^4}{lat_{heat,i}}$, when longwave downward radiation is read in

Second, the gradient of the saturated vapor pressure is calculated after eq. 4.2.2 and 4.2.3 in Shuttleworth (1993).

$$\Delta_i = 4098 \cdot e_{s,i}/(237.3 + T_i)^2 \tag{4.4}$$

$$with\ e_{s,i} = 0.6108\ \cdot \exp\left(\frac{17.27 T_i}{237.3 + T_i}\right)$$

$$\Delta_i = 4098 \cdot \left(0.6108 \cdot \frac{\exp\left(\frac{17.27\ T_i}{T_i + 237.3}\right)}{(T_i + 237.3)^2} \right)$$

Then, the potential evapotranspiration can be estimated using the approach from Priestley-Taylor where the soil heat flux is neglected for simplification. The values for α_{P} are based on values in (Shuttleworth 1993) where α_{PT} is set to 1.74 in arid regions and 1.26 in humid regions. These regions are defined as input in G_ARID_HUMID.UNF2.

$$PET_i = \alpha_{PT} \cdot \frac{\Delta_i}{\Delta_i + \gamma_{pr}} \left(R_{n,i} - G\right). \tag{4.5}$$

$$PET_i = \alpha_{PT} \cdot \frac{\Delta_i}{\Delta_i + 0.65} \cdot \left(R_{n,i} - 0\right)$$

4.2 Water Balance

Water Balance is calculated for every cell in every (daily) time step. The following equations are applied to the land fraction of all cells and every day.

4.2.1 Interception storage

The interception storage is representing the canopy. The basic principles of this storage are shown in Figure 4.1. It is assumed that this is the first storage which falling precipitation needs to pass.

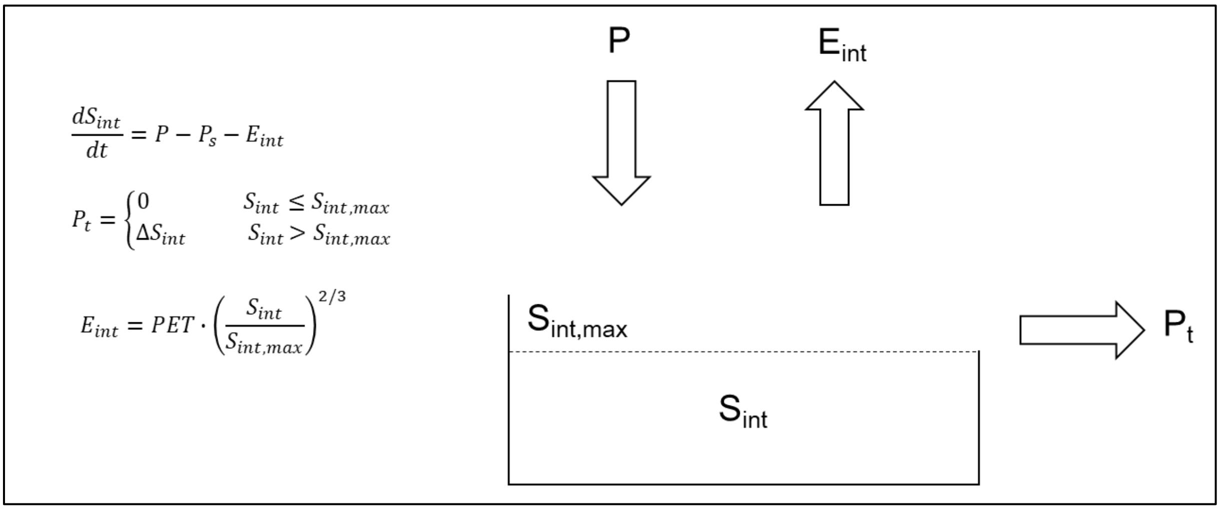
$\frac{dS_{int}}{dt} = P - P_s - E_{int}$
$P_t = \begin{cases} 0 & S_{int} \leq S_{int,max} \\ \Delta S_{int} & S_{int} > S_{int,max} \end{cases}$
$E_{int} = PET \cdot \left(\frac{S_{int}}{S_{int,max}}\right)^{2/3}$
P
E$_{int}$
S$_{int,max}$
S$_{int}$
P$_t$

Figure 4.1: Simplified principle of interception storage

Calculation of maximum storage size for interception

The **storage size of the interception storage** is time-variable to account for seasonal effects. The actual storage size is estimated based on information about temperature and precipitation (to account for seasonal effects) as well as on landcover information (to account for vegetation specific effects). For further information on how dailyLAI [m²/m²] is estimated see Section 4.2 in Müller Schmied et al. (2021). The dailyLAI value is then further multiplied by a constant which estimates the height of stored water on the canopy. This value is called maxCanopyStoragePerLAI and is estimated with a value of 0.3 mm (Deardorff 1978).

Balance of Interception storage

The differential equation for the interception storage is solved in a simple way. First, **precipitation** is added to the storage, overflow is calculated, and the storage level is updated again (s. eq. (4.6)-(4.7)).

$$P_{s,i} = -\min\left(S_{int,max,i} - (S_{int,i-1} + P_i); 0\right) \tag{4.6}$$

$$S_{int,i} = \min\left(S_{int,i-1} + P_i; S_{int,max,i}\right) \tag{4.7}$$

Second, **evaporation** from the storage is subtracted and the storage level is updated a second time (s. eq. (4.8)-(4.9)). When estimating the evaporation from the canopy reduction effects of PET are considered when the storage level is low. Negative storage level is not allowed so when the estimated evaporation is greater than the actual storage level it is reduced to the actual storage level. In this case, evaporation from the canopy is also reduced to ensure a closed water balance.

$$E_{int,i} = \min\left(PET_i \cdot \left(\frac{S_{int,i}}{S_{int,max,i}}\right)^{\frac{2}{3}} ; S_{int,i} \right) \tag{4.8}$$

$$S_{int,i} = \max\left(S_{int,i} - E_{int,i}; 0\right) \tag{4.9}$$

If the actual storage size is neglectable small there is **no evaporation** and all fallen precipitation is defined as throughfall. To ensure a close water balance the amount of water which is evaporated from the interception storage is subtracted from potential evapotranspiration which is furthermore used in the model.

4.2.2 Snow storage

The basic principles of snow storage are shown in Figure 4.2. In the following the model implementation and arising problems are explained in detail. The discussed problems are mainly caused by the problem that snow processes are happening in a low resolution whereas the model has a simulation resolution which is way above these snow processes and can’t simulate these processes in an accurate way without going into a subscale.

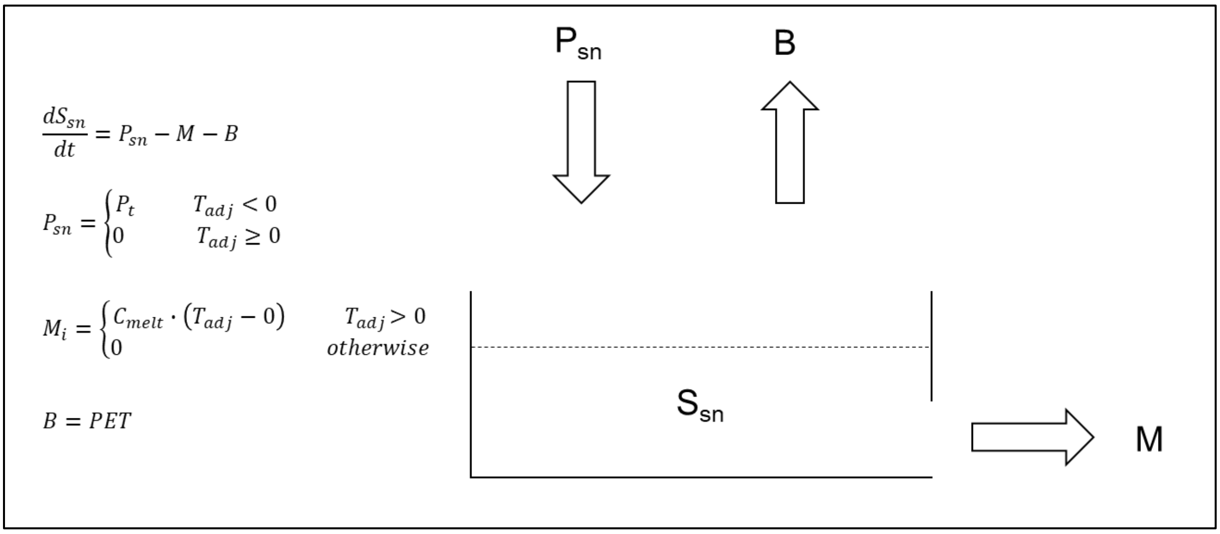
$\frac{dS_{sn}}{dt} = P_{sn} - M - B$
$P_{sn} = \begin{cases} P_t & T_{adj} < 0 \\ 0 & T_{adj} \geq 0 \end{cases}$
$M_i = \begin{cases} C_{melt} \cdot (T_{adj} - 0) & T_{adj} > 0 \\ 0 & otherwise \end{cases}$
$B = PET$
P_{sn}
B
S_{sn}
M

Figure 4.2: Simplified principle of snow storage

The snow routine is based on the sub-scale modeling scheme evolved by Verzano (2009) for WaterGAP2. The major difference between the described snow simulation in Verzano

(2009) and the applied snow simulation in WaterGAPLite is the spatial resolution. Whereas WaterGAP2 works with a 30min resolution and 3min resolution in a sub-scale for snow simulation WaterGAPLite works with a 5min resolution and a 1min resolution in a sub-scale for snow simulation as it is applied in WaterGAP3 (Eisner 2015). To simulate on a sub-scale, it is necessary to adjust the temperature. This is done by applying an elapse rate of 0.6° C/100m (Dunn und Colohan 1999). The mean altitude for the sub-scale, as well as the mean altitude of the 5min x 5min cell, are obtained by the basin information G_ELEV_RANGE.26.UNF2.

Due to the spatially coarse information about precipitation and the unperfect modeling of relevant snow processes in higher altitudes, e.g. glacier processes, snow cover may increase unlimited in very high altitudes. To account for that an arbitrary threshold of 1000mm = 1m of water equivalent in the snow cover is used to stop this unlimited increase by adjusting the sub-scale temperature.

In the following **the adjustment of sub-scale temperature** is shown in eq. (4.10).

$$T_{adj_{cell,k,i}} = \begin{cases} T_{cell,i} - \left(Elev_{cell,k} - Elev_{cell}\right) \cdot 0.006 & ,S_{sn_{cell,k,i-1}} > 1000mm \\ T_{cell,i} - \left(TresholdElev_{cell,i} - Elev_{cell}\right) \cdot 0.006 & ,S_{sn_{cell,k,i-1}} \leq 1000mm \end{cases} \quad (4.10)$$

where ThresholdElev is the minimal Elevation of the subscale grid where $S_{sn,k,cell}$ > 1000mm does occur for the specific day. Note that in the snow storage there is only the water equivalent calculated not the actual snow cover.

With the assumption that precipitation falls as snow when temperature is below 0° C, the **input of the snow storage** can be calculated as described in eq. (4.11)-(4.12). Furthermore, snow storage can be updated by the amount of fallen snow (s. eq. (4.13))..

$$B_{pot_{cell,k,i}} = \begin{cases} PET_{cell,i} - E_{c,cell,i} & ,T_{adj_{cell,k,i}} < 0 \\ 0 & ,T_{adj_{cell,k,i}} \geq 0 \end{cases} \quad (4.11)$$

$$P_{sn_{cell,k,i}} = \begin{cases} P_t & ,T_{adj_{cell,k,i}} < 0 \\ 0 & ,T_{adj_{cell,k,i}} \geq 0 \end{cases} \quad (4.12)$$

$$S_{sn_{cell,k,i}} = S_{sn_{cell,k,i-1}} + P_{sn_{cell,k,i}} \quad (4.13)$$

To account for **sublimation effects** the available potential evaporation (PET) is subtracted from the snow storage content and afterwards the storage level is updated again (s. eq. (4.14)-(4.15)). Here, the simplification is used that sublimation energy for the sub-scale grid is the same as the potential evapotranspiration energy from the whole grid.

$$B_{cell,k,i} = \min\,(B_{pot_{cell,k,i}};\ S_{sn_{k,cell,i}}) \quad (4.14)$$

$$S_{sn_{cell,k,i}} = S_{sn_{cell,k,i}} - \quad B_{cell,k,i} \quad (4.15)$$

Afterwards, a simple **degree day approach** is applied on the sub-scale to determine the melting water and snow water storage is updated again (s. eq. (4.16)-(4.17)).

$$M_{cell,k,i} = \min(C_{melt} \cdot (T_{adj_{cell,k,i}} - 0); S_{sn_{cell,k,i}}) \tag{4.16}$$

$$S_{sn_{cell,k,i}} = S_{sn_{cell,k,i}} - \quad M_{cell,k,i} \tag{4.17}$$

At the end of the snow routine, everything is aggregated to the simulation resolution of 5min s. eq. (4.18)-(4.21)). Differences in the area of the sub grid cells are neglected so a simple building of the mean is applied for the **spatial aggregation**.

$$M_{cell,i} = \frac{1}{25} \sum_{k=1}^{25} M_{cell,k,i} \tag{4.18}$$

$$B_{cell,i} = \frac{1}{25} \sum_{k=1}^{25} B_{cell,k,i} \tag{4.19}$$

$$S_{sn_{cell,i}} = \frac{1}{25} \sum_{k=1}^{25} S_{sn,cell,k,i} \tag{4.20}$$

$$P_{sn_{cell,i}} = \frac{1}{25} \sum_{k=1}^{25} P_{sn,cell,k,i} \tag{4.21}$$

4.2.3 Soil storage

The soil storage is the most powerful storage in the model. Within this storage the run-off generation process takes part. Moreover, the parameter γ is incorporated in this storage which is traditionally calibrated in the model versions WaterGAP2 and WaterGAP3. A simplified principle of the soil storage is shown in Figure 4.3.

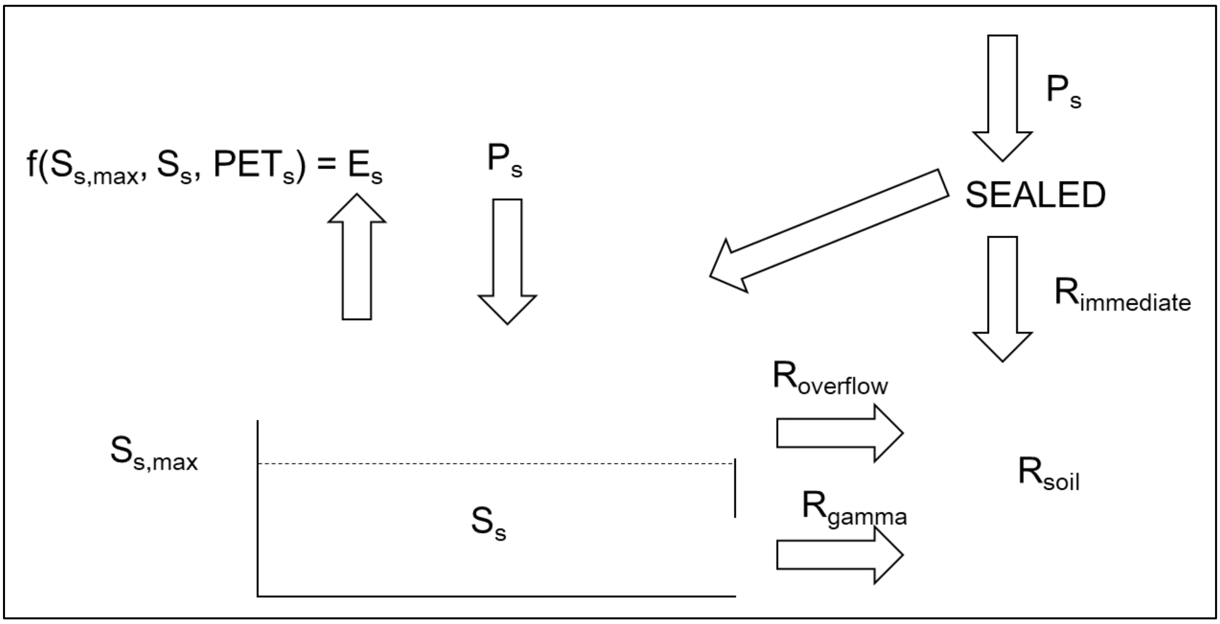

Figure 4.3: Simplified principle of soil storage

The input for the soil storage is based on the water balance of the interception storage and snow storage (s. eq. (4.22)) The **available energy for evaporation from soil** is defined as difference between available energy and already used energy (s. eq. (4.23)).

$$P_{s_{cell,i}} = (P_{t_{cell,i}} - P_{sn_{cell,i}}) + M_{cell,i} \tag{4.22}$$

$$PET_{s,cell,i} = (PET_{cell,i} - E_{int_{cell,i}} - B_{cell,i}) \tag{4.23}$$

For **sealed regions** it is assumed that 50 % of precipitation is immediate run-off and contributed therefore immediately to the total cell run-off (s. eq (4.24)). The other half is assumed to flow within the cell in unsealed regions and is therefore added to the water balance of the unsealed fraction of the cell (s. eq (4.25)). The fraction of sealed area is obtained by the basin information GBUILTUP.UNF0.

$$R_{immediate_{cell,i}} = 0.5 \cdot \; P_{s_{cell,i}} \cdot f_{builtUp_{cell}} \tag{4.24}$$

$$P_{s,unsealed_{cell,i}} = \; P_{s_{cell,i}} + 0.5 \cdot \; P_{s_{cell,i}} \cdot f_{builtUp} \tag{4.25}$$

To create a formula with the same reference area as it is used in all other formulas, the above formula can also be expressed as a corrected mean precipitation for the whole soil layer. Using this formula, the volume for the flux can be obtained by multiplication of the land fraction without considering the built-up fraction (s. eq (4.26)).

$$\begin{aligned} & P_{s,corr_{cell,i}} = \; \left(1 - f_{builtUp}\right) \cdot P_{s_{cell,i}} + 0.5 \cdot \; P_{s_{cell,i}} \cdot f_{builtUp} \\ \Leftrightarrow \quad & P_{s,corr_{cell,i}} = \; P_{s_{cell,i}} - f_{builtUp} \cdot \; P_{s_{cell,i}} + 0.5 \cdot \; P_{s_{cell,i}} \cdot f_{builtUp} \\ \Leftrightarrow \quad & P_{s,corr_{cell,i}} = \; P_{s_{cell,i}} - f_{builtUp} \cdot (\, P_{s_{cell,i}} - 0.5 \cdot \; P_{s_{cell,i}}) \\ \Leftrightarrow \quad & P_{s,corr_{cell,i}} = \; P_{s_{cell,i}} - f_{builtUp} \cdot 0.5 \cdot \; P_{s_{cell,i}} \\ \Leftrightarrow \quad & P_{s,corr_{cell,i}} = \; P_{s_{cell,i}} - R_{immediate_{cell,i}} \end{aligned} \tag{4.26}$$

Then, the soil water balance is calculated for the unsealed fraction of the grid cell. First, the daily run-off is estimated by an equation which has its origin in the HBV-96 model (Lindström et al. 1997) which is a result of a model revision from the HBV model (Bergström 1976). In this equation, which is shown in eq. (4.27), the **daily run-off** is a function of soil saturation, incoming precipitation and one parameter, called γ. The soil saturation is estimated by using a maximal soil storage which is derived by information of rooting depth that is a land cover dependent variable and soil information (G_BATJES.UNF2). The assumption behind this equation is that a higher soil moisture and more incoming precipitation leads to more run-off.

$$R_{gamma_{cell,i}} = \; P_{s,corr_{cell,i}} \cdot \left(\frac{S_{s_{cell,i}}}{S_{s,max_{cell}}}\right)^{\gamma} \tag{4.27}$$

Usually values between 0.1 and 5 are used for the parameter γ. Because the term right next to the incoming precipitation is limited to values in the span of 0 and 1 because $S_s \leq S_{s,max}$, run-off can be seen as certain percentage of incoming precipitation. To illustrate the meaning of the parameter γ there are several run-off/precipitation curves for different γ values shown in Figure 4.4.

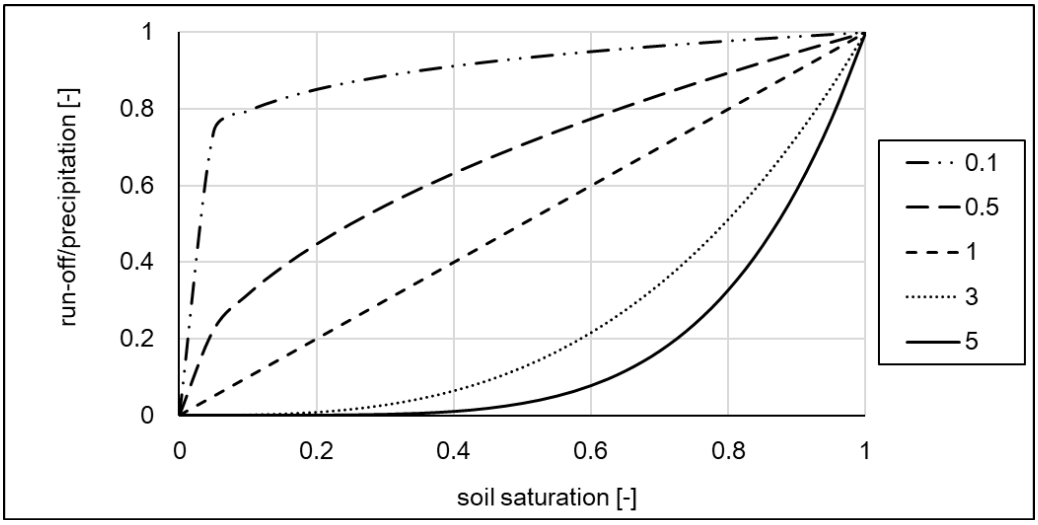

Figure 4.4: Role of calibration parameter γ in the run-off generation process

Second, the **evaporation** from soil is estimated (s. eq. (4.28)) by using the assumption that more soil moisture leads to more actual evaporation because of reduced soil moisture tension. Within this formula there is also a systematic difference between humid and arid regions assuming that in arid regions more soil water can evaporate whereas in humid regions there is a lower threshold of maximal evapotranspiration due to higher air humidity in those regions.

$$E_{s,i} = \min\left(PET_{s,i}, \left(PET_{s,max} - E_{int,i} - B_i\right) \cdot \frac{S_{s,i}}{S_{s,max}}\right) \tag{4.28}$$

with: $PET_{s,max} = \begin{cases} 10 & ,humid\ regions \\ 20 & ,arid\ regions \end{cases}$

In Figure 4.5 the systematic difference between arid and humid regions are demonstrated when calculating the actual evapotranspiration from soil with $PET_{s,i}$=10mm or $PET_{s,i}$=15mm. Whereas in humid regions there is no difference between the two graphs there is quite a huge difference for arid region when potential evapotranspiration changes from 10mm to 15mm.

a) b)

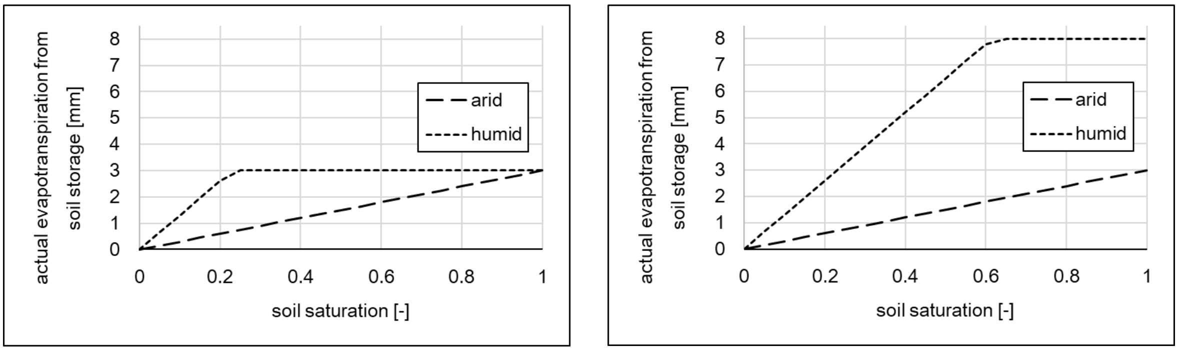

Figure 4.5: Actual evaporation from soil for two different cases in arid and humid region a) PET = 10mm & E_c + B = 7mm b) PET = 15mm & E_c + B = 7mm

After calculation all fluxes from the soil storage the soil storage is updated and corrected if soil moisture exceeds maximal soil moisture or is below zero because of two much actual evapotranspiration and run-off generation. To ensure a **closed water balance** the fluxes are also corrected when one of these two cases succeeds. To account for overflow from the soil layer if the soil moisture exceeds maximal soil moisture there is a new overflow flux defined. It is assumed that overflow and immediate run-off from sealed regions contribute to fast surface run-off (s. eq (4.29)-(4.32)).

$$S_{s_{cell,i}} = S_{s_{cell,i-1}} + P_{s,corr_{cell,i}} - R_{gamma,cell,i} - E_{s_{cell,i}} \tag{4.29}$$

$$S_{s,corr_{cell,i}} = \begin{cases} 0, & S_{s,cell,i} < 0 \\ S_{s,max_{cell}} & ,S_{s,cell,i} \geq S_{s,max_{cell}} \end{cases} \tag{4.30}$$

$$E_{s,corr_{cell,i}} = \begin{cases} E_{s_{cell,i}} + S_{s_{cell,i}}, & S_{s,cell,i} < 0 \\ E_{s_{cell,i}} & ,S_{s,cell,i} \geq S_{s,max_{cell}} \end{cases} \tag{4.31}$$

$$R_{overflow_{cell,i}} = \begin{cases} 0, & S_{s,cell,i} < 0 \\ (S_{s,cell,i} - S_{s,max_{cell}}) & ,S_{s,cell,i} \geq S_{s,max_{cell}} \end{cases} \tag{4.32}$$

4.2.4 Run-off splitting

After calculating the amount of run-off, the run-off needs to be separated into a fast and a slow component. The fast component can be interpreted as surface flow whereas the slow component can be seen as groundwater or slow interflow. The splitting is applied on the run-off component which is created by the HBV-formula without regarding immediate run-off from sealed regions and overflow from the soil storage. A heuristic approach is applied which uses information from permafrost (G_PERMAGLAC.UNF1), soil texture (G_TEXTURE.UNF1), slope (G_SLOPE_CLASS.UNF1), and aquifer information (G_AQ_FACTOR.UNF1) (Döll und Fiedler 2008). This information is used to determine a maximal groundwater recharge $R_{g,max}$, and a so-called groundwater factor f_g which is used to determine the fraction of generated run-off that contributes to groundwater inform of groundwater recharge. **Groundwater recharge** then is estimated in form of a minimum function between the maximal groundwater recharge rate and the determined groundwater recharge rate which is obtained by using the groundwater factor (s. eq (4.33)).

$$R_g = \min\,(R_{g,max}; f_g \cdot R_{soil}) \tag{4.33}$$

There is a special case regarding arid regions when precipitation is low and texture is medium/fine. If these conditions are met the groundwater recharge rate is set to zero. This special case avoids that too much groundwater is built in arid regions.

In the following, it will be explained briefly how the two variables $R_{g,max}$, and f_g are determined by using the information of permafrost, soil texture, slope class, and the aquifer information.

Maximal Groundwater Recharge Rate: $\mathbf{R_{g,max}}$

The maximal groundwater recharge rate is defined with the information of soil texture. Soil texture is defined with values in the range of 10 and 30 where 10 means coarse texture and 30 means fine texture. The value for $R_{g,max}$ is then defined by linear interpolation according to the shown relations in Table 4.1.

Table 4.1: Relation between texture information and maximal groundwater recharge rate

Texture [-]	10	15	20	25	30
$R_{g,max}$ [mm/d]	7.5	5.75	4.5	3.5	2.5

For rock and glaciers is it assumed that no ground water recharge occur and $R_{g,max}$ is set to zero.

Groundwater Splitting Factor: $\mathbf{f_g}$

The calculation of the groundwater factor uses all soil information mentioned above. Factors between zero and one are defined for the different information and all factors are multiplied at the end to define the factor f_g (s. eq. (4.34)).

$$f_g = f_r \cdot f_t \cdot f_h \cdot f_{pg} \in [0,1] \tag{4.34}$$

with: $f_h = \mathrm{G}_{\mathrm{AQ}_{\mathrm{FACTOR}}}/100 \in [0,1]$

$$f_{pg} = 1 - \mathrm{G}_{\mathrm{PERMAGLAC}}/100 \in [0,1]$$

$$f_t = Interpol(\mathrm{G}_{\mathrm{TEXTURE}}; 10,15,20,25,30; 1,0.975,0.95,0.825,0.7) \in [0,1]$$

$$f_r = Interpol(\mathrm{G}_{SLOPECLASS}\,; 10,20,30,40,50,60,70;\ 1,0.95,0.9,0.75,0.6,0.3,0.15) \in [0,1]$$

Following relations are established for the groundwater splitting:

- the more permafrost and glaciers, the lower the groundwater recharge rate,
- the coarser the soil texture, the higher the groundwater recharge rate,
- the stepper the slope, the lower the groundwater recharge rate.

Note: The relation for the aquifer factor is not that clear because metadata how this input is calculated has been lost.

After obtaining the groundwater recharge rate, the amount of **fast (surface) flow** can be calculated as following (s. eq. (4.35)).

$$R_{soil,cell,i} = R_{overflow,cell,i} + R_{immediate,cell,i} + (R_{gamma,cell,i} - R_{g,cell,i}) \qquad (4.35)$$

In the **model settings** there is the option to set the splitting factor. This means that $R_{g,max}$ as well as f_g are multiplicated by the defined factor. This approach is included in the model as a first step to avoid the usage of so much (uncertain) information.

4.2.5 Groundwater storage

The groundwater storage accounts for slow flow components in the model. For groundwater storage, there is no limitation of storage size. This means that there is no minimal nor maximal limitation. When the groundwater storage is negative (e.g. due to too much groundwater net abstraction) there occurs no outflow Q_g. The simplified principle for groundwater storage is shown in Figure 4.6.

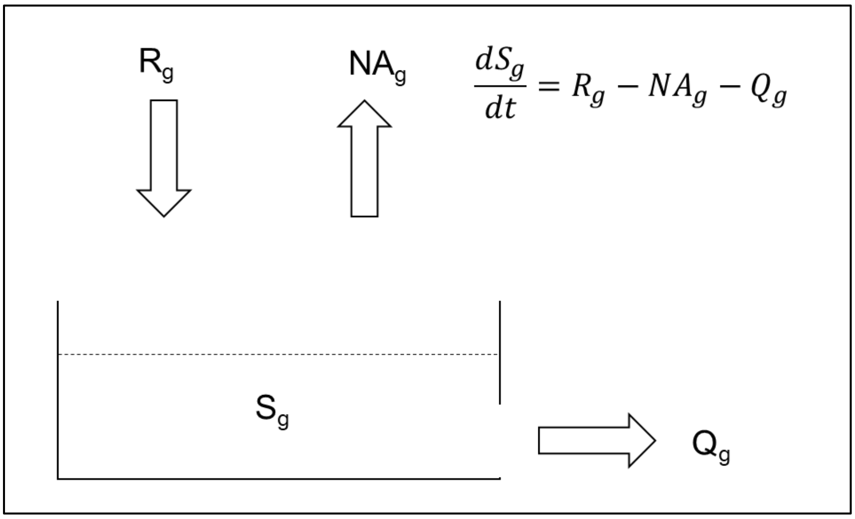

Figure 4.6: Simplified principle of groundwater storage

The differential equation of the **groundwater storage** is stepwise solved. First, the incoming groundwater recharge is added to the groundwater storage level (s. eq. (4.36)). Then, the outflow of the storage is defined by multiplying the storage level with a constant k_g=1/100d (s. eq. (4.37)). At last, the storage is updated again subtracting the defined outflow and possible net abstraction from groundwater (if human water use is considered and not equal zero, s. eq (4.38)).

$$S_{g_{cell,i}} = S_{G_{cell,i-1}} + R_{g_{cell,i}} \qquad (4.36)$$

$$Q_{g_{cell,i}} = \max\left(S_{g_{cell,i}} \cdot 0.01 \frac{1}{d}; 0 \right) \tag{4.37}$$

$$S_{g_{cell,i}} = S_{g_{cell,i}} - Q_{g_{cell,i}} - NA_{g,cell,i} \tag{4.38}$$

4.3 Routing

After the calculation of the water balance within the cell, the water needs to be routed through the simulation grid to the basin’s outlet. To account for different types of open water bodies there are five classes, namely

- local lakes,
- local wetlands,
- global lakes,
- reservoirs,
- global wetlands.

These open water bodies are passed by the water in the order which they are listed above. A draft of the routing process is shown in Figure 4.7.

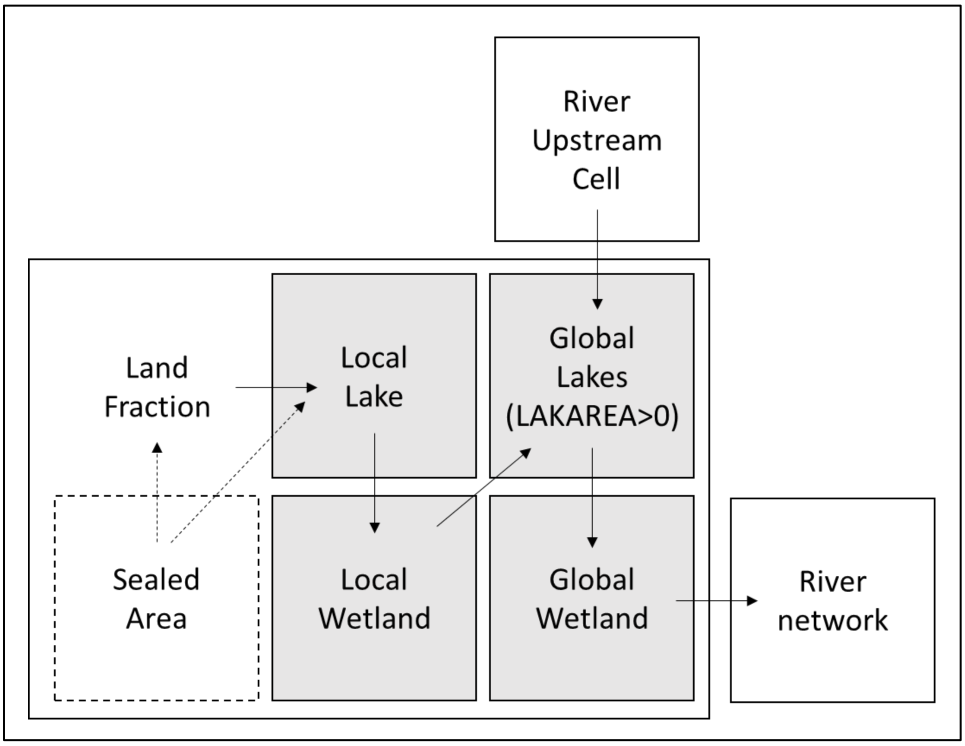

Figure 4.7: Draft of the routing concept of WaterGAPLite (reservoirs excluded)

First, the net cell run-off (groundwater and soil outflow) flows through the local lake and then through the local wetland. Afterward, water from upstream is added and the water is routed through the global lake and the global wetlands. The routing through the global lake is only performed when the lake area in LAKAREA is greater than 0. The information in GLOLAK doesn’t play a role in the routing. It serves as a placeholder, so the area is not accounted twice in the water balance (no operation at all is done for the cell fraction related to GLOLAK). By the way, the same is happening to reservoirs, where RESAREA is equivalent to LAKAREA. The information in GLOLAK doesn’t separate between reservoirs and global lakes to achieve savings regarding computational resources. Moreover, the water passes the global wetlands in the cell. In the last step, routing effects from the river network are estimated. If there are one or more water body types not included in the cell, the water flux will be passed directly to the next available water body type. When there is no water body type at all in the cell, the upstream water is directly added to the net cell run-off and only routing effects from the river network are accounted for.

The outflow from the cell (‘net cell run-off’) which is the water that drains through the water body-network is defined as sum of groundwater run-off Q_g (s. eq. (4.39)) and outflow from soil storage.

$$Q_{total} = R_{soil} \cdot (cell_{area} \cdot fraction_{land}) + Q_g \tag{4.39}$$

In the following, the routing process is explained in more detail. Also, it should be mentioned that the water balance of open water bodies is not described in chapter 4.2 because it is strongly connected with the routing process.

4.3.1 Local water bodies

The first water bodies that are passed by the net run-off from the cell are local water bodies. The principle of the routing through local water bodies is shown in Figure 4.8.

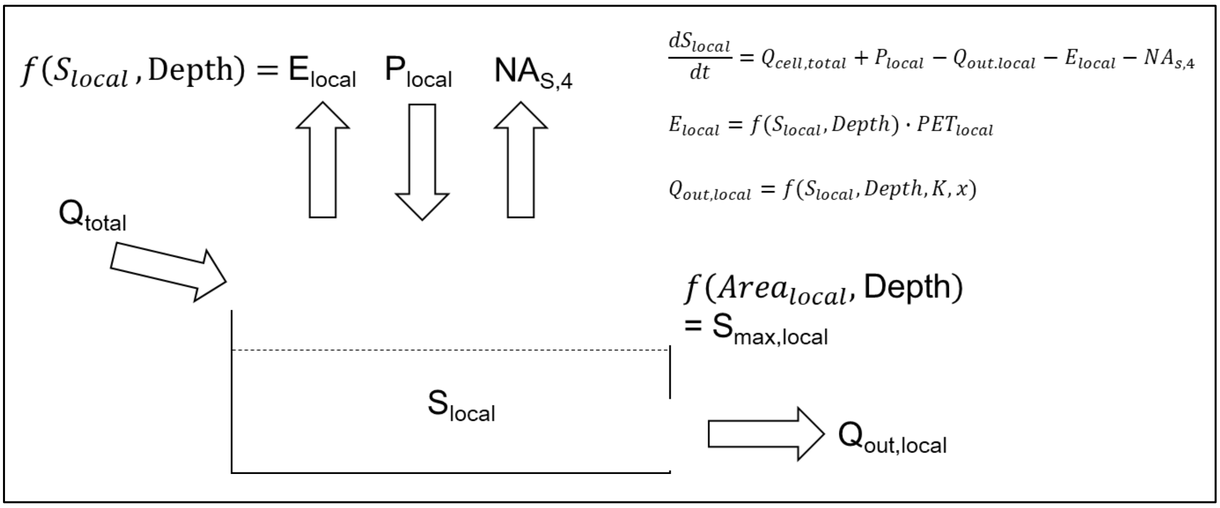
$f(S_{local}, \mathrm{Depth}) = E_{local}$
P_{local}
$NA_{S,4}$
$\frac{dS_{local}}{dt} = Q_{cell,total} + P_{local} - Q_{out.local} - E_{local} - NA_{s,4}$
$E_{local} = f(S_{local}, Depth) \cdot PET_{local}$
$Q_{out,local} = f(S_{local}, Depth, K, x)$
Q_{total}
$f(Area_{local}, \mathrm{Depth}) = S_{max,local}$
S_{local}
$Q_{out,local}$

Figure 4.8: Simplified principle of local water body – note that $NA_{S,4}$ is only present in the case of local lakes

The main equation which are applied to realize routing through local water bodies are eq. (4.40) and (4.41),where the values of K, x and Depth are dependent on whether the local water body is a wetland or a lake. The net abstraction from surface water ($NA_{S,4}$) is only present in the case of local lakes. For local wetlands there is no abstraction due to human water use.

$$\frac{dS_{local}}{dt} = Q_{total} + P_{local} - E_{local} - Q_{out,local} - NA_{s,4} \tag{4.40}$$

with: $P_{local} = P_{cell} \cdot (cell_{area} \cdot fraction_{localWB})$

$$Q_{out,local} = \frac{1}{K} \cdot S_{local} \cdot \left(\frac{S_{local}}{S_{max,local}}\right)^{x} \tag{4.41}$$

with: $S_{max,local} = fraction_{localWB} \cdot cell_{Area} \cdot Depth\ \ [km]$

The values for K, x and Depth are listed in the following table.

Table 4.2: Parameters for local water bodies

	lake	wetland
K	0.01 (=100d)	0.01 (=100d)
x	1.5	2.5
Depth	5 m	2 m

To calculate the water balance of the local water body the **evaporation** is calculated first as described in eq. (4.42). To account for different water levels, a reduction factor is applied when the actual storage level is smaller than the maximal storage capacity. When the water level equals zero there is no evaporation.

$$E_{local} = \begin{cases} PET_{water} \cdot (cell_{area} \cdot fraction_{localWB}), & S_{local} = S_{max,local} \\ red_{local} \cdot PET_{water} \cdot (cell_{area} \cdot fraction_{localWB}), & 0 > S_{local} < S_{max,local} \\ 0, & S_{local} = 0 \end{cases} \tag{4.42}$$

with: $$red_{local} = 1 - \left(\frac{S_{local} - S_{max}}{S_{max,local}}\right)^{3,32193}$$

After calculating the evaporation from the water body, the local water body storage is **updated** (s. eq. (4.43)).

$$S_{local} = S_{local} - E_{local} \tag{4.43}$$

Afterward, the **incoming inflow** from the surface flow within the cell and precipitation on lake surface is **routed** through the updated local water body and outflow from the water body is defined (s. eq. (4.44)).

$$Q_{out,local} = \frac{1}{K}(S_{local} + Q_{total} + P_{local}) \cdot \left(\frac{S_{local} + Q_{total} + P_{local}}{S_{max,local}}\right)^{X} \tag{4.44}$$

This **formula is numerical not stable** because the sum of inflows can be greater than $S_{max,local}$. Therefore, it is possible that more water is defined as outflow than water is in the lake storage. To avoid negative storage, it is checked if this case occurs and if so, outflow is reduced to be equal to the actual storage level.

After defining the outflow, the local water body storage is **updated** again (s. eq. (4.45)).

$$S_{local} = S_{local} - Q_{out,local} \tag{4.45}$$

When the local storage level is bigger than the defined maximal storage size **overflow** takes place (s. eq. (4.46)).

$$Q_{overflow,local} = \begin{cases} 0, & S_{local} \leq S_{max,local} \\ S_{local} - S_{max,local}\,, & S_{local} > S_{max,local} \end{cases} \tag{4.46}$$

Finally, the local water body storage and the outflow is **updated** again as described in eq. (4.47) and eq. (4.48).

$$S_{local} = S_{local} - Q_{overflow,local} \tag{4.47}$$

$$Q_{out,local} = Q_{out,local} + Q_{overflow,local} \tag{4.48}$$

When **human water** use is considered in the model settings, it is possible that net water abstraction from surface water is subtracted from local lakes as a last step from the remaining water level (s. eq. (4.49)). When there is not enough water stored in the lakes, lake storage is set to zero and the remained deficits is stored as ‘unsatisfied use’. This use may be satisfied by temporal or spatial distribution of the uses when defined so in the model settings. For more details see chapter 4.4.

$$S_{local} = S_{local} - NA_{S,4} \tag{4.49}$$

In the formula above the index 4 is used because the subtraction from local water bodies to satisfy water uses is only the fourth step: First, water is used from the river network. Second, if water use is not satisfied, water is taken from reservoirs (if present in the considered cell). Third, if water use is still not satisfied, water is taken from global lakes (if present in the considered cell). And as the last step, water is taken from local lakes to satisfy the uses (if present in the considered cell).

For clarification: When there is a **local wetland fraction as well as a local lake fraction** in the considered cell, the routed outflow from the local lake ($Q_{out,local,lake}$) is used as input for the local wetland (instead of Q_{total}). When there is **no local water body at all** in the cell, $Q_{out,local}$ corresponds simply to the built run-off in the cell (Q_{total}), so no routing effects are considered at all for local water bodies.

4.3.2 Global water bodies

Global water bodies are assumed to be part of the river network. Therefore, the routed flow in the river is used as input and not only the (routed) net-cell run-off. First, routing through global lakes is realized and then – when no reservoir is in the cell– routing through global wetlands is done. The two global water bodies types (lakes vs. wetlands) are treated differently. Both methods for routing are explained briefly in the following.

4.3.2.1 Global Lakes

Routing through global lakes is applied only for the outlet cell of a global lake (LAKAREA > 0). Inside this outlet cell routing effects and water balance for the whole water body is calculated. To avoid using the area of global lakes twice within the model– the fractions of global lakes (GLOLAK) within each cell is not considered when calculating the water balance inside each cell (s. Chapter 4.2). The **main equation** which are applied to realize routing through global lakes are shown in eq. (4.50) and (4.51) where the values of K is 0.0125 [1/d] (≙ 80 d of mean traveling time through the water body).

$$\frac{dS_{glolak}}{dt} = Q_{out,river,upstream} + P_{glolak} + Q_{out,local} - E_{glolak} - NA_{s,3} \tag{4.50}$$

$$Q_{out,glolak} = S_{glolak} - S_{glolak} \cdot e^{\frac{1}{K}} + Q_{in,glolak} - Q_{in,glolak} \cdot \frac{1}{K} \cdot \left(1 - e^{\frac{1}{K}}\right) \tag{4.51}$$

$$\Leftrightarrow Q_{out,glolak} = S_{glolak} \cdot (1 - e^{\frac{1}{K}}) + Q_{in,glolak} \cdot \left(1 - \frac{1}{K} \cdot \left(1 - e^{\frac{1}{K}}\right)\right)$$

The **inflow** for the global lakes is the inflow obtained from the upstream river network (which flows into the considered cell) and the precipitation which falls into the surface of the global water body (s. eq. (4.52)).

$$Q_{in,glolak} = Q_{out,river,upstream} + Q_{out,local} + P_{glolak} \tag{4.52}$$

with: $P_{glolak} = P_{cell} \cdot LAKAREA$

For huge global lakes like Lake Victoria, there arises a problem since the precipitation above the outlet is considered to be valid for the whole lake area (this is also valid for evaporation). This leads to 1) discrepancy in water balance when calculating precipitation sums for the whole continent based on input data and 2) over or underestimation of incoming precipitation on lakes and therefore possibly lower model efficiency.

There is also the concept of a **maximal storage capacity** for global water bodies. In contrary to the local water bodies this maximal storage capacity is not part of the routing equation. However, it is used to reduce evaporation and to calculate overflow when the actual storage volume exceeds the maximal storage capacity. For global lakes, the maximal storage capacity is calculated according to eq. (4.53).

$$S_{max,glolak} = LAKAREA \cdot DEPTH_{lake} = LAKAREA \cdot 0.005\ [km^3] \tag{4.53}$$

To calculate the water balance of the global lake, **evaporation** is calculated first. To account for different water levels a reduction factor is applied when the actual storage level is smaller than the maximal storage capacity. When there is a water level equal to zero in the local water body then there is no evaporation (s. eq. (4.54) -note that it is the same formula as for local lakes).

$$E_{glolak} = \begin{cases} PET_{water} \cdot LAKAREA, & S_{glolak} = S_{max,glolak} \\ red_{global} \cdot PET_{water} \cdot LAKAREA, & 0 > S_{glolak} < S_{max,glolak} \\ 0, & S_{glolak} = 0 \end{cases} \qquad (4.54)$$

with:
$$red_{global} = 1 - \left(\frac{S_{glolak} - S_{max,glolak}}{S_{max,glolak}}\right)^{3,32193}$$

With the estimated evaporation from the global lake the water body storage is **updated**.

$$S_{glolak} = S_{glolak} - E_{glolak} \qquad (4.55)$$

Afterward, the **inflow** is **routed** through the updated global lake, and outflow from the water body is defined (s, eq. (4.56)). The defined outflow is the inflow to reservoirs and if a reservoir does not exist in the considered outlet cell, it goes as inflow to global wetlands. When there also a global wetland does not exist in the considered cell it contributed directly to the river network.

$$Q_{out,glolak} = S_{glolak} - S_{glolak} \cdot e^{\frac{1}{K}} + Q_{in,glolak} - Q_{in,glolak} \cdot \frac{1}{K} \cdot \left(1 - e^{\frac{1}{K}}\right) \qquad (4.56)$$

$$\rightarrow \; Q_{out,glolak} = S_{glolak} \cdot (1 - e^{\frac{1}{K}}) + Q_{in,glolak} \cdot \left(1 - \frac{1}{K} \cdot \left(1 - e^{\frac{1}{K}}\right)\right)$$

Then the global lake storage is **updated** again as described in eq. (4.57).

$$S_{glolak} = S_{glolak} - Q_{out,glolak} \qquad (4.57)$$

When the local storage is bigger than the defined maximal storage size **overflow** takes place (s. eq. (4.58)).

$$Q_{overflow,glolak} = \begin{cases} 0, & S_{glolak} \leq S_{max,glolak} \\ S_{glolak} - S_{max,glolak}\,, & S_{glolak} > S_{max,glolak} \end{cases} \qquad (4.58)$$

Finally, the global water body storage and the outflow is **updated** again (s. eq. (4.59) and (4.60)).

$$S_{glolak} = S_{glolak} - Q_{overflow,glolak} \qquad (4.59)$$

$$Q_{out,glolak} = Q_{out,glolak} + Q_{overflow,glolak} \qquad (4.60)$$

When **human water** use is considered in the model settings, it is possible that net water abstraction from surface water is subtracted from global lakes (s. eq. (4.61)). When there is not enough water stored in the lake, lake storage is set to zero and the remained deficits is stored as ‘unsatisfied use’. This use may be satisfied by local lakes and temporal or spatial distribution of the uses when defined so in the model settings. For more details see chapter 4.4.

$$S_{glolak} = S_{glolak} - NA_{S,3} \tag{4.61}$$

When there is **no global lake** in the cell (LAKAREA = 0), no routing through global lake is simulated and $Q_{out,glolak}$ corresponds to $Q_{out,river,upstream}$ + $Q_{out,local}$.

4.3.2.2 Global Wetlands

The routing through global wetlands is like the routing through global lakes. The main difference between both routing algorithm is that the global lake algorithms is only applied for the outlet cell of a global lake whereas the global wetland algorithm is applied for every cell fraction where global wetland exists (like it is also done in the case of local water bodies). The inflow for the global wetland is defined as the outflow from the global lake plus the precipitation that falls above the global wetland (s. eq.

$$Q_{in,glowet} = Q_{out,glolak} + Q_{overflow,glolak} + P_{glowet} \tag{4.62}$$

with: $P_{glowet} = P_{cell} \cdot glowet_{cell} = P_{cell} \cdot fraction_{glowet} \cdot cell_{Area}$

The **maximal storage** capacity for global wetlands (s. eq. (4.63)) to account for evaporation reduction and overflow is calculated with a standard depth of 2 m (which is also used for local wetlands).

$$S_{max,glowet} = glowet_{cell} \cdot DEPTH_{wetland} = glowet_{cell} \cdot 0.002\ [km^3] \tag{4.63}$$

Then the formulas from global lakes (**evaporation**, **routing**, **overflow**) are applied analogous (s. eq. (4.64)-(4.70)). Human water use is not subtracted from global wetlands (neither from local wetlands).

$$E_{glowet} = \begin{cases} PET_{water} \cdot glowet_{cell}, & S_{glowet} = S_{max,glowet} \\ red_{global} \cdot PET_{water} \cdot glowet_{cell}, & 0 > S_{glolak} < S_{max,glowet} \\ 0, & S_{glowet} = 0 \end{cases} \tag{4.64}$$

with: $$red_{global} = 1 - \left(\frac{S_{glowet} - S_{max,glowet}}{S_{max,glowet}}\right)^{3,32193}$$

$$S_{glowet} = S_{glowet} - E_{glowet} \tag{4.65}$$

$$Q_{out,glowet} = S_{glowet} \cdot (1 - e^{\frac{1}{K}}) + Q_{in,glowet} \cdot \left(1 - \frac{1}{K} \cdot \left(1 - e^{\frac{1}{K}}\right)\right) \tag{4.66}$$

$$S_{glowet} = S_{glowet} - Q_{out,glowet} \tag{4.67}$$

$$Q_{overflow,glowet} = \begin{cases} 0, & S_{glowet} \leq S_{max,glowet} \\ S_{glowet} - S_{max,glowet}, & S_{glowet} > S_{max,glowet} \end{cases} \tag{4.68}$$

$$S_{glowet} = S_{glowet} - Q_{overflow,glowet} \tag{4.69}$$

$$Q_{out,glowet} = Q_{out,glowet} + Q_{overflow,glowet} \tag{4.70}$$

When there is **no global wetland** in the cell (GLOWET = 0), no routing through global wetland is simulated and $Q_{out,glowet}$ corresponds to $Q_{out,glolake}$.

4.3.3 Reservoir Algorithm

Reservoir management is quite difficult to estimate because every reservoir has its own management scheme and a lot information are needed to fulfill at least the known purposes of reservoir management (avoidance of low flow situation, protection against high flow) by an estimated management scheme. In the model settings there exists several options for reservoir simulation.

The most basic setting is to treat reservoirs **as global lakes** to simulate at least the retention effect of the water body. In the traditional calibration approach this is the first step of calibration where further information for the reservoir algorithm are produced by estimating e.g. mean inflow to the reservoir by assuming that reservoirs are non-regulated and only retentions effect from the water body are influencing the flow. Of course, this will never meet the reality - however, it is a way of getting at least the magnitude of inflow in the right dimension to asses simulated flow in high/low for further management operation schemes. When simulation reservoirs as global lakes the information in RESAREA is seen to be equal to the information in LAKAREA. This is done in the model by the following calculation in eq. (4.71) and (4.72).

$$LAKAREA = LAKAREA + RESAREA \tag{4.71}$$

$$RESAREA = RESAREA - RESAREA \tag{4.72}$$

The updated information is than used for a 'normal' model run where RESAREA equals zero in every cell so no reservoir scheme is applied. Instead the reservoir area is treated as additional global lake area. Because the information in GLOLAK refers to global lakes and reservoirs no further preparation is needed to treat reservoirs as global lakes.

4.3.3.1 Hanasaki

The **Hanasaki** algorithm for reservoir management is the first management option for reservoirs included in WaterGAP. It goes back to the management scheme introduced by Hanasaki et al. (2006). Because the reservoir management option also needs information about human water use, it should be noted that when reservoirs are simulated as reservoirs, human water use should be turned on in the model settings. The algorithm of Hanasaki needs the following information to estimate the reservoir management scheme:

- G_ALLOC_COEFF.20.UNF0,
- G_MEAN_INFLOW.UNF0,
- G_START_MONTH.UNF1,
- G_MEAN_INFLOW.12.UNF0,
- RESAREA.UNF,
- RES_TYPE.

The allocation coefficient is used to determine how much water is needed to satisfy water use by the reservoir for the next twenty downstream cells. With the information 'G_MEAN_INFLOW' and 'G_MEAN_INFLOW.12' the magnitude of the simulated discharge can be classified as high or low. The information 'G_START_MONTH' defines the beginning of the operational year. The reservoir area and the reservoir type are reservoir specific input data that do no not change with calibration. The information regarding water use that are needed to the reservoir algorithm are

- G_NUs_7100.UNF0,
- G_NETUSE_SW_HISTAREA_m3_[YEAR].12.UNF0,

where the first is the mean demand and the latter is used to define the actual water demand.

First, **precipitation** and **incoming flow** is added to the storage (s. eq.(4.73)).

$$S_{reservoir} = S_{reservoir} + P_{reservoir} + Q_{in,reservoir} \tag{4.73}$$

Second, **evaporation** from the reservoir area is estimated and the water level in the reservoir is updated as described in eq. (4.74) and (4.75).

$$E_{reservoir} = \begin{cases} PET(\alpha = 0.08), & S_{reservoir} = S_{max,reservoir} \\ red_{reservoir} \cdot PET((\alpha = 0.08), & 0 > S_{reservoir} < S_{max,reservoir} \\ 0, & S_{reservoir} = 0 \end{cases} \tag{4.74}$$

with:
$$red_{reservoir} = 1 - \left(\frac{S_{reservoir} - S_{max,reservoir}}{S_{max,reservoir}}\right)^{2.81383}$$

$$S_{reservoir} = S_{reservoir} - E_{reservoir} \tag{4.75}$$

It can be noted that eq. (4.74) is almost the same as the formula to estimate the evaporation from global lakes (s. eq. (4.54)). The only difference is the exponent which is slightly smaller in the case of the formula for the reservoir. To illustrate the meaning of the exponent the calculated reduction factor is show for a global lake and a reservoir with varying relative storage volume in Figure 4.9.

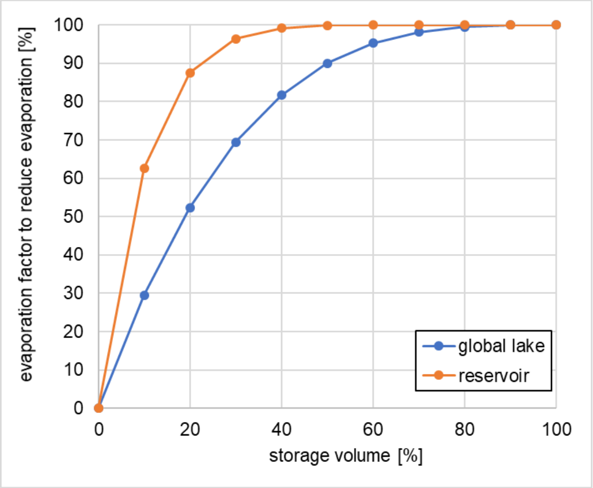

Figure 4.9: Meaning of the exponent for the reduction factor

It is appearing that evaporation from reservoirs is reduced less than evaporation from global lakes until a certain storage volume is reached. By this point (ca. 30 % storage volume) the evaporation is reduced more drastically for reservoirs. These differences in the curves are related to differences in (assumed) geometry of reservoirs and natural lakes. Whereas it is assumed that lakes have extensions that are flatter, for reservoirs it is assumed that they have a steeper extension. The steeper the extension, the more water volume is needed to change the area of the water body. So, steeper extensions lead to less reduction of evaporation because the size of the water body changes only slowly.

After calculating the water balance of the reservoir, the management scheme is applied. Each reservoir has it's one **ratio regarding its storage capacity and its mean inflow** which is called c_{ratio} in the following and which is calculated as defined in eq. (4.76).

$$c_{ratio} = G\_STORAGE\_CAPACITY/G\_MEAN\_INFLOW \cdot 12 \qquad (4.76)$$

The **maximal reservoir storage capacity** is defined as 85% of the information in G_STORAGE_CAPACITY (s. eq. (4.77)).

$$S_{max,reservoir} = G\_STORAGE\_CAPACITY \cdot 0.85 \qquad (4.77)$$

Of the beginning of the operational year the **release factor** is defined based on the actual storage filling when starting the operation year (s. eq. (4.78)).

$$K_{release} = \begin{cases} 0.1 & , \ S_{reservoir} < 0.1 \cdot S_{max,reservoir} \\ \dfrac{S_{reservoir}}{S_{max,reservoir}} & , \ else \end{cases} \qquad (4.78)$$

Then, depending on the reservoir type (non-irrigation vs. irrigation) a first **provisional release** is determined. For non-irrigation reservoirs, e.g. reservoirs that primary function is hydropower, the provisional release is determined as the mean yearly inflow to the reservoir. For irrigation reservoirs there are three cases used to define the provisional release.

These three cases depend on the mean inflow, the mean demand and the actual water demand and are shown in eq. (4.79).

$$prov_{rel} = \begin{cases} meanInflow, & meanDemand < 0 \text{ } or \text{ } actualDemand < 0^{*} \\ \frac{meanInflow}{2} \cdot \left(1 + \frac{actualDemand}{meanDemand}\right), & MeanDemand > 0.5 meanInflow \\ meanInflow + actualDemand - meanDemand, & else \end{cases} \tag{4.79}$$

with:

$$actualDemand = \sum_{j=1}^{20} Alloc_{Coeff,j} \cdot NA_{Sj}$$

**I think this is only the case when the used information in the water use model does not correspond to the used information in the HydroModel.*

Afterwards, the **release of the reservoir** is estimated according to the formulas in eq.

$$release = \begin{cases} K_{release} \cdot prov_{rel}, & c_{ratio} \geq 0.5 \\ \left(4c_{ratio}^2 \cdot K_{release} \cdot prov_{rel} + 1 - 4c_{ratio}^2\right) \cdot Q_{in,reservoir}, & else \end{cases} \tag{4.80}$$

When the actual reservoir storage is less then 10% of the value in G_STORAGE_CAPACITY the **outflow is reduced** by 90% (eq. (4.81)-(4.82)). This is applied to ensure that ecosystem demands are still served.

$$Q_{out,reservoir} = \begin{cases} release, & S_{reservoir} > 0.1 \cdot G\_STORAGE\_CAPACITY \\ 0.1 \cdot release, & else \end{cases} \tag{4.81}$$

$$S_{reservoir} = S_{reservoir} - Q_{out,reservoir} \tag{4.82}$$

At last, it is checked whether **overflow** occurs and the storage level and reservoir outflow is updated (s. eq. (4.83)-(4.85)

$$Q_{overflow,reservoir} = \begin{cases} S_{reservoir} - S_{max,reservoir}, & S_{reservoir} > S_{max,reservoir} \\ 0 & , else \end{cases} \tag{4.83}$$

$$S_{reservoir} = S_{reservoir} - Q_{overflow,reservoir} \tag{4.84}$$

$$Q_{out,reservoir} = Q_{out,reservoir} + Q_{overflow,reservoir} \tag{4.85}$$

The actual **abstraction** from the reservoir is done in the water use part of HydroModel (s. eq. (4.86)). There is the limitation that reservoir storage cannot be negative due to abstraction. Therefore, when the needed water amount exceeds the actual stored water in the, unsatisfied uses remains. These unsatisfied uses can be satisfied by global lakes or local lakes when present in the considered cell or by temporal or spatial distribution when set in the model options.

$$S_{reservoir} = S_{reservoir} - NA_{S,2} \tag{4.86}$$

4.3.3.2 Dynamic Optimization

Not implemented yet, originally based on van Beek et al. (2011)

4.3.4 River Channel

There are two options to simulate the river channel: constant and variable flow velocity. The usage of a constant flow velocity offers the opportunity to simulate the basin without the need for a-priori knowledge of a so-called bankfull flow. In this model option, it is assumed that the routing from one cell to another has always the same flow velocity neglecting effects of hysteresis or flooding. The option *variable flow velocity* tries to account for the effect that higher flow levels usually have higher flow velocities until flooding occurs.

When water use is considered in the model, water use volume ($\mathrm{NA}_{S,1}$) is subtracted from the river storage as a last step until the water use is satisfied or the river storage volume equals zero.

4.3.4.1 Constant flow velocity

The algorithm for the river channel routing is inspired by single linear storage where the flow velocity K determines the relation between inflow and outflow. The formula in eq. (4.87) is applied to simulate the routing in the river channel with a constant flow velocity.

$$S_{river,i+1} = S_{river,i} \cdot e^{-\frac{1}{h}} + Q_{in,river,i+1} \cdot \left(1 - e^{-\frac{1}{h}}\right) \tag{4.87}$$

$$Q_{out,river,i+1} = Q_{in,river,i+1} + S_{river,i} - S_{river,i+1}$$

with: $$h = \frac{riverLength}{K} = \frac{riverLength\,[km]}{86.4\,\left[\frac{km}{d}\right]}\,[d]$$

It should be noted that this is kind of a numerical approximation of the analytical solution of the single linear storage which would be the described formula in eq. (4.88).

$$Q_{out,river,i+1} = Q_{out,river,i} + (Q_{in,river,i+1} - Q_{out,river,i}) \cdot \left(1 - e^{-\frac{1}{h}}\right) \tag{4.88}$$

$$S_{river,i+1} = S_{river,i} + Q_{in,river,i+1} - Q_{out,river,i+1}$$

with $$h = \frac{riverLength}{K} = \frac{riverLength\,[km]}{86.4\,\left[\frac{km}{d}\right]}\,[d]$$

Note that Q is usually in mm/d and S in mm. For simplification the ·1d to get Q in mm is neglected in the formulas because it doesn't change the plain number.

4.3.4.2 Variable flow velocity

The variable flow velocity uses the same equation as the constant flow velocity to estimate the routing. However, the flow velocity which is by default estimated with 86.4 km/d is now approximated by formulas based on the river channel geometry and the actual flow (and the so-called bankfull flow from a previous calibration run). This allows increasing the flow velocity until the bankfull flow is reached by increasing simulated flow. The full explanation of the scheme can be found in Verzano et al. (2012). In eq. (4.89) the calculation is briefly shown by reducing the introduced formulas from Verzano et al. (2012) to the core relations which are used. The base for the formula is the Manning-Strickler formula.

$$K = \frac{1}{n} \cdot HR^{\frac{2}{3}} \cdot \sqrt{s} = \frac{1}{n} \cdot \sqrt{s} \cdot \left(\frac{3.7831Q^{0.898} + 5.846448Q^{0.682}}{2.710Q^{0.557} + 1.396\sqrt{5}Q^{0.341}} \right) \tag{4.89}$$

with: $\frac{1}{n}$: River Roughness information (read into model)

s: River Slope information (read into model)

HR: Hydraulic Radius (function of simulated Q)

The hydraulic radius increases with the increase of the simulated discharge and until the simulated discharge is equal to the estimated bankfull flow. When this point is reached, HR is set to the constant (maximal) level until the simulated discharge is again below the bankfull flow. Also, it should be mentioned that the calculated flow velocity can’t be less than 0.00001 km/d.

The River Roughness depends on a variety of information like mountain/non-mountain or urban/rural.

The river slope is calculated in a pre-processing step by considering the maximal vertical difference between the 25 sub-grids of a 5min cell and the belonging horizontal distance (a meandering ratio is also applied to consider the slope being steeper when the river is less curvy).

The Bankfull Flow is estimated by a a-priori model calibration where reservoirs being treated as global lakes and a constant flow velocity is used. After simulating ~30a a partial duration series is determined and the Flow for T=0.92 (equal to 1.5a in an annual maximal series) is calculated using logarithmic or a normal Pearson-III-distribution.

It should be noted that this approach has the benefit that more flow means faster flow – but it has a crucial weakness because. although the flow velocity is variable, flooding effects are neglected.

4.4 Water Use

In some regions in the world human water use play a key role in the water cycle. Therefore, the global hydrological model has model options to consider these effects in the model. The estimation of net abstraction – which can be negative or positive – is based on calculation of the water use model from the model framework WaterGAP3. In WaterGAPLite this information is seen as normal model input, so estimation methods are beyond the scope of this manual. For further information see for example Alcamo et al. (2003) and Vassolo und Döll (2005).

4.4.1 Abstraction from groundwater

Implemented but not described yet.

4.4.2 Abstraction from surface water

Implemented but not described yet.

4.4.2.1 Temporal allocation

Implemented but not described yet.

4.4.2.2 Spatial allocation

Implemented but not described yet.

5 Model Output

At the moment, there is no option to write model output automatically. However, all available model output is saved in the R environment as a returned list when executing the model run. Parts of the list can be saved or written out with a self-defined subroutine in R. Also, the complete list can be stored with:

```
        save(modelOutputList, file = “my_data.RData”).
```

Even the whole workspace, including the modelling results can be saved with the R command:

```
        save.image(file = “my_work_space.RData”)
```

Both save options can be load again into the workspace with

```
        load(“my_data.RData”) / load(“my_work_space.RData”).
```

In the following the elements of the list are described briefly, where a point between two word means that the word after the point is an element of the higher-ordered list of the first word, e.g., with

```
        BasinOutput.daily.Storages.SoilContent
```

Someone can access from the model output (‘BasinOutput’) the land fraction of the model (‘daily’); specifically, the storages of all land module components (‘Storages’); and in this case the soil storages (‘SoilContent’). In R itself the points needs to be displaced by the ‘$’ symbol.

6 Executing the model code

To run the model, it is only needed to open Rstudio with R>3.6 and type the following commands.

6.1 How to simply run the model

Preparation

```
library(WaterGAPLite)
```

Usually afterwards the model settings to use as well as the period to simulate are defined:

```
Settings = c(0, # WaterUse --> 0=off, 1=on 2=on (w transport to cities)
             0, # WaterUseAllocation
             0, # flowVelocity --> 0=const, 1=variable
             0, # GapYear --> 0=With 29.02, 1=Without 29.02
             1, # reservoirType --> 0: hanasaki, 1: global lakes
             0  # splitting factor → 0: calculating 1: setting
             1  # Calculting longwave → 0: yes, 1: no)

start = "01.01.1996" #%d.%m.%Y –always full month is simulated
end   = "31.12.2011" #%d.%m.%Y –always full month is simulated
```

At last, usually the base folder where the folder structure of WGL lies is defined:

```
base <- "U:/WaterGAPlite"
```

Initializing Basin and Climate data

To initialize a basin, it is recommendable to use always the same procedure:

(1) Time independent information of the basin is loaded:

```
LOBITH <- init.model(grdc_number, lat, long, cont, base)
```

(2) Time dependent information of the model is loaded (climate & water use)

```
LOBITH.climate <- init.climate(LOBITH, start, end, force2read=F)
LOBITH.waterUse <- init.waterUse(LOBITH, start, end, WaterUse_Setting)
```

(3) Afterwards a list is prepared to pass to c++ to simulate the basin with the prepared information:

```
LOBITH.model <- basin.prepareRun(LOBITH,
                                 LOBITH.climate,
                                 LOBITH.waterUse)
```

Simulating the basin

To simulate the basin all prepared information is passed to the c++ function of the model:

```
LOBITH.output <- runModel(LOBITH.model[["SimPeriod"]], LOBITH.model, Settings)
```

The model function returns a huge list with all information about fluxes and storages that are simulated within the basin.

Literature

Alcamo, J.; Döll, P.; Henrichs, T.; Kaspar, F.; Lehner, B.; Rösch, T.; Siebert, S. (2003): Development and testing of the WaterGAP 2 global model of water use and availability. In: *Hydrological Sciences Journal* 48 (3), S. 317–337. DOI: 10.1623/hysj.48.3.317.45290.

Bergström, S. (1976): Deveplopment and application of a conceptual runoff model for scandinavian catchments. Hg. v. SMHI. Norrköping (Report No. RHO 7).

Deardorff, J. W. (1978): Efficient prediction of ground surface temperature and moisture, with inclusion of a layer of vegetation. In: *Journal of Geophysical Research* 83, S. 1889–1903. Online verfügbar unter https://doi.org/10.1029/JC083iC04p01889.

Döll, P.; Fiedler, K. (2008): Global-scale modeling of groundwater recharge. In: *Hydrology and Earth System Science* (12), S. 863–885.

Duffie, John A.; Beckman, William A. (2013): Solar engineering of thermal processes John A. Duffie, William A. Beckman. 4th ed. Hoboken: John Wiley.

Dunn, S. M.; Colohan, R. J. E. (1999): Developing the snow component of a distributed hydrological model: a step-wise approach based on multi-objective analysis. In: *Journal of Hydrology* 223, S. 1–16. Online verfügbar unter https://doi.org/10.1016/S0022-1694(99)00095-5.

Eisner, S. (2015): Comprehensive Evaluation of the WaterGAP3 Modell across Climatic, Physiographic, and Anthropogenic Gradients. diss. University of Kassel, Kassel.

Hanasaki, Naota; Kanae, Shinjiro; Oki, Taikan (2006): A reservoir operation scheme for global river routing models. In: *Journal of Hydrology* 327 (1-2), S. 22–41. DOI: 10.1016/j.jhydrol.2005.11.011.

Kaspar, Frank (2004): Entwicklung und Unsicherheitsanalyse eines globalen hydrologischen Modells. Zugl.: Kassel, Univ., Diss, 2004. Kassel: Kassel Univ. Press. Online verfügbar unter http://d-nb.info/971852022.

Lindström, G.; Johansson, B.; Persson, M.; Gardelin, M.; Bergström, S. (1997): Development and test of the distributed HBV-96 hydrological model. In: *Journal of Hydrology* (201), S. 272–288.

Maniak, U. (2016): Hydrologie Und Wasserwirtschaft. Eine Einführung für Ingenieure. 7. Aufl. Berlin: Springer Vieweg.

Müller Schmied, Hannes; Cáceres, Denise; Eisner, Stephanie; Flörke, Martina; Herbert, Claudia; Niemann, Christoph et al. (2021): The global water resources and use model WaterGAP v2.2d: model description and evaluation. In: *Geosci. Model Dev.* 14 (2), S. 1037–1079. DOI: 10.5194/gmd-14-1037-2021.

Priestley, C. H. B.; Taylor, R. J. (1972): On the Assessment of Surface Heat Flux and Evaporation Using Large-Scale Parameters. Hg. v. Monthly Weather Review (100, 2).

Schulze, K.; Döll, P. (2003): Neue Ansätze zur Modellierung von Schneeakkumulation und -schmelze im globalen Wassermodell WaterGAP. In: R. Ludwig, D. Reichert und W. Mauser (Hg.): Tagungsband zum 7. Workshop zur großskaligen Modellierung in der Hydrologie. Kassel: Kassel University Press, S. 145–154.

Shuttleworth, J. W. (1993): Evaporation. In: D. R. Maidment (Hg.): Handbook of Hydrology. New York: McGraw-Hill, 4.01-4.53.

van Beek, L. P. H.; Wada, Yoshihide; Bierkens, Marc F. P. (2011): Global monthly water stress: 1. Water balance and water availability. In: *Water Resour. Res.* 47 (7). DOI: 10.1029/2010WR009791.

Vassolo, Sara; Döll, Petra (2005): Global-scale gridded estimates of thermoelectric power and manufacturing water use. In: *Water Resour. Res.* 41 (4). DOI: 10.1029/2004WR003360.

Verzano, K. (2009): Climate Change Impacts on Flood Related Hydrological Processes: Further Development and Application of a Global Scale Hydrological Model. diss. University of Kassel, Kassel.

Verzano, Kerstin; Bärlund, Ilona; Flörke, Martina; Lehner, Bernhard; Kynast, Ellen; Voß, Frank; Alcamo, Joseph (2012): Modeling variable river flow velocity on continental scale: Current situation and climate change impacts in Europe. In: *Journal of Hydrology* 424-425, S. 238–251. DOI: 10.1016/j.jhydrol.2012.01.005.

Wilber, A. C.; Kratz, D. P.; Gupta, S. K. (1999): Surface Emissivity Maps for Use in Satellite Retrievals of Longwave Radiation. In: *NASA / TP-1999-209362*.