

---

# HYPE model description

The document describes how HYPE model the flow and transformation of water, nutrients and organic carbon in soil, lakes and rivers. This includes the effect of irrigation, point sources, floodplains and aquifers.

## Contents:

1. [Processes above ground](#)
2. [Land routines](#)
3. [Rivers and lakes](#)
4. [Nitrogen and phosphorus in land routines](#)
5. [Nitrogen and phosphorus in rivers and lakes](#)
6. [Water management](#)
7. [Deep processes](#)
8. [Organic carbon](#)
9. [Tracers](#)
10. [Sediment](#)
11. [Silica](#)

# Processes above ground

## Temperature and precipitation

HYPE has many possibilities to adjust the precipitation and temperature given in the Pobs.txt and Tobs.txt files. Adjustment is done because the supplied data can be of different reference height or (known to be) wrong in some areas, or because data vary over the subbasin but the input is given as one value, and or to compensate for non modelled processes.

### Temperature adjustments

Subbasin air temperatures ( $T_i$ ) in the input file are normally assumed to be for the subbasin's average elevation (*elev*). The same adjustments are made for maximum and minimum temperatures if available.

(1) If data with other reference height is used it is possible to adjust for that. The subbasin temperature can be adjusted depending on subbasin elevation with general parameter *tcelevadd* ( $^{\circ} / 100m$ ). This adjustment assumes observations are located at sea level.

(2) Alternatively it can be adjusted based on the difference between subbasin elevation and temperature observation elevation by general parameter *tcobselev* ( $^{\circ} / 100m$ ). The temperature observation elevation ( $elev_{obs}$ ) then need to be supplied to the model.

(3) It can be adjusted based on month with the monthly dependent parameter *monthlapse* ( $^{\circ} / 100m$ ). These three subbasin elevation adjustments should not be used together. Subbasin temperature can be adjusted equally over all subbasins within a region with the parameter region dependent parameter *tempcorr*.

$$T_{gc} = T_i + tempcorr - \frac{tcelevadd \times elev}{100} - \frac{tcobselev \times (elev - elev_{obs})}{100} - \frac{monthlapse \times elev}{100}$$

The temperature can also be adjusted for each class depending on their deviation from the subbasin average elevation ( $\Delta h$ ). The class-dependent temperature (*T*) is calculated using the parameter *tcalt*. The temperature lapse rate often has a value of 0.6 ( $^{\circ} / 100m$ ).

$$T = T_{gc} - \frac{tcalt \times \Delta h}{100}$$

### Precipitation adjustments

Subbasin input precipitation ( $P_i$ ) can be adjusted equally over all subbasins with the general parameter *pcaddg* or for some subbasins with the parameter region dependent parameter *preccorr*. Additionally it is possible to adjust precipitation for undercatch with different parameters (*pcurain*, *pcusnow*) depending on if the precipitation falls as snow or rain.

Subbasin precipitation ( $P_{gc}$ ) is for subbasin average elevation ( $elev$ ), but can be adjusted for elevation variations within the subbasin. The precipitation of a class ( $P$ ) is adjusted for classes where the class average elevation is greater than a threshold (general model parameter  $pcelevth$ ). The adjustment is determined by a general parameter ( $pcelevadd$ ) that is the correction per 100m. The class elevation adjustment can alternatively be determined from the basin standard deviation of elevation ( $elev_{std}$ ) and a parameter  $pcelevstd$ . The class height adjustment is limited by a general parameter  $pcelevmax$ . The precipitation of a class can additionally be adjusted with land-use dependent parameter  $pcluse$ , e.g. for interception evaporation.

$$P_{gc} = P_i \times (1 + pcaddg) \times (1 + preccorr) \times \left( 1 + \left( pc_{rain} \times (1 - snfrac) + pc_{snow} \times snfrac \right) \right)$$

$$pc_{height} = \begin{cases} 0 & elev + \Delta h < pcelevth \\ MIN \left( \frac{elev + \Delta h - pcelevth}{100} \times pcelevadd + \frac{elev_{std}}{100} \times pcelevstd, pcelevmax \right) & else \end{cases}$$

$$P = P_{gc} \times (1 + pc_{height}) \times (1 - pcluse)$$

Where  $\Delta h$  is a class's elevation deviation from the subbasin average elevation and  $snfrac$  is the average fraction of precipitation that falls as snow calculated from subbasin temperature ( $T_i$ ) and temperature thresholds (see equation below), or from input.

## Rainfall and snowfall separation

The rain/snow fraction of precipitation is calculated based on temperature or given as an input time series. Different temperatures can be used in the equation, i.e. basin average or class temperature. When the air temperature ( $T$ ) is around the threshold temperature for mixed precipitation (land-use dependent parameter  $ttmp$  plus general parameter  $ttpd$ ) both rain and snow. The interval for mixed precipitation is given by the parameter  $ttpi$ . For temperature below threshold minus  $ttpi$ , the precipitation is assumed to be in solid form only and is added to the snowpack. If the air temperature is greater than the threshold temperature plus  $ttpi$ , the precipitation is assumed to be solely in liquid form. For intermediate temperatures, the precipitation is assumed to be a mixture of liquid and solid forms i.e. as both rain and snow. The proportion ( $a_{rain}$ ) of precipitation that falls as rain depends linearly on the temperature.

$$a_{rain} = \begin{cases} 0 & T < ttmp + ttpd - ttpi \\ 1 & T > ttmp + ttpd + ttpi \\ \frac{\left( T - (ttmp + ttpd - ttpi) \right)}{(2 \times ttpi)} & \text{else} \end{cases}$$

Alternatively snowfall fraction (*sfrac*) may be read from input file and  $a_{rain}$  calculated as:

$$a_{rain} = 1 - sfrac$$

The amount of rainfall and snowfall is calculated from the corrected precipitation (*P*).

$$rainfall = P \times a_{rain} \quad snowfall = P \times (1 - a_{rain})$$

## Snowfall distribution

The optional snowfall distribution models redistribute the snowfall of a catchment among the classes within a subbasin depending on wind. The model uses Winstral coefficients. For each subbasin and class and wind direction a Winstral coefficient (*WSF*) is given as input data to the model. Wind direction is calculated from forcing data of westerly (U) and southerly (V) wind. The wind direction is translated to the closest quadrant or octant, and the Winstral coefficient of that direction is used.

The Winstral coefficient is, with some scaling parameters, used to calculate the relative snowfall for each class this time step. There are two models; one that use a linear function (Gustafsson et al, 2015) and one that is log-linear (Clemenzi et al., 2022):

$$relative_{snowfall} = 1 + WSFscale \times WSFluse \times (WSF + WSFbias)$$

$$relative_{snowfall} = 10^{\left( WSFscale \times WSFluse \times (WSF + WSFbias) \right)}$$

*WSFscale* and *WSFbias* are general parameter, *WSFluse* is a scaling parameter depending on land-use. The relative snowfall is limited to the interval  $[1 - sfdmax, 1 + sfdmax]$  for the linear model and  $[0, sfdmax]$  for the log-linear. *sfdmax* is a general parameter. The relative snowfall is normalized within each subbasin so that the subbasin mean snowfall is preserved. The snowfall distribution is applied only if snow is falling for a fraction of the subbasin area that is larger than a limit (*limit*).

## References

Clemenzi et al., 2022, submitted. Gustafsson et al., 2015

## Links to file reference

Section	Symbol	Parameter/Data	File
Temperature adjustments	$T_i$		<a href="#">Tobs.txt</a>
	<i>elev</i>	<i>elev_mean</i>	<a href="#">GeoData.txt</a>
	$\Delta h$	<i>dhsic_nn</i>	
	<i>elev<sub>obs</sub></i>	<i>tobselev</i>	<a href="#">ForcKey.txt</a>
		<i>tcelevadd, tcobselev, monthlapse, tempcorr, tcalt</i>	<a href="#">par.txt</a>
Precipitation adjustments	$P_i$		<a href="#">Pobs.txt</a>
		<i>pcaddg, preccorr, pcurain, pcusnow, pcelevth, pcelevadd, pcelevstd, pcelevmax, pcluse</i>	<a href="#">par.txt</a>
	$\Delta h$	<i>dhsic_nn</i>	<a href="#">GeoData.txt</a>
	<i>elev<sub>std</sub></i>	<i>elev_std</i>	
	<i>snfrac</i>	calculated based on $T_i$ or from	<a href="#">SFobs.txt</a>
Rainfall and snowfall calculations		<i>ttmp, ttpd, ttpi</i>	<a href="#">par.txt</a>
	<i>sfrac</i>		<a href="#">SFobs.txt</a>
Snowfall distribution	U, V		<a href="#">UWobs.txt, VWobs.txt</a>
	<i>WSF</i>	<i>wsf_nn_d</i>	<a href="#">GeoData.txt</a>
		<i>WSFscale, WSFluse, WSBias, sfdmax</i>	<a href="#">par.txt</a>
	<i>limit</i>	<i>sfdlim</i>	

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">modelmodule (model_hype.f90)</a>	model
<a href="#">soil_processes (soil_proc.f90)</a>	calculate_snow
	infiltration
<a href="#">atmospheric_processes (atm_proc.f90)</a>	calculate_class_atmospheric_forcing
	calculate_rain_snow_from_precipitation
	calculate_subbasin_temperature
	calculate_subbasin_precipitation
	set_atmospheric_parameters_corrections
	calculate_winddirspeed
	calculate_snowfall_distribution

## Potential evaporation

Potential evaporation is calculated for each class based on land use and atmospheric variables. The potential evaporation is then used to calculate the current actual evapotranspiration from each land class (see [Soil water - Evaporation](#)). A river with an area (i.e. is a class), flooded floodplains and lakes are assumed to evaporate at the potential rate, when the air temperature is above the threshold temperature (*ttmp*). Evaporation is limited by the water body's volume.

Potential evaporation (*epot* in mm) is calculated based on the temperature if it is not read in from file

(*repo* in Xobs.txt). Alternative PET models exist, and is described [below](#). When the air temperature ( $T$ ) is greater than the threshold temperature *ttmp* evaporation is assumed to occur. Snow melting, snow density and evaporation use the same threshold temperature. The basic potential evapotranspiration ( $epot_{base}$ ) depends on the land use dependent rate parameter *cevp*.

$$cseason = 1 + cevpam \times \sin \left( \frac{2 \times \pi \times (dayno - cevpph)}{365} \right)$$

$$epot_{base} = (cevp \times cseason) \times (T - ttmp)$$

A seasonal factor *cseason* adjusts the potential evaporation rate (*cevp*) e.g. making it higher in the spring when the air is often dry, and lower in autumn when the air is often more humid than in spring. The factor is sinusoidal with two parameters *cevpam* and *cevpph*. It is not used if *cevpam* is zero. A *cevpph* around 45 days give a maximum correction in mid May (dayno=45+91=136). The minimum correction will then be a half year later in September (dayno=136+182). For an earlier maximum, reduce *cevpph*.

The basic potential evapotranspiration may be adjusted with a regional correction factor (*cevpcorr*) equally over the year depending on parameter region.

$$epot = epot_{base} \times (1 + cevpcorr)$$

## Alternative potential evaporation models

HYPE give the option to exchange the default potential evaporation model for another model. Only the basic potential evaporation ( $epot_{base}$ ) differs between models. Models 0-2 only use air temperature forcing. Models 3-5 want shortwave radiation and minimum and maximum daily air temperature, although if lacking approximations are made (see section [Input to alternative potential evaporation models](#) below). In addition, model 5 wants relative humidity and wind speed if available. Note, regardless of potential evaporation model, the actual evaporation is limited to temperatures above parameter *ttmp*.

### Model 0 (default)

As described above; evapotranspiration depends on the rate parameter *cevp* and air temperature ( $T$ ) above a threshold *ttmp*. If the variable *repo* is given in Xobs.txt those values are used.

$$epot_{base} = (cevp \times cseason) \times (T - ttmp)$$

### Model 1

Model 1 is the same as model 0, but it will not be using *repo* from input data, even if it is present.

## Model 2 - Modified Jensen-Haise/McGuinness

The modified Jensen-Haise/McGuinness model follow Oudin et al. (2005). The potential evaporation depends on extraterrestrial radiation ( $radext$ ), latent heat of vaporization ( $\lambda$ ) and temperature ( $T$ ). Two general parameters ( $jhtadd$  and  $jhtscale$ ) are used and one land use dependent (crop coefficient  $kc2$  or  $kc$ ).

$$epot_{base} = \frac{kc}{jhtscale} \times MAX \left( 0, \frac{radext}{\lambda} \times (T + jhtadd) \right)$$

## Model 3 - Modified Hargreaves-Samani

The Hargreaves-Samani evaporation is modified to limit the “turbidity-factor”. The potential evaporation depends on extraterrestrial radiation ( $radext$ ), latent heat of vaporization ( $\lambda$ ), temperature ( $T$ ) and turbidity ( $turbidity$ ). One general parameter ( $krs$ ) is used and one land use dependent (crop coefficient  $kc3$  or  $kc$ ).

$$epot_{base} = MAX \left( 0, kc \times 0.0023 \times \frac{radext}{\lambda} \times \frac{turbidity}{krs} \times (T + 17.8) \right)$$

## Model 4 - Priestly-Taylor

The Priestly-Taylor potential evaporation depends on net downward radiation ( $netrad$ ), slope of saturated vapour pressure curve ( $dsatvap$ ), latent heat of vaporization ( $\lambda$ ) and a psychrometric constant ( $\gamma$ ). One general parameter ( $alfapt$ ) are used and one land use dependent (crop coefficient  $kc4$  or  $kc$ ).

$$epot_{base} = MAX \left( 0, kc \times alfapt \times \frac{dsatvap \times netrad}{\lambda \times (dsatvap + \gamma)} \right)$$

## Model 5 - FAO Penman-Monteith

The FAO Penman-Monteith potential evaporation depends on net downward radiation ( $netrad$ ), slope of saturated vapour pressure curve ( $dsatvap$ ), saturated and actual vapour pressure ( $satvap$  and  $actvap$ ), temperature ( $T$ ), wind speed ( $wind$ ) and a psychrometric constant ( $\gamma$ ). One land use dependent parameter (crop coefficient  $kc5$  or  $kc$ ) is used.

$$epot_{base} = MAX \left( 0, \frac{kc \times 0.408 \times dsatvap \times netrad + \gamma \times \frac{900}{T + 273} \times wind \times (satvap - actvap)}{dsatvap + \gamma \times (1 + 0.34 \times wind)} \right)$$

## Input to alternative potential evaporation models

Summary of alternative input to PET models, and link to file reference.

Model	Parameters ( <a href="#">par.txt</a> )	Static data ( <a href="#">GeoData.txt</a> )	Forcing data ( <a href="#">files</a> )
0	cevp, ttmp		Xobs:repo, Tobs
1	cevp, ttmp		Tobs
2: modified Jensen-Haise/McGuinness	jhtadd, jhtscale, kc	latitude	Tobs
3: modified Hargreaves-Samani	kc, krs	elevation, latitude	SWobs, Tobs
3: modified Hargreaves-Samani	kc, krs	elevation, latitude, cloudiness	Tobs
3: modified Hargreaves-Samani	kc, krs	elevation, latitude	TMINobs, TMAXobs, Tobs
4: Priestly-Taylor	alb, alfapt, kc	elevation, latitude	RHobs, SWobs, TMINobs, TMAXobs, Tobs
4: Priestly-Taylor	alb, alfapt, kc, krs	elevation, latitude	SWobs, TMINobs, TMAXobs, Tobs
4: Priestly-Taylor	alb, alfapt, kc, krs	elevation, latitude	TMINobs, TMAXobs, Tobs
4: Priestly-Taylor	alb, alfapt, kc, krs	elevation, latitude, cloudiness	Tobs
5: FAO Penman-Monteith	alb, kc, roughness, zphd, zwind, zwish	elevation, latitude	RHobs, SWobs, TMINobs, TMAXobs, Tobs, Uobs
5: FAO Penman-Monteith	alb, kc, mwind	elevation, latitude	RHobs, SWobs, TMINobs, TMAXobs, Tobs
5: FAO Penman-Monteith	alb, kc, krs, roughness, zphd, zwind, zwish	elevation, latitude	SWobs, TMINobs, TMAXobs, Tobs, Uobs
5: FAO Penman-Monteith	alb, kc, krs, mwind	elevation, latitude	SWobs, TMINobs, TMAXobs, Tobs
5: FAO Penman-Monteith	alb, kc, krs, roughness, zphd, zwind, zwish	elevation, latitude	TMINobs, TMAXobs, Tobs, Uobs
5: FAO Penman-Monteith	alb, kc, krs, mwind	elevation, latitude	TMINobs, TMAXobs, Tobs
5: FAO Penman-Monteith	alb, kc, krs, mwind	elevation, latitude, cloudiness	Tobs



## Actual vapour pressure

Actual vapour pressure (*actvap*) is calculated following FAO recommended procedure and function/data priority. Depending on availability of minimum, mean and maximum relative humidity (*rh*) and minimum, mean and maximum air temperature ( $T_{\min}, T, T_{\max}$ ) equations with different combinations of saturated vapour pressure (calculated from temperature) times relative humidity is used. For example:

$$actvap = \frac{satvap(T_{\max}) \times rh_{\min} + satvap(T_{\min}) \times rh_{\max}}{2}$$

$$actvap = satvap(T_{\min}) \times rh_{\max}$$

$$actvap = satvap(T) \times rh_{mean}$$

$$actvap = satvap(T_{\min}) \times 1$$

In case not enough data is available, minimum temperature is calculated from turbidity and a general parameter (*krs*) and the last equation of those above is used.

$$T_{\min} = T - 0.5 \times \left( \frac{turbidity}{krs} \right)^2$$

Eventually actual vapour pressure is limited by the calculated saturated vapour pressure.

## Air pressure

Air pressure (*pa*) is calculated as a function of elevation for the class (*elev*).

$$pa = 101.3 \times \left( \frac{293 - 0.0065 \times elev}{293} \right)^{5.26}$$

## Shortwave radiation

Shortwave radiation (*swrad*) is supplied as input forcing time series or otherwise calculated from extraterrestrial radiation.

$$swrad = rade_{xt} \times turbidity$$

## Extraterrestrial radiation

Extraterrestrial solar radiation ( $rad_{ext}$ ) is estimated from day of year and latitude. The equations used comes from FAO. The day of the year are used to estimate distance to the sun and declination. The sunset hour angle are calculated from latitude, but with special care for high latitudes (polar night and midnight sun). These variables together with latitude and the solar constant eventually give the current extraterrestrial radiation for each subbasin.

## Latent heat of vaporization

Latent heat of vaporization ( $\lambda$ ) is a function of temperature ( $T$ ).

$$\lambda = 2.501 - 0.002361 \times T$$

## Net downward radiation

The net downward radiation ( $net_{rad}$ ) is used explicitly for PET model 4 and 5 (Priestly-Taylor and FAO Penman-Monteith). The net radiation is calculated following FAO recommended procedure. It is calculated as net shortwave radiation minus net longwave radiation. Net shortwave radiation ( $net_{short}$ ) is calculated from the shortwave radiation ( $sw_{rad}$ ) and the land use dependent albedo parameter ( $alb$ ). Net longwave radiation ( $net_{long}$ ) is calculated using temperature ( $T_{min}, T_{max}, T$ ), actual vapour pressure ( $act_{vap}$ ) and relative shortwave radiation ( $rel_{sh}$ ) if those are available, otherwise it is set to zero. The relative shortwave radiation is shortwave radiation in relation to clear sky shortwave radiation.

$$net_{short} = sw_{rad} \times (1 - alb)$$

$$net_{long} = 4.903 \times 10^9 \times \frac{(T_{max} + 273.15)^4 + (T_{min} + 273.15)^4}{2} \times \left( 0.34 - 0.14 \times act_{vap}^{0.5} \right) \times (1.35 \times rel_{sh} - 0.35)$$

$$rel_{sh} = \frac{turbidity}{clear_{turb}}$$

## Psychrometric constant

The psychrometric constant ( $\gamma$ ) is a function of air pressure ( $pa$ ) and latent heat of vaporization ( $\lambda$ ) following FAO.

$$\gamma = \frac{0.001013 \times pa}{0.622 \times \lambda}$$

## Saturated vapour pressure

Saturated vapour pressure (*satvap*, kPa) is calculated from temperature (*temp*) following FAO. If daily minimum and maximum air temperature is available *satvap* is set to the average of the saturated vapour pressure calculated for each of those two temperatures, otherwise it is calculated from daily average air temperature.

$$satvap = 0.6108 \times EXP \left( \frac{17.27 \times temp}{temp + 237.3} \right)$$

## Slope of saturated vapour pressure curve

The slope of saturated vapour pressure temperature function (*dsatvap*) is used explicitly for PET model 4 and 5 (Priestly-Taylor and FAO Penman-Monteith). The slope is calculated from daily mean air temperature (*T*).

$$dsatvap = 4098 \times 0.6108 \times EXP \left( \frac{17.27 \times T}{T + 237.3} \right) \times \frac{1}{(T + 237.3)^2}$$

## Turbidity

The turbidity factor is used explicitly for the PET model 3 - modified Hargreaves-Samani. If shortwave radiation (*swrad*) has been given as forcing data time series, the turbidity factor is calculated as

$$turbidity = \frac{swrad}{radext}$$

but limited by a minimum turbidity value (0.25) and an estimated clearsky turbidity. The clearsky turbidity (*clearturb*) is estimated by the Ångström formula (FAO):

$$clearturb = 0.75 + elev \times 0.00002$$

where *elev* is subbasin elevation in meter above sea level. If no shortwave radiation time series are given, but monthly cloudiness climatology is provided (*cloudiness*), the turbidity is calculated as

$$turbidity = (clearturb - 0.25) \times (1 - cloudiness) + 0.25$$

If no cloudiness are given either, but time series of daily minimum and maximum temperature are, the turbidity is calculated as in the “ordinary” Hargreaves-Samani turbidity function:

$$turbidity = krs \times SQRT(T_{\max} - T_{\min})$$

but still limited by the minimum turbidity value and the calculated clearsky turbidity.

## Wind speed

Wind speed is used in PET model 5 - FAO Penman-Monteith. Wind speed (*wind*) may be given as a constant general parameter (*mwind*) or as a forcing data time series. The time series wind is given for each subbasin ( $U_i$ ). It is possible to adjust the time series wind speed to different height than observations. If the general parameters *zwind*, *zwish*, *roughness*, and *zpdh* is set, wind speed is adjusted with the transformation factor *windtrans*.

$$wind = U_i \times windtrans$$

$$windtrans = \frac{\ln(zwind - zpdh) - \ln(roughness)}{\ln(zwish - zpdh) - \ln(roughness)}$$

## Links to file reference

Section	Symbol	Parameter/Data	File
Evaporation		<i>cevp</i> , <i>ttmp</i> , <i>cevpam</i> , <i>cevpph</i> , <i>cevpcorr</i> , <i>epotdist</i> , <i>lp</i> , <i>ttrig</i> , <i>tredA</i> , <i>tredB</i>	<a href="#">par.txt</a>
	<i>wp</i>	calculated from <i>wcwp</i> , <i>wcwp1</i> , <i>wcwp2</i> , <i>wcwp3</i> and <i>soillayerdepth</i>	
	<i>fc</i>	calculated from <i>wcfc</i> , <i>wcfc1</i> , <i>wcfc2</i> , <i>wcfc3</i> and <i>soillayerdepth</i>	
	<i>soillayerdepth</i>		<a href="#">GeoClass.txt</a>
Alternative potential evaporation models	<i>epot</i>		<a href="#">Xobs.txt</a>
		<i>jhtadd</i> , <i>jhtscale</i> , <i>kc</i> , <i>kc2</i> , <i>kc3</i> , <i>kc4</i> , <i>kc5</i> , <i>krs</i> , <i>alfapt</i>	<a href="#">par.txt</a>
Input to alternative potential evaporation models		<i>alb</i> , <i>mwind</i> , <i>zwind</i> , <i>zwish</i> , <i>roughness</i> , <i>zpdh</i>	<a href="#">par.txt</a>
	$U_i$		<a href="#">Uobs.txt</a>
	<i>swrad</i>	calculated or from	<a href="#">SWobs.txt</a>
	<i>elev</i>	<i>elev_mean</i>	<a href="#">GeoData.txt</a>
		<i>latitude</i> , <i>cloudiness</i>	

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">soil_processes (soil_proc.f90)</a>	<i>initiate_soil_water</i>
	<i>calculate_potential_evaporation</i>
	<i>calculate_actual_soil_evapotranspiration</i>
<a href="#">surfacewater_processes (sw_proc.f90)</a>	<i>calculate_river_evaporation</i>
	<i>calculate_actual_lake_evaporation</i>

## References

Richard G. Allen, Luis S. Pereira, Dirk Raes & Martin Smith, 1998. Crop Evapotranspiration – Guidelines for Computing Crop Water Requirements. FAO Irrigation and drainage paper 56. Rome, Italy: Food and Agriculture Organization of the United Nations. ISBN 92-5-104219-5

Oudin, L., F. Hervieu, C. Michel, C. Perrin, V. Andreassian, F. Anctil and C. Loumagne 2005. Which potential evapotranspiration input for a lumped rainfall-runoff model?: Part 2—Towards a simple and efficient potential evapotranspiration model for rainfall-runoff modelling, *Journal of Hydrology* 303(1-4):290-306.

## Atmospheric deposition

Atmospheric deposition can be added to the model as load or as concentration of precipitation.

### Wet deposition

Atmospheric deposition in the form of wet deposition of IN and SP is added as a concentration of rainfall. You can specify a time series of the concentration in precipitation in Xobs.txt (implemented for IN, SP, and T1). Otherwise wet deposition (as a concentration) is specified in AtmdepData.txt for each subbasin.

If parameter (*aloadconst*) is set, the wet deposition load is considered constant and not dependent on precipitation corrections. Thus the concentration of precipitation is changed to keep the load constant, when precipitation correction is applied.

### Atmospheric deposition to the soil

Dry deposition is specified in AtmdepData.txt for each subbasin. It can be specified for different vegetation types or land uses.

Dry deposition of nitrogen and phosphorus is added to the snow or, if there is no snow, directly to the ground. Phosphorus deposition is added to PartP-pool in the upper soil layer and nitrogen deposits to the IN dissolved in soil water if the parameter *ponatm* is not set. A new concentration of IN in the soil water is then calculated. The parameter *ponatm* indicates that some of the nitrogen deposition will be added to organic nitrogen pool (*fastN*) instead of to the IN pool.

For both nitrogen and phosphorus wet deposition is added through the concentration of precipitation (as described above). Precipitation may fall as snow and be added to the snow pack or if it is rain added to the potential infiltration. Depending on the fate of the infiltration, the wet atmospheric deposition will mix with the receiving water (e.g. surface runoff, soil layer water, macropore flow).

### Atmospheric deposition to rivers and lakes

Deposition is specified in AtmdepData.txt for each subbasin, as load and/or as concentration for wet deposition. It can be specified for different vegetation types or land uses, and thus water surfaces may have a different deposition than land surfaces. In addition monthly deposition can be specified.

For both nitrogen and phosphorus wet deposition is added through the concentration of precipitation (as described above), while for dry deposition an amount of the nutrient is added to the river (if it has a class-area) or lake water.

## Links to file reference

Section	Parameter/Data	File
Wet deposition	<i>cpT1, cpIN, cpSP</i>	<a href="#">Xobs.txt</a>
	<i>XX_WD, XX_WD_Ln, XX_WD_Vn</i> ( <i>XX</i> is substance, <i>n</i> is index of landuse or vegetation type)	<a href="#">AtmdepData.txt</a>
	<i>aloadconst</i>	<a href="#">par.txt</a>
Atmospheric deposition to the soil	<i>XX_DD, XX_DD_Ln, XX_DD_Vn, XX_DD_Mn_Vn</i> ( <i>XX</i> is substance, <i>n</i> is index of landuse, vegetation type, or month)	<a href="#">AtmdepData.txt</a>
	<i>ponatm</i>	<a href="#">par.txt</a>
Atmospheric deposition to rivers and lakes	<i>XX_DD, XX_DD_Ln, XX_DD_Vn, XX_DD_Mn_Vn</i> ( <i>XX</i> is substance, <i>n</i> is index of landuse, vegetation type, or month)	<a href="#">AtmdepData.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">npc_soil_processes</a> ( <a href="#">npc_soil_proc.f90</a> )	<code>add_dry_deposition_to_landclass</code>
	<code>atmdep_in_loss</code>
<a href="#">atmospheric_processes</a> ( <a href="#">atm_proc.f90</a> )	<code>set_class_precipitation_concentration_and_load</code>
	<code>set_T2_concentration_in_precipitation_on_water</code>
<a href="#">npc_surfacewater_processes</a> ( <a href="#">npc_sw_proc.f90</a> )	<code>add_deposition_to_river_as_load</code>
	<code>add_deposition_to_lake_as_load</code>

# Land routines

This section explains the computations in the land routines of HYPE. If you want an interactive overview of how the routines simulates runoff please have a look at [the HYPE Runoff Explorer](#).

## Basic assumptions

A subbasin in HYPE is divided into classes depending on land use, soil type etc. The classes are called SLC:s which stands for Soil type Land use Combination. This division can be compared to hydrological response units. Lakes and river is also classes, but this section consider only land classes.

HYPE model soil routine contains (up to) three soil layers. The number of soil layers and their respective lower limits (*soillayerdepth* in meters, figure 1) are listed by class in GeoClass.txt. It is possible to have a different number of layers and depth of these different classes.

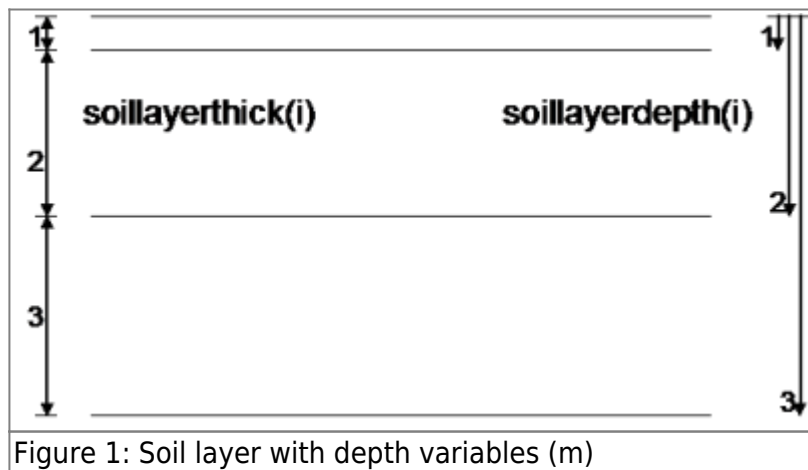


Figure 1: Soil layer with depth variables (m)

The parameters of water retention in the soil are wilting point (*wcwp*), field capacity (*wcfc*) and effective porosity (*wcep*). These are exclusive and are specified in units of depth (figure 2). The model allocates the water retention capacity evenly between soil layers, depending on their thickness, if only one value set. It is also possible to specify one value per parameter and layer. The model uses water holding capacity in mm for the layers. These are represented in the model code and this document (figure 3) as *wp(i)*, *fc(i)* and *ep(i)*. The model parameters for the water storage capacity *wcfc*, *wcwp*, *wcep* (and layer depending alternative *wcwp1*, *wcwp2*, *wcwp3*, *wcfc1*, *wcfc2*, *wcfc3*, *wcep1*, *wcep2*, *wcep3*) depend on soil type.

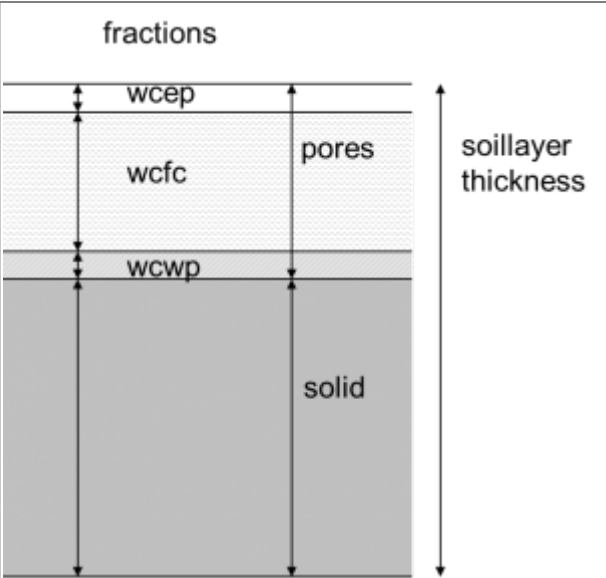


Figure 2: Water retention parameters.

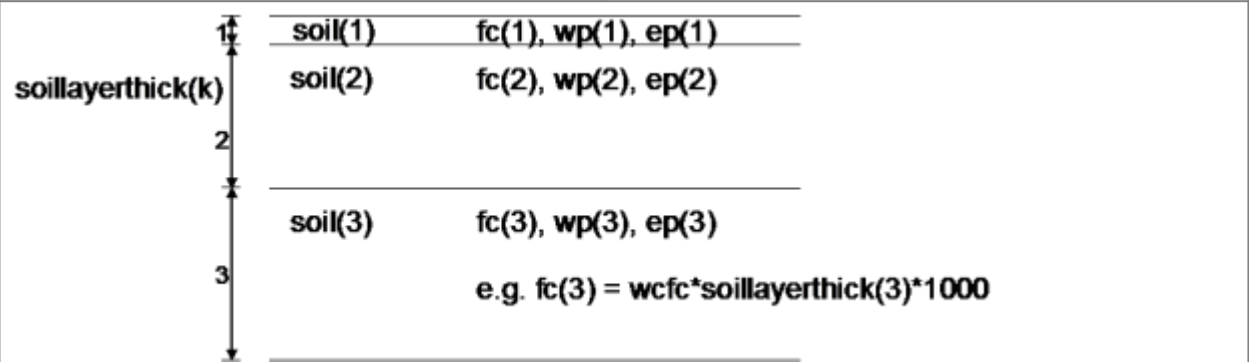


Figure 3: Soil layers water content (mm) and parameters of the water storage capacity (mm).

The soil layer water content (mm) is represented in the document as *soil(i)*, where *i* = layer, but the value also depends on the subbasin and class (figure 3). The initial value of soil water is set to *wp* + *fc*. Optionally the initial value may be set to saturation (*wp* + *fc* + *ep*). The upper soil layer can hold more water than the porevolume. Standing water is not treated as a separate pool.

Tile drainage can be placed in any soil layer (figure 4). The depth of the drainage pipe in meters (*tiledepth*) is specified in GeoClass.txt for each class. A depth of 0 m is interpreted as drainage pipes are missing.

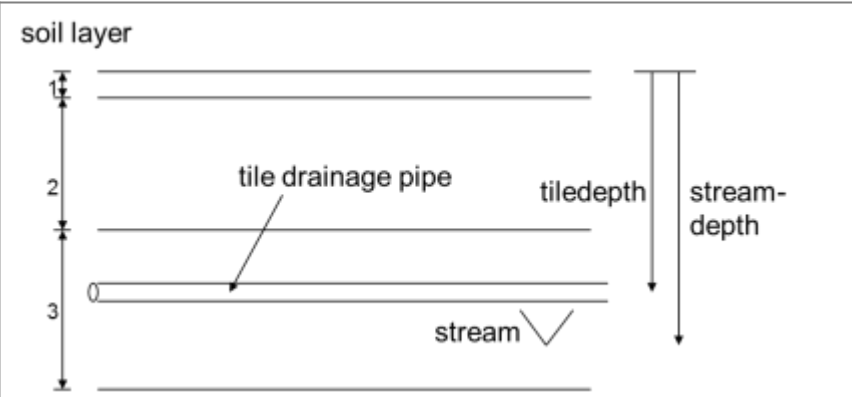


Figure 4: Drainage pipe and creek bottom in the third soil layer.

Another depth (*streamdepth*) is specified in GeoClass.txt; this is the maximum depth of the drainage to stream (or ditch). Soil water below this level does not contribute to the local runoff. Note that all



land runoff goes through the local river (and possibly local lake) and then the main river before it reaches the outlet lake. There is no direct runoff to lakes.

## Overview of flow paths

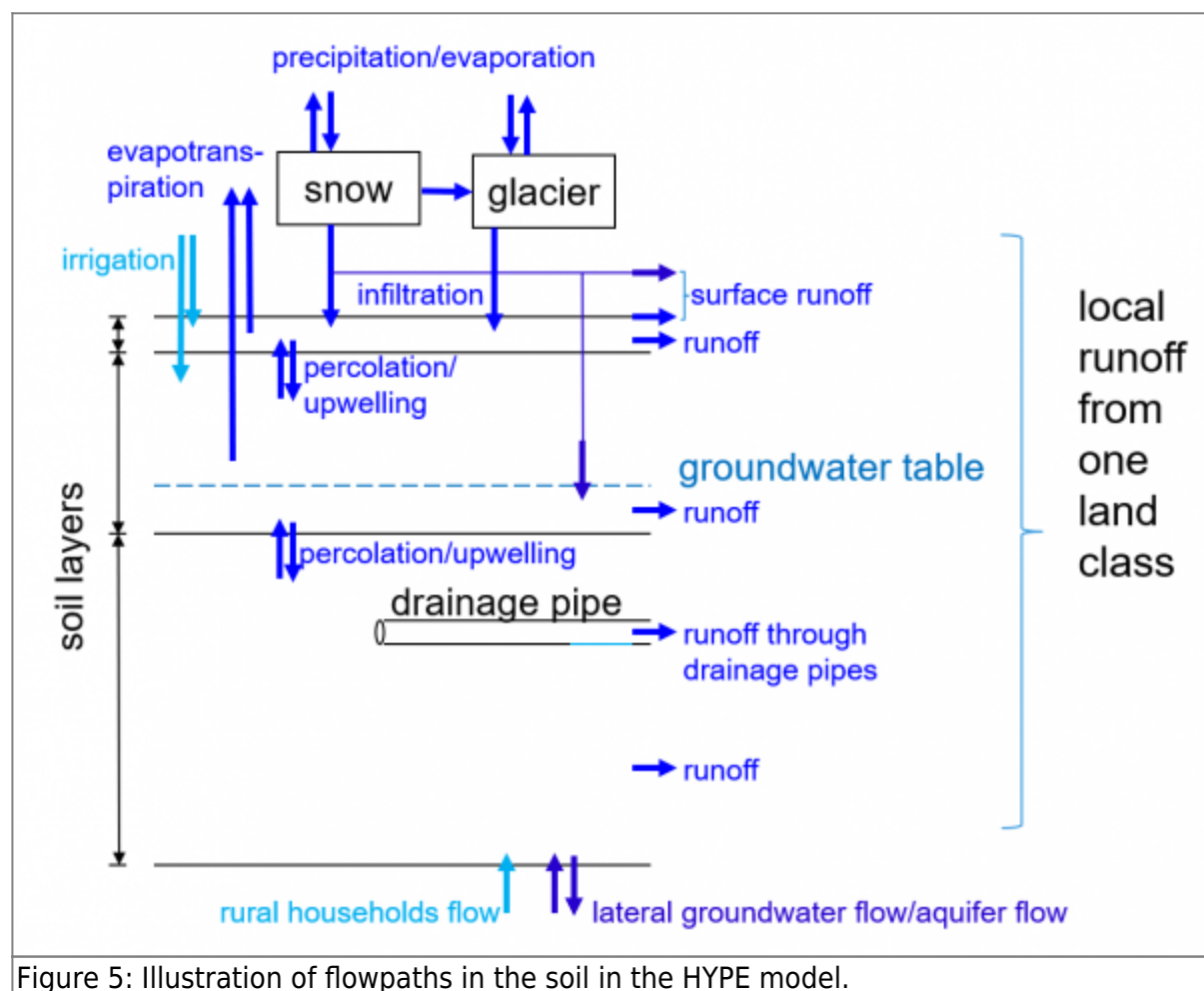


Figure 5: Illustration of flowpaths in the soil in the HYPE model.

## Diagnostic variables

Some additional output variables are calculated from the soil state variables.

### Groundwater level

The groundwater level is measured negative from surface (0m) to bottom of the soil layers. A positive groundwater level means that the soil surface is below water. If the ground water table reaches above the surface, the water is calculated with 100% porosity.

The water table is found in the lowest soil layer that is not completely filled with water. Soil layers above this layer may have water in its effective porosity, but that is not included in the groundwater level output variable. The water table for a soil layer is calculated linearly from the proportion of water-filled pores of effective porosity part of the soil pore volume. If the soil moisture of a soil layer is at field capacity (or below), the groundwater level of that soil layer is at the bottom of the layer. If the pore volume is filled, the groundwater level of that soil layer is at the top of the layer.

## Soil moisture deficit

Soil moisture deficit (*smdf*) is calculated for the root zone, i.e. the upper two soil layers. It is the water (in *mm*) needed to fill the soil (*soil*) to field capacity (*wp+fc*).

$$smdf = \sum_{k=1}^2 MAX \left\{ fc_k + wp_k - soil_k, 0 \right\}$$

## Frost depth

Ground frost depth (which is only calculated if the soil temperature is less than zero) depends on soil temperature (*soiltemp*), but also on soil water content (*soil*), field capacity (*fc*), wilting point (*wp*) and two parameters *frost* and *sfrost*.

$$frostdepth = \frac{frost \times sfrost \times soiltemp \times (fc + wp)}{soil}$$

There are two parameters in order to be able to choose if you want the frost depth to be land use dependent or soil dependent. The not used parameter is set to one.

## Links to file reference

Section	Symbol	Parameter/Data	File
Basic assumptions		<i>soillayerdepth, tiledepth, streamdepth</i>	GeoClass.txt
	<i>soillayerthick</i>	calculated from <i>soillayerdepth</i>	
	<i>wp, fc, ep</i>	calculated from <i>wcwp,wcfc,wcep,wcwp1-3,wcfc1-3,wcep1-3</i>	par.txt
Diagnostic variables		<i>frost,sfrost</i>	
	<i>soiltemp</i>	see needed data in <a href="#">Links for soil temperature</a>	

## Links to relevant procedures in the code

Modules (file)	Procedures	Section
<a href="#">soil_processes (soil_proc.f90)</a>	<i>initiate_soil_water_state</i>	basic assumptions
	<i>initiate_soil_water</i>	
	<i>calculate_groundwater_table</i>	ground water level
	<i>calculate_soil_moisture_deficit</i>	soil moisture deficit
	<i>calculate_frostdepth</i>	frost depth

## Snow routines

The basic simulation of snow is accumulation of snowfall to a snow pack and snow melt that releases the water to infiltration. Other processes may be included dependent on model options and parameters set. These include liquid fraction of snow simulating snow water holding capacity,

evaporation and sublimation, snow heat content, refreezing of liquid water in snow.

## Snow melt

For the simplest snow model, snow melt is calculated by temperature index. Snow melting occurs proportionally to temperature, when the temperature is greater than a threshold temperature. Other options are to consider the effect of radiation and snow cover.

The alternative snowmelt models are not fully described here yet.

### Model 0 (default)

The default model is a temperature index model, with or without snow cover scaling. Snow melting occurs when the temperature is greater than a threshold temperature. The potential snow that melts ( $pmelt$ ) depends on the snowmelt parameter  $cmlt$ , threshold temperature parameter  $ttmp$  and air temperature ( $temp$ ). Additionally snow melt may be adjusted by the snow cover ( $effcover$ ). The parameter  $fsceff$  determine how large effect the snow cover scaling should have, between zero and one. The actual snow melt is limited by the amount of snow ( $snow$  in mm).

$$effcover = 1 - (1 - fsceff) \times snowcov$$

$$pmelt = cm \times (temp - ttmp) \times effcover$$

$$cm = cmlt \times (1 + cmltcrr)$$

$$melt = MIN(pmelt, snow)$$

The parameters  $cmlt$  and  $ttmp$  are related to land use and  $cmltcrr$  depend on parameter region, while  $fsceff$  is general.

### Model 2

the second model is a temperature and radiation index model, with or without snow cover scaling. The temperature index potential snow melt is calculated the same way as the default model ( $pmelt$ ). In addition potential snow melt by radiation index ( $rpmelt$ ) is calculated from shortwave radiation ( $swrad$ ) and albedo of the snow ( $albedo_{snow}$ ). Additionally snow melt may be adjusted by the snow cover ( $effcover$ ), see model 0.

$$rpmelt = cmrad \times swrad \times (1 - albedo_{snow}) \times effcover$$

The parameter  $cmrad$  is related to land use. Snow albedo is calculated as decreasing with the age of snow ( $snowage$ ), and depend on land use specific parameters ( $albmax$ ,  $albmin$  and  $albkexp$ ).

$$albedo_{snow} = albmin + (albmax - albmin) \times EXP(-albkexp \times snowage)$$

The two potential melting parts are added for this model. Snow melt routine consider also that snow melt (and liquid content of snow) can refreeze for temperatures below the threshold ( $ttmp$ ). The refreezing resuces the snow melt. The refreezing occur when temperature is below the threshold ( $ttmp$ ) and is a fraction of the (negative) potential temperature index "snow melt". The fraction is given by the general model parameter  $cmrefr$ .

$$refreeze = cmrefr \times cmlt \times (ttmp - temp)$$

$$melt = MIN \left( pmelt + rpmelt - refreeze, snow \right)$$

## Snow heat

A model option for snow heat delays the snow melt until the temperature of snow is zero. Snow temperature is calculated from snow heat and snow water equivalent. Snow heat model uses general parameters  $sdnsnew$  and  $snkika$ .

## Snow cover

Normally snow is assumed to cover the whole class if present. Alternatively if parameters are given, snow cover fraction ( $fsc$ ) within a class is calculated based on snow water equivalent ( $snow$ ). The formulation is based on Samuelsson et al. (2006). During snow build up the snow cover increase as a function of snow water equivalent until a maximum value (general parameter  $fscmax$ ) is reached.

$$fsc = fscmax \times \tanh \left( 0.1 \times snow \right)$$

It is also possible to specify a minimum snow cover (general parameter  $fscmin$ ). As soon as the fractional snow cover area reaches above a certain threshold ( $fscmax - fscmin$ ), the snow cover area is determined by another relation that represents the redistribution of snow during winter. In this case snow cover is dependent on maximum snow pack during the winter ( $snowmax$ ) and a snow cover redistribution factor that is dependent on variation in elevation ( $stdelev$ , the standard deviation of elevation within the subbasin) and land use.

$$fsc = \frac{snow}{snowmax \times fscdist}$$

$$fscdist = fscdist0 + fscdist1 \times stdelev$$

The snow distribution factor ( $fscdist$ ) is determined by three land use dependent parameters;  $fscdist0$  and  $fscdist1$  in the linear equation and a maximum value ( $fscdistmax$ ). Also in this case the snowcover is limited by the maximum and minimum value parameters. When the end of the snow season approaches (defined by general parameter  $fsck1$ ) the  $snowmax$  variable is gradually decreased in order to be reset before next winter season:

$$snowmax = snowmax - \frac{\left( fsck1 \times snowmax - snow \right) \times 1 - e^{-fscexp \times ts}}{fsck1}$$

$$\frac{snow}{snowmax} < fsck1$$

The equation depends on two general parameters, *fsck1* and *fsckexp*, where *fsckexp* depend on time (*ts* is seconds per timestep of simulation).

For winters when the snow pack does not reach the definition of large snow pack, the first equation is used during the whole season. For winters when the snow does not melt completely, the second equation is continued to be used.

## Snow depth

In the default snow depth model, snow density (*snowdens*) depends on the snow's age in days (*snowage*). Snow density for fresh snow (*sdnsnew*) and the increase of density with snow age (*snowdensdt*) are general parameters (~ 0.1 and ~0.002). The snow's age increases by one every time step, but are weighted with age (0) for any new snow.

$$snowage = \frac{(snowage + 1) \times oldsnow}{(oldsnow + snowfall)}$$

$$snowdens = sdnsnew + snowdensdt \times snowage$$

$$snowdepth = \frac{0.1 \times snow}{snowdens}$$

In the alternative snow depth model, snow density is calculated by a compacting factor. Snow density for fresh snow (*sdnsnew*), maximum snow density (*sdnsmax*), compactation rate for low temperatures (*sdnsrate*) and additional compactation for high temperature (*sdnsradd*) are all general parameters. The change in snowdensity (*densdt*) due to compactation each time step is calculated as:

$$densdt = sdnsrate \times (sdnsmax - snowdens)$$

for cold days (temperature is below threshold temperature parameter *ttmp*), and

$$densdt = (sdnsrate + sdnsradd) \times (sdnsmax - snowdens)$$

for warm days.

## Links to file reference

Section	Symbol	Parameter/Data	File
Snow		<i>whcsnow</i>	<a href="#">par.txt</a>
Snow melt	<i>ttmp, fcseff</i>	<i>ttmp, fcseff</i>	<a href="#">par.txt</a>
	<i>cm</i>	<i>cmlt, cmltcorr</i>	
	<i>albmin, albmax, albkexp, cmrad</i>	<i>albmin, albmax, albkexp, cmrad</i>	
		<i>sdnsnew, snkika</i>	<a href="#">Tobs.txt</a>
	<i>T</i>	calculated from	

Section	Symbol	Parameter/Data	File
Snow cover	<i>stdelev</i>	<i>elev_std</i>	<a href="#">GeoData.txt</a>
	<i>fscmax, fscdist0, fscdist1, fsck1, fsckexp</i>	<i>fscmax, fscdist0, fscdist1, fsck1, fsckexp</i>	<a href="#">par.txt</a>
		<i>fscmin, fsclim, fscdistmax</i>	
Snow depth		<i>sdnsnew, snowdensdt, sdnsmax, sdnsrate, sdnsradd, ttmp</i>	<a href="#">par.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures	Section
<a href="#">soil_processes (soil_proc.f90)</a>	<i>calculate_snow</i>	snow melt
	<i>calculate_snowmelt</i>	
	<i>snowalbedo_function</i>	
	<i>calculate_snowheat_processes</i>	
	<i>snow_thermal_conductivityfunction</i>	
	<i>calculate_fractional_snowcover</i>	snow cover
	<i>calculate_snowdepth</i>	snow depth

## Soil water

### Soil temperature and frozen soil

Soil layer temperature (*soiltemp*) is calculated as a balance of three temperatures; previous time step soil layer temperature, soil temperature at deep depth (*deeptemp*) and air temperature (*T*). The model is based on Lindström et al. (2002). The weight of the deep soil is constant (0.001), while the weight of the air temperature (*weight<sub>air</sub>*) depends on snow depth (*snowdepth*) and parameters. The soil memory (*soilmem*) depends on depth and land use, with parameters *surfmem* and *depthrel*. The memory of deep soil temperature is a general parameter (*deepmem*).

$$soilmem = \begin{cases} deepmem & \text{for } deeptemp \\ surfmem \times e^{-depthrel \times depth(k)} & \text{for soil layer } k \end{cases}$$

$$weight_{air} = \frac{1}{soilmem + 10 \times snowdepth}$$

$$deeptemp = weight_{air} \times T + \left(1 - weight_{air}\right) \times deeptemp$$

$$soiltemp = weight_{air} \times T + \left(1 - weight_{air} - weight_{deep}\right) \times soiltemp + weight_{deep} \times deeptemp$$

Negative soil temperature will freeze part of the soil water and affect evaporation, percolation and runoff. It was used by Stadnyk et al (2020). The effect of frozen soil can be included in simulation by a model option. The model option come in two variants.

A fraction of the soil water is assumed in liquid phase for each soil layer (*liqfrac*). It is assumed equal in the different “pores”, i.e the same fraction frozen in water below wilting point, in water in field capacity and in water available for runoff. The fraction of liquid water for a soil layer is calculated from the temperature of the soil layer (*soiltemp*, degree Celsius), soil water (*water*), porosity (*pw*) and two soil type dependent parameters ( $par_{logsatm}$ ,  $par_{bcosby}$ ):

$$liqfrac = 1, soiltemp > 0$$

$$liqfrac = \frac{pw}{water} \times \left( \frac{-334000 \times soiltemp}{9.81 \times (soiltemp + 273.16) \times 10^{\frac{par_{logsatm}}{100}} \times \frac{1}{100}} \right)^{-\frac{1}{par_{bcosby}}}, soiltemp < 0$$

An alternative model for calculation of liquid fraction is available. In the alternate model each soil layer is divided into three equal thick temporary layers and a soil temperature for each of these are determined. Then the fraction of liquid phase is calculated for the temporary layers based on their soil temperatures. The average of the *liqfrac* for the temporary layers is then applied to the soil layer in the following calculations.

Frozen soil affects evapotranspiration, percolation, soil layer runoff and tile runoff. Actual evapotranspiration is decreased by the fraction of frozen water in soil. Percolation is only acting on the liquid water of the soil. For runoff the frozen soil influences in two ways; one, the water available for runoff is reduced with the frozen fraction, and two the frozen water is assumed to expand.

The expansion of ice decreases pore volume for liquid water. In HYPE this is assumed to affect soil layer runoff by increasing the pressure level in the soil layers. Since the ice is assumed equally divided between pores, this can actually force water to fill ep-pores and have some water available for runoff even though  $water < wp + fc$ . Expansion is set with a general parameter, *fzsexpand*, and could be up to 10% (i.e. parameter value 0.1).

## Evapotranspiration

Evaporation from soil is assumed to occur from the two upper layers. The potential evaporation is assumed to decrease exponentially with depth (depending on the parameter *epotdist*). The potential evaporation is divided between the two layers (*epotfrac*) with the distribution depending on the potential evaporation in the midpoint of each soil layer (figure 6). This is then used by approximating to a rectangle. Since soil layers differ between classes, the evaporation distribution do to.

$$epot1 = EXP \left( - epotdist \times soillayerdepth(1) / 2 \right)$$

$$epot2 = EXP \left( - epotdist \times \left( soillayerdepth(1) + \frac{soillayerdepth(2) - soillayerdepth(1)}{2} \right) \right)$$

$$area1 = soilayerdepth(1) \times epot1$$

$$area2 = \left( soilayerdepth(2) - soilayerdepth(1) \right) \times epot2$$

$$epotfrac1 = \frac{area1}{area1 + area2}$$

$$epotfrac2 = \frac{area2}{area1 + area2}$$

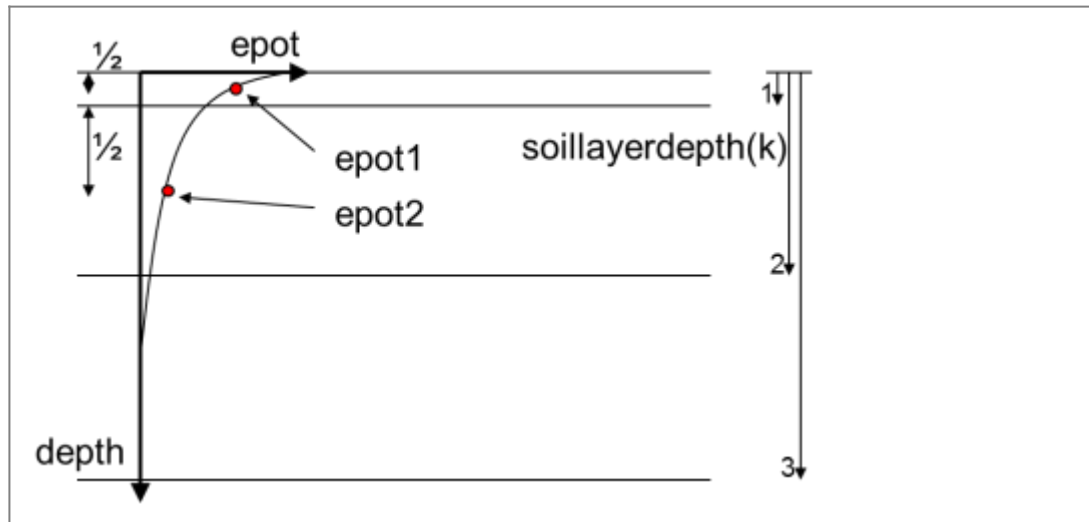


Figure 6 The distribution of potential evaporation between the top two soil layers.

The actual evaporation from a soil layer (*evap*) is limited by the availability of water in the soil (*soil*) above the wilting point (*wp*, mm). Evaporation is at potential rate only if the water exceeds field capacity (*fc*, mm) or a (large) proportion (general parameter *lp*) of field capacity. In between these limits evaporation increase linearly.

$$evapp = \begin{cases} 0 & soil - wp < 0 \\ epot \times epotfrac & soil - wp > lp \times fc \\ epot \times epotfrac \times \frac{soil - wp}{lp \times fc} & else \end{cases}$$

$$evap = MAX \left( evapp, soil - wp \right)$$



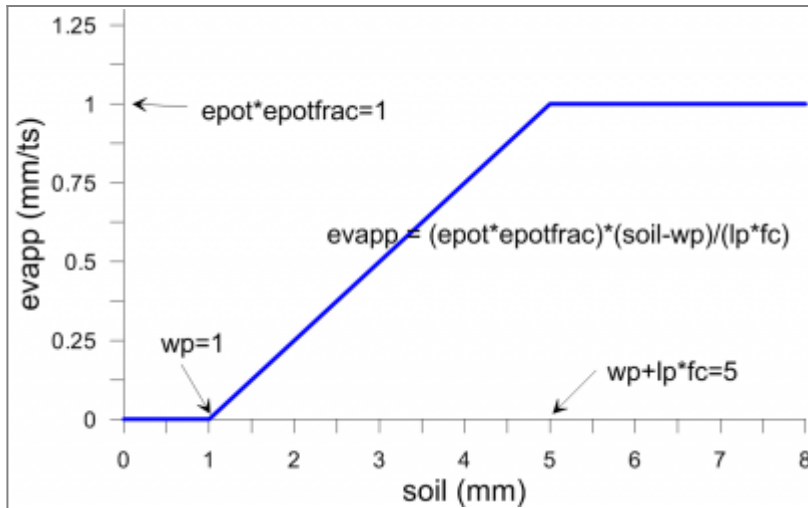


Figure 7 Evaporation as a function of soil water.

The actual evaporation may also depend on soil temperature (*soiltemp*). It is then reduced for temperatures above land use parameter *ttrig* and depend on two other land use parameters (*tredA*, *tredB*) as well (Figure 3).

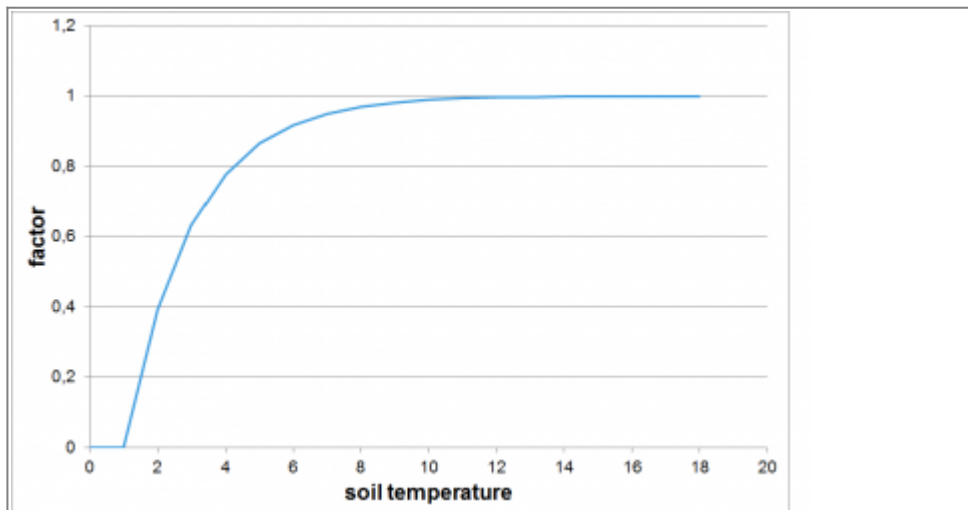


Figure 8 Soil temperature factor for reduction of soil evapotranspiration. Parameter values: *ttrip*=1, *tredA*=0.5, *tredB*=1.

The soil temperature evapotranspiration reduction is calculated as:

$$factor = 1 - e^{\left( -tredA \times (soiltemp - ttrig)^{tredB} \right)}$$

$$evapp = evapp \times factor$$

The actual soil evaporation is set to zero for temperatures below the threshold temperature and for negative potential evaporation estimates (condensation). It may also be affected by frozen soil model, which then limit evaporation to the liquid part of soil water.

## Groundwater runoff

Runoff depends on the water table in relation to the drainage level. Runoff occurs when soil water

reaches above field capacity in the soil layers. Runoff depends on soil water in the effective porosity (also used to calculate the groundwater table) and a recession coefficient ( $rc$ ). If the soil is not saturated, runoff from the soil layer depends only on the water of that soil layer. Runoff occurs from all three soil layers ( $runoff(k)$ ,  $k=1-3$ ) down to the drainage level, which is the depth of the stream (streamdepth).

$$runoff(k) = \begin{cases} rc(k) \times (soil(k) - wp(k) - fc(k)) & wp(k) + fc(k) < soil(k) < wp(k) + fc(k) + ep(k) \\ 0 & soil(k) < wp(k) + fc(k) \end{cases}$$

For the soil layer at drainage level, e.g. in the third layer of figure 6, and if the soil layer is not saturated, the runoff depends on the water level above the stream depth and the following equation replaces the one above.

$$runoff(k) = \begin{cases} rc(k) \times \frac{deltah \times ep(k)}{soillayerthick(k)} & deltah > 0 \\ 0 & deltah < 0 \end{cases}$$

where

$$deltah = \frac{(soil(k) - wp(k) - fc(k))}{ep(k)} \times soillayerthick(k) - (soillayerdepth(k) - streamdepth)$$

If the frozen soil model is used, it influences in two ways; one, the water available for runoff is reduced to the liquid fraction, and two, the frozen water is assumed to expand possibly increasing the water level in the soil.

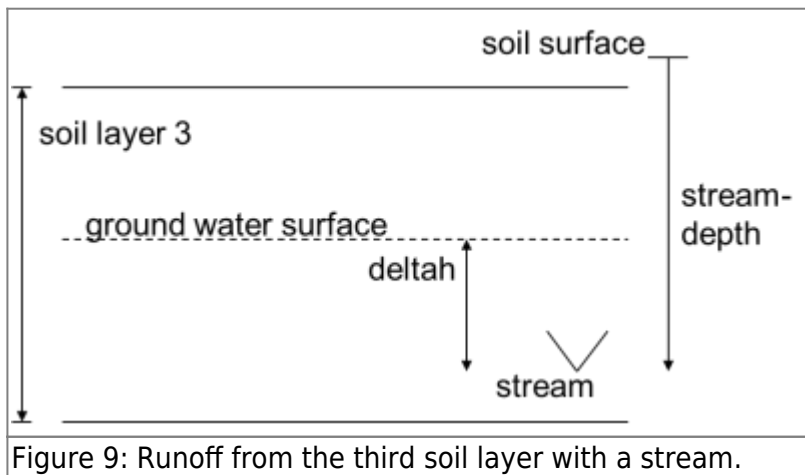


Figure 9: Runoff from the third soil layer with a stream.

Soil layers that lie entirely below the stream depth have no groundwater runoff.

If a soil layer is saturated, i.e.  $soil \geq fc + wp + ep$ , the runoff of the soil layer depends also on the water

in the soil layer(s) above. For example if the drainage is in soil layer 3 and both soil layer 2 and 3 are saturated, the groundwater table in soil layer 1 determines the runoff of soil layer 3. The runoff is limited to the water above field capacity in the third layer though.

$$runoff(3) = MIN \left( soil(3) - wp(3) - fc(3), rc(3) \times \frac{deltah}{soillayerthick(3)} \times ep(3) \right)$$

$$deltah = \frac{\left( soil(3) - wp(3) - fc(3) \right) \times soillayerthick(3)}{ep(3)} - \left( soildepth(3) - streamdepth \right) + \frac{\left( soil(2) - wp(2) - fc(2) \right) \times soillayerthick(2)}{ep(2)} + \frac{\left( soil(1) - wp(1) - fc(1) \right) \times soillayerthick(1)}{ep(1)}$$

If the stream depth is below the bottom of the lowest soil layer. The extra distance will act as a level to increase *deltah* and the runoff from the lowest soil layer.

### Recession coefficient of groundwater runoff

The recession coefficient is calculated from two parameters, *rrcs1* and *rrcs2* which depend on soil type and a parameter that is general *rrcs3*. The recession coefficient is assumed to decrease with depth and the parameters indicate the coefficient value in the topmost layer (*rrcs1*) and in the bottom layer (*rrcs2*). If *rrcs2* is not specified it is assumed to be similar to the recession in the topmost layer. The third parameter adjusts the upper layer recession due to the subbasin gradient (*slope*).

$$rrcs1 = rrcs1 \times (1 + rrcscorr) + rrcs3 \times slope$$

$$rrcs2 = rrcs2 \times (1 + rrcscorr)$$

The correction factor corrects *rrcscorr* parameters *rrcs1* and *rrcs2* for different parameter regions (*parreg*). It is defined as an increase. Note that the recession is limited to one. The recession is assumed to diminish exponentially and values of *rrcs1* and *rrcs2* applies to the midpoint of each layer ( $d_k$ ).

$$rc(k) = rrcs \times e^{-b \times d_k}$$

The variable *b* is an auxiliary variable.

$$b = \frac{\log \left( \frac{rrcs1}{rrcs2} \right)}{\left( soilayerdepth(3) - \frac{soillayerthick(3)}{2} \right) - \frac{soillayerthick(1)}{2}}$$

The result is:

$$rc(1) = trrcs1$$

$$rc(2) = trrcs1 \times \exp \left( -b \times \left( \frac{soillayerthick(1)}{2} + \frac{soillayerthick(2)}{2} \right) \right)$$

$$rc(3) = trrcs2$$

## Runoff through drainage pipes

Runoff in the drainage pipes occurs when the water table (the percentage of filled pores of the effective porosity) rises above the pipe's depth (figure 10). Runoff depends on the groundwater surface elevation over the pipe (*deltah*, m), and a recession coefficient *trrcs*. Recession parameter *trrcs* depends on soil type, while drainage pipe level depends on the class. The recession parameter is adjusted with the correction parameter *rrcscorr* for different parameter regions (*parreg*). It is defined as an increase.

$$trrcs = trrcs \times (1 + rrcscorr)$$

In addition the tile drainage can be adjusted if not the whole class is assumed to be drained by ditches. The fraction of drain area for the class is allowed to affect the recession parameter to reduce the effect of tile drainage.

$$trrcs = trrcs \times \text{tilefrac}$$

where *tilefrac* is given per subbasin and class, or a group of classes called *tilegroup*.

Depending on which soil layer drainage pipe is in, the runoff will be calculated for water in that soil layer. For the soil layer *k* (*soil(k)* is the water content in soil layer *k*) runoff is calculated as the parameter *trrcs* times the water found in the effective porosity of the layer and of the overlying soil layers if it is full.

```
deltah = (soil(k)-wp(k)-fc(k))/ep(k) * soillayerthick(k) -
          (soillayerdepth(k) - tiledepth)
IF(soil(k)-wp(k)-fc(k)-ep(k)>=0.) deltah= deltah + (soil(k-1)-wp(k-1)-
          fc(k-1))/ep(k-1) * soillayerthick(k-1)
IF(deltah>0.)
  runoffd = trrcs * deltah / soillayerthick(k) * ep(k)
```

If the frozen soil model is used, it influences in two ways; one, the water available for runoff is reduced to the liquid fraction, and two, the frozen water is assumed to expand possibly increasing the water level in the soil.

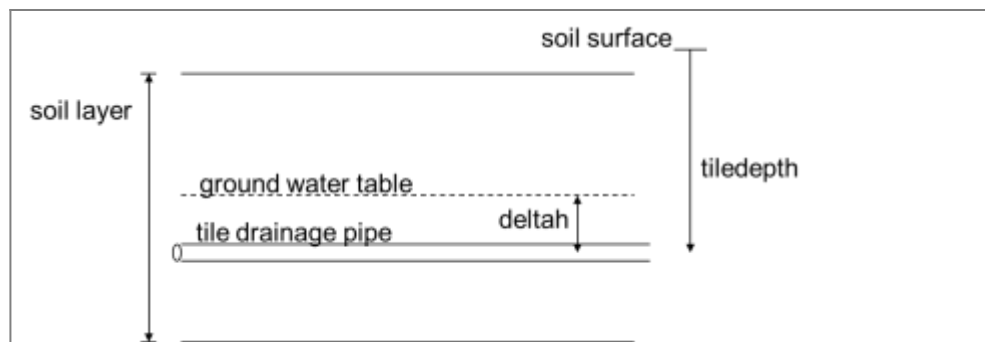


Figure 10: Illustration for calculation of runoff through the drainage pipes.

## Infiltration and surface runoff

Infiltration is calculated from the sum of rain and snowmelt (*infilto*, mm/time step) .

$$in\ filto = rain\ fall + melt$$

Part of the available water for infiltration (*infilto*) may not infiltrate into the soil, due to limitations by the soil's infiltration capacity and other properties of the soil. The calculation of actual infiltration will consider effects of surface runoff, macropore flow and frozen soil. If the finally calculated infiltration is greater than zero, it is added to the upper layer soil water. This is done regardless of whether there is space in the soil pores there or not. If the water exceeds the water pore volume it is assumed to lie on the ground, but it still belongs to the upper soil layer, is totally mixed and thus has the same concentrations.

HYPE has an option for alternative calculation order of soil processes during a timestep. As default it calculates and add infiltration (and let the soil water percolate) before runoff and evaporation is calculated and removed from the soil water. Alternatively runoff and evapotranspiration is calculated before infiltration and percolation to slow the response of soil runoff. These options is tested during development of the soil routine.

## Diversion of surface runoff and macropore flow

There are alternative models for diversion of flow from water available for infiltration (*infilto*) to be chosen from. All model options result in a division of available water into three flow paths; surface runoff to the nearest stream, infiltration to top soil layer, and macropore flow to the soil layer where the groundwater table is currently located.

The original and default model is based on runoff coefficients and thresholds for infiltration and soil moisture. Surface runoff due to excess infiltration and macropore flow are calculated from the sum of snow melt and rainfall; the water available for infiltration (*infilto*).

If the current infiltration rate is greater than a threshold (*mactrinf*, mm/timestep) then macropore flow (*macroflow*) and surface runoff (*inoverflow*) may occur. In addition, the water in the upper soil layer needs to be larger than another threshold. This threshold is determined by a soil type dependent parameter (*mactrsm*) multiplied by the water not available for runoff (i.e. *fc+wp*). The two flows are calculated as a percentage (*macrate* respective *srrate*) of the infiltration above the first threshold;

$$macroflow = macrate \times (in\ filto - mactrinf)$$

$$in\ fover\ flow = srrate \times (in\ filto - mac\ trinf)$$

All the four aforementioned parameters are soil type dependent. If *macrate* and *srrate* together are greater than one, they are weighted so that their sum is one prior to calculation of the surface runoff and macropore flow;

$$macro\ flow = \left( \frac{macrate}{macrate + srrate} \right) \times (in\ filto - mac\ trinf)$$

$$in\ fover\ flow = \left( \frac{srrate}{macrate + srrate} \right) \times (in\ filto - mac\ trinf)$$

The actual infiltration is calculated by subtracting the macropore flow and surface runoff from the sum of snow melt and rain.

$$in\ filt = in\ filto - macro\ flow - in\ fover\ flow$$

An alternative flow diversion model calculates surface runoff and macroporeflow from a surface runoff coefficient based on current (*S*) and maximum soil moisture (*Smax*). The soil moisture can be based

$$S = SUM \left( soil(j) \right)_{j=1, nlayer}$$

on one, two or all three soil layers.  $Smax = SUM$  where *soil(j)* is soil water of soil layer *j* (mm), *pw(j)* is pore volume of soil layer *j* (mm) and calculated from the model parameters for water holding capacity (*wcwp*, *wcfc* and *wcep*), *nlayer* is set by a general model parameter (*srnlayer*). The surface runoff coefficient (*srratio*) is a "beta-function", with the general

$$srratio = \left( \frac{S}{Smax} \right)^{\beta}$$

parameter *srbeta* as exponent *beta*, but limited to one. A fraction of the diverted flow goes to macropore flow (*macroflow*), while the rest become surface runoff (*inflow*). The infiltration (*infiltration*) is the rest of the available water.  $macro\ flow = mac\ frac \times srratio \times in\ filto$

$$in\ fover\ flow = (1 - mac\ frac) \times srratio \times in\ filto \quad in\ filt = (1 - srratio) \times in\ filto$$

Another alternative flow diversion model calculates surface runoff and macroporeflow from a surface runoff coefficient based on current (*S*) and maximum soil moisture (*Smax*) and from current available water for infiltration. This model is similar to the one above but the exponent of the surface runoff

$$\beta = \frac{sralpha}{in\ filto^{sr\gamma}}$$

coefficient (*beta*) is calculated from two general parameters and *infiltration*.

### Additional infiltration limitation by frozen soil

An optional model for infiltration limitation and diversion of flow considers the effect of frozen soil. It is developed based on Zhao and Gray (1999) and tried by Stadnyk et al (2020). This model redirects all or part of the remaining infiltration, after calculating the diversion of surface runoff and macropore flow as described above. Note that this is not part of the frozen soil model option described above ([Soil temperature and frozen soil](#)).

If the minimum daily temperature is less than 10 degrees and the infiltration is larger than 5mm/d an ice lens is created in the soil. In this case, and as long as the maximum daily temperature is below zero, the ice lens redirect all infiltration to surface runoff and macropore flow.

$$redirect = in\ filt$$

If there is no ice lens, but the soil temperature of the upper soil layer (*soiltemp*) is below zero the infiltration is restricted but not blocked. The infiltration is restricted by a potential infiltration adapted from Zhao and Gray (1999). The potential infiltration (*potinfilt*) depends on a model parameter (*bfroznsol*) that is soil type dependent. It also depend on the "opportunity time" (*t0*), which is an estimate of the time with possible infiltration in hours;

$$potinfilt = \begin{cases} 0 & \text{for } soil \geq pw \\ \frac{bfroznsol \times 0.99^{2.92} \times \left(1 - \frac{soil}{pw}\right)^{1.64} \times \left(\frac{0 - soiltemp}{273.15}\right)^{-0.45} \times t_0^{0.44}}{\frac{t_0}{24}} & \text{for } soil < pw \end{cases}$$

$$t_0 = \begin{cases} 1 & \text{for } 0.65 \times snow < 6 \\ 0.65 \times snow - 5 & \text{for } 0.65 \times snow > 6 \end{cases}$$

Here *soil* is soil water of the upper soil layer (mm), *pw* is pore volume of the upper soil layer (mm) and calculated from the model parameters for water holding capacity (*wcwp*, *wcfc* and *wcep*), and *snow* is snow water equivalent (mm).

If the actual infiltration is greater than the potential infiltration then the overshoot is redirected to surface runoff and macropore flow.

$$redirect = in\ filt - potinfilt$$

The here redirected infiltration is added to the macropore flow and overland flow of the basic model proportionally to their respective model parameters (*macrate* and *srrate*).

$$macroflow = macroflow + \left( \frac{macrate}{macrate + srrate} \right) \times redirect$$

$$in\ fover\ flow = in\ fover\ flow + \left( \frac{srrate}{macrate + srrate} \right) \times redirect$$

## Percolation

The flow of water downward through the soil layers is only possible for water over field capacity (water in the effective porosity part of the pore volume). For a frozen soil, percolation is only acting on the liquid water of the soil. A maximum percolation (mm/d) limits the flow between soil layers. For the upper soil layer it is *mperc1*, and for the second soil layer it is *mperc2*. These parameters are soil type dependent. Flow is also limited by how much water the lower layer can receive.

Drainage from soil layer 1 to soil layer 2 is

$$perc1x = MIN \left( \left( soil(1) - wp(1) - fc(1) \right), mperc1 \right)$$

but if there is not enough capacity in soil layer 2 the drainage is instead

$$perc1 = wp(2) + fc(2) + ep(2) - soil(2) + perc2$$

and fills the second soil layer.

Drainage from soil layer 2 to soil layer 3 can be at most

$$perc2x = MIN \left( wp(3) + fc(3) + ep(3) - soil(3), mperc2 \right)$$

for that is what the soil layer 3 can receive. If soil layer 2 does not reach field capacity with *perc1x* added

$$perc2 = 0$$

but if soil layer 2 with *perc1x* added exceeds field capacity

$$perc2 = MIN \left( soil(2) + perc1 - wp(2) - fc(2), perc2x \right)$$

Regional groundwater flow is created by additional percolation from soil (see Section on Regional groundwater flow).

## Upwelling

Flow may enter the lowest soil layer, i.e. regional groundwater flow. Upwelling to soil layers above may occur if the soil layer is filled.

## Saturated surface runoff

Surface runoff due to a high ground water table (*q*, mm/time step) occurs when the water table in the upper soil layer reaches above the surface. It depends on a parameter *srrcs* which is dependent on land use. The recession parameter is corrected with the correction factor *rrcscorr* for different parameter regions (*parreg*). It is defined as an increase.

$$srrcs = srrcs \times (1 + rrcscorr)$$



$$q = MAX \left( srrcs \times \left( soil(1) - wp(1) - fc(1) - ep(1) \right), 0. \right)$$

Runoff is removed from the uppermost soil layer. The total surface runoff (due to high ground water table and low infiltration capacity) is calculated and printed.

## Macropore flow

Macropore flow occurs when the potential infiltration and water in the upper soil layer is large (see Diversion of surface runoff and macropore flow above). It is caused by a limited infiltration capacity of the soil. Macropore flow (*macroflow*) is added to the layer in which the water table is located (see Diagnostic variables above). The water is added to this layer only until it is full and the excess is trapped in the layer above.

Thus water does not flow up into the layer above when macropore flow is larger than the empty space in the soil layer with the water table, as in the case of groundwater inflow. Instead the excess flow stays in the soil layer above before reaching the soil layer of the water table. This distinction is important for the substances following the macropore flow.

## Links to file reference

Section	Symbol	Parameter/Data	File
Soil temperature and frozen soil	$T$		<a href="#">Tobs.txt</a>
	$deeptemp$	$init2$	<a href="#">par.txt</a>
	$par_{logsatmp}$ $par_{bcosby}$	$logsatmp, bcosby$	
		$deepmem, surfmem, depthrel, fzsexpand$	
	$pw$	$pw = wp + fc + ep$ and calculated from $wcwp, wcfc, wcep, wcwp1-3, wcfc1-3, wcep1-3$	
Groundwater runoff	$wp, fc, ep$	calculated from $wcwp, wcfc, wcep, wcwp1-3, wcfc1-3, wcep1-3$	<a href="#">par.txt</a>
		$rrcs1, rrcs2, rrcs3, rrcscorr$	<a href="#">GeoClass.txt</a>
		$soillayerdepth, streamdepth, tileddepth$	
	$soillayerthick$	calculated from $soillayerdepth$	
	$slope$	$slope\_mean$	<a href="#">GeoData.txt</a>
		$parreg$	
Runoff through drainage pipes		$trrcs$	<a href="#">par.txt</a>
		$tilegroup$	<a href="#">ClassData.txt</a>
	$tilefrac$	$tilefrac\_1, tilefrac\_2, ..., tilefrac\_10$	<a href="#">GeoData.txt</a>
Infiltration		$mactrinf, mactrsm, macrate, srrate, bfroznsoil, macfrac$	<a href="#">par.txt</a>
	$pw$	calculated from $wc + fc + wp$	
	$beta$	$srbeta$ or calculated from $sralpha, srgamma$	
	$nlayer$	$srnlayer$	
Percolation		$mperc1, mperc2$	<a href="#">par.txt</a>

Section	Symbol	Parameter/Data	File
Saturated surface runoff		<i>srrcs</i>	<a href="#">par.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures	Section
<a href="#">soil_processes (soil_proc.f90)</a>	calculate_soiltemp	soil temperature and frozen soil
	calculate_weigthed_temperature	
	calculate_unfrozen_soil_water	
	calculate_three_soil_temperature	
	calculate_liquid_water_fraction	
	initiate_soil_water	groundwater runoff
	calculate_soil_runoff	
	calculate_tile_drainage	runoff thorough drainage pipes
	calculate_infiltration_flow_diversion	diversion of surface runoff and macropore flow, infiltration
	add_infiltration	
	percolation	percolation
	add_macropore_flow	macropore flow
<a href="#">general_water_concentration (general_wc.f90)</a>	inflow_lowest_soillayer	upwelling

## Glaciers

The glacial class' area ( $^{class}area$ ) is divided into snowfield and glacier ( $^{glac}area$ ). The division may vary with simulation time, but the glacier cannot be larger than the slc class' area. At simulation start the area is considered to be all glacier, unless specific information on which year the class area is valid is given. The initial glacier volume is calculated based on glacier area or set for each glacier. If the initial year is given the initial volume is valid for that date. In this case, year of the area and/or volume and the average annual change in massbalans (*annmb* in m/yr) is given as input data or model parameters. The initial glacier volume is then adjusted according to the difference in years (*yeardiff*) between simulations start and glacier area data.

The glacier volume - area relationship:

$$^{glac}_{vol} = coe f \times ^{glac}_{area}^{exp}$$

$$coe f = coe fo \times e^c$$

The initial glacier volume if calculated from class area:

$$^{glac}_{vol} = coe f \times ^{class}_{area}^{exp} + yeardiff \times \frac{annmb \times ^{class}_{area}}{^{glac}_{dens} \times 1000}$$

The glacier area is correspondingly calculated from the glacier ice volume, but is limited by the class area. The relation is:

$$glac_{area} = \left( glac_{vol} \times \frac{1}{coef} \right)^{\frac{1}{exp}}$$

The equation coefficients *coef* and *exp* can have different values for specific glaciers. The first coefficient *coef* is calculated as the product of *EXP*(*c*), where *c* is a glacier volume correction, and a general parameter (*glacvcoef*/*glacvcoef1*) depending on glacier type. The second coefficient, *exp* is a general parameter (*glacvexp*/*glacvexp1*) depending on glacier type. Glacier density (*glacdens*) is a general model parameter (m<sup>3</sup> water / m<sup>3</sup> ice).

Glaciers are divided into four types. The default type is mountain glacier, the alternatives are ice cap, ice sheet and infinite glacier. Glacier type is given as input, or determined by the glacier area (a threshold (*glac2arlim*, a general parameter). The glacier area is used to determine the glacier type if it is not given as input and the threshold parameter is set. The glaciers will then be divided into mountain glaciers and ice caps.

The default glacier and the ice cap glacier type differ by having different parameter values for the volume-area relationship. The default values for the volume-area relationship are based on Radic and Hock (2010).

The ice sheet glacier do not simulate an snow field, and the glacier will have a constant area. The ice sheet glacier type is thus not depending on the volume-area relationship, except for initial volume if that is not given. In that case the initial volume of the ice sheet glacier will used the ice cap parameter values.

The infinite glacier type is independent on glacier volume, because the glacier melting is not limited by the glacier volume. It has a constant area (the class area) and do not simulate an snow field.

For snowfields, all soil processes are calculated as for the common land classes. Snow pack is only calculated for the snowfield, i.e. snow depth is assumed zero on glaciers. For glaciers, all precipitation is added to the glacier ice. Glacier melting is calculated as:

$$melt = cmlt \times (T - tmp)$$

where *cmlt* is the general parameter *glaccmlt* and *tmp* the general parameter *glacttmp*.

An alternative glacier melting model exist, which depend on class' temperature (*T*) and radiation (*swrad*) and has a refreezing component (goverened by general parameter *crefr*). The radiation component depend on the albedo of the glacier, which in turn depends on if the glacier is snow covered (*snowcov*) or not, and general parameters for rate (*cmrad*) and glacier ice albedo (*glacalb*). The refreezing component is limited to the glacier melt.

$$melt = \begin{cases} cmrad \times swrad \times (1 - albedo_{glacier}) + cmlt \times (T - tmp) & \text{for } T \geq tmp \\ cmrad \times swrad \times (1 - albedo_{glacier}) - crefr \times cmlt \times (tmp - T) & \text{for } T < tmp \end{cases}$$

$$albedo_{glacier} = albedo_{snow} \times snowcov + glacalb \times (1 - snowcov)$$

Glacier evaporation is optional and set by an model option (*snowevaporation*). The evaporation is determined as a fraction of the potential evaporation by the general model parameter *fepotglac*.

Glacier melting, snow melting on snowfield and/or rain are added to the soil. Thus the soil represents the whole glacier class area. The concentrations in the glacial melt water are zero. This means that any atmospheric deposition of nutrients is lost on the glacier.

## Links to file reference

Symbol	Parameter/Data	File
$class_{area}$	$slc\_nn, area$	<a href="#">GeoData.txt</a>
$annmb$	$annualmb$	<a href="#">GlacierData.txt</a>
	or $glacannmb$	<a href="#">par.txt</a>
$yeardiff$	calculated from $slcdate$	<a href="#">GlacierData.txt</a> or <a href="#">par.txt</a>
	and $bdate$	<a href="#">info.txt</a>
$c$	$logvolcorr$	<a href="#">GlacierData.txt</a>
	$glactype$	
$T$	see needed data in <a href="#">Links for temperature</a>	
$swrad$	calculated or from	<a href="#">SWobs.txt</a>
$coef$	$glacvcoef$ or $glacvcoef1$	<a href="#">par.txt</a>
$exp$	$glacvexp$ or $glacvexp1$	
$cmlt, ttmp$	$glaccmlt, glacttmp$	
$albedo_{snow}$	calculated from $snalbmin, snalbmax, snalbkexp$	
	$glacalb, glacdens, glac2arlim, fepotglac$	
$crefr, cmrad$	$glaccrefr, glaccmrad$	

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">glacier_soilmodel</a> ( <a href="#">glacier_soilmodel.f90</a> )	<code>soilmodel_3</code>
	<code>initiate_glacier</code>
	<code>initiate_glacier_state</code>
	<code>calculate_glacier_area</code>
	<code>get_glacier_parameters</code>

## References

Lindström, G., K. Bishop, and M. Ottosson Löfvenius, 2002. Soil frost and runoff at Svartberget, northern Sweden - measurements and model analysis, *Hydrological Processes*, 16:3379-3392.

Radic, V. and Hock, R., 2010. Regional and global volumes of glaciers derived from statistical upscaling of glacier inventory data, *J. Geophys. Res.*, 115, F01010, doi:10.1029/2009JF001373.

Samuelsson, P., S. Gollvik, and A. Ullerstig, 2006. The land-surface scheme of the Rossby Centre regional atmospheric climate model (RCA3), SMHI Report Meteorologi Nr 122, 25 pp.

Stadnyk, TA, MK MacDonald, A Tefs, SJ Déry, K Koenig, D Gustafsson, K Isberg, and B Arheimer. 2020. Hydrological modeling of freshwater discharge into Hudson Bay using HYPE. *Elem Sci Anth*, 8: 43. doi:10.1525/elementa.439

Zhao, L., and D.M. Gray 1999. Estimating snowmelt infiltration into frozen soils, *Hydrological*

---

Processes, 13:1827-1842.

# Rivers and lakes

## Basic assumptions

The HYPE model simulates most lakes and rivers in a conceptual way. The HYPE model can contain two types of rivers, local river and main river, and two types of lakes, local lakes and outlet lakes (Figure 1). The local river represent all ditches, streams, rivers within a subbasin that recieve only runoff within the subbasin. It is modelled as one conceptual river with a length, regardless of the actual distance runoff from different land has to travel through the subbasin. Similarly a local lake represents all lakes within a subbasin receiving inflow only from within the subbasin. The modelled local lake recieves a fraction of the flow in the local river. The fraction is defined as a fraction of the subbasin area. The main river is the river part(s) of a subbasin that water from upstream subbasins flows through. Depending on subbasin delieation it may be clearly defined or represent several tributaries that comes to the subbasin outlet. Finally the modelled outlet lake represents a lake located at the outlet of a subbasin.

Local and main rivers are present in all subbasins and the length of each watercourse is calculated as the square root of the subbasin area, unless it is given as input. The river can be a SLC class and has then an surface area, but rivers can also be one-dimensional (i.e. no fraction of the subbasin area are occupied by the river and no precipitation added to the river). All runoff enters the local river. Local lakes (ilake) receive a portion of the local flow. The flow leaving the local river and the outflow from local lake, goes to the main river of the same subbasin. If there are upstream subbasins their flow is added to the main river together with the local flow. Outlet lakes (olake) receive the outflow from the main river, i.e. all upstream and local flows.

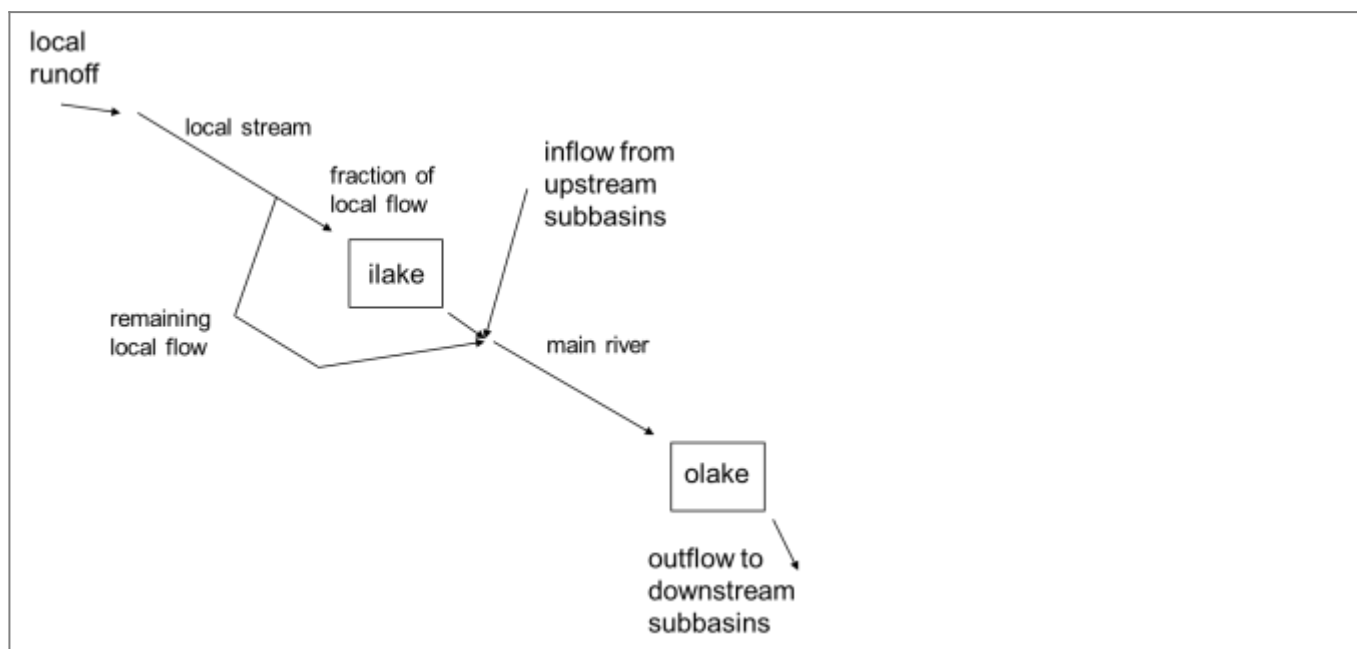


Figure 1: Schematic representation of streams and lakes in a subbasin of HYPE, and the links between them.

The two lake types are separate classes. The lake classes have characteristics such as land use and soil type, which are defined together with the other classes' characteristics (in GeoClass.txt). Precipitation, atmospheric deposition and evaporation of rivers and lakes are calculated first, while river flow and inflow, transformation processes and the outflow of the lakes is calculated thereafter.

Lakes and rivers are calculated in the model's routing part after all classes are calculated for the subbasin.

An outlet lake can be part of a larger lake. It is then called a lake basin. Lake basins are lakes in nearby subbasins.

A simple outlet lake has a threshold. If the water level drops below the threshold, the outflow ceases. Lake mean depth below the threshold is specified in GeoData.txt or LakeData.txt as *lake\_depth* in meters. Lake depth can also be set by parameters, i.e. general parameter *gldepo* or lake region parameter *olldepth*. At the start of a simulation the water depth of the lake is set to be at the threshold. The current water depth, denoted *wlm* in Fig. 2, can be above or below the threshold.

For output, the outlet lake water level (output variable *wcom*) is given in a user set reference system. The reference level is used to get the same height system as any observations of the lake's water level. It uses the unit of meter, and the reference level (*w0ref*) gives the level of the threshold in the reference system (Fig. 2). The output *wcom* is calculated as the water level above the threshold plus *w0ref*. A regulated lake (dam) has two thresholds. The distance between the thresholds are determined by the regulation volume (*regvol*) of the dam with the assumption that the dam has constant area with depth. In reality the area may decrease with depth. This means that the the observed variation in water level may be larger than the simulated variation. It is therefore possible to adjust the output *wcom* (and *wcav*) with the actual amplitude of the regulation volume (*wamp*). This will make the simulated and recorded water stage comparable below the upper threshold for a regulated lake.

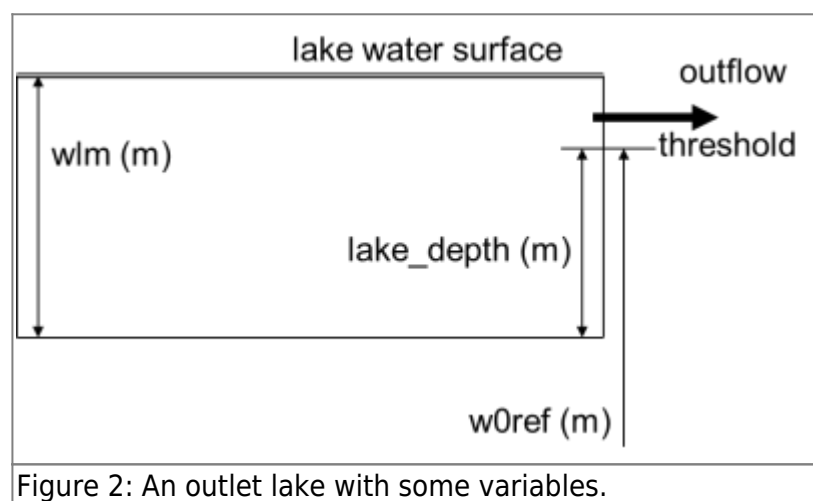


Figure 2: An outlet lake with some variables.

A local lake also has a threshold depth that is used as start value. The depth is given by general parameter *gldepi* and is then the same for all the local lakes, or by lake region parameter *illdepth*. It is measured in meters. A percentage of flow from the local stream flows into the local lake. The rest of the local flow runs directly to main river watercourse.

The streams may have a delay and an attenuation of the flow. The former shifts the peaks, but leaves them otherwise untouched, while the attenuation both delays and smooths out peaks. The streams have a dead volume in which the residence time of solutes is increased relative to the water and the concentration smoothed out further over time.

## Links to file reference

Parameter/Data	File
<i>rivlen, loc_rivlen, slc_nn</i>	<a href="#">GeoData.txt</a>
special class code 1,2,11 and 12	<a href="#">GeoClass.txt</a>
<i>lake_depth</i>	<a href="#">GeoData.txt</a> or <a href="#">LakeData.txt</a> or <a href="#">DamData.txt</a>
<i>gldepo, olldepth, gldepi, illdepth</i>	<a href="#">par.txt</a>
<i>w0ref, regvol, wamp</i>	<a href="#">LakeData.txt</a> or <a href="#">DamData.txt</a>

## Links to relevant modules in the code

Modules (file)	Procedures
<a href="#">modelmodule (model_hype.f90)</a>	initiate_model_state
	initiate_model
	calculate_flow_from_undivided_lake
	calculate_flow_for_lakebasin_lake
<a href="#">surfacewater_processes (sw_proc.f90)</a>	calculate_riverlength
	calculate_ilake_outflow
	calculate_outflow_from_outlet_lake
	calculate_outflow_from_lakebasin_lake
	remove_outflow_from_lake
	calculate_branched_flow

## Rivers

### Common rivers processes

#### Precipitation

Daily precipitation is added to the river if it has an area (is a class) and a new concentration is calculated. Precipitation is divided between river water in damping box and queue according to the respective volumes.

#### Evaporation

If the river has an area (is a class), it evaporates and a new concentration is calculated. The river is assumed to evaporate at potential evaporation rate (see [Processes above ground - Evaporation](#)). Normally the river area is constant over time, but with parameters a reduction of riverarea can be simulated for low volume/flow. The reduced river area is also used for heat exchange calculations.



$$frac_{area} = \begin{cases} 0 & x \leq 0 \\ \left(1 + \frac{(frac_{xe} - x)}{(frac_{xe} - frac_{xm})}\right) \times \left(\frac{x}{frac_{xe}}\right)^{\frac{frac_{xe}}{(frac_{xe} - frac_{xm})}} & 0 < x < frac_{xe} \\ 1 & x \geq frac_{xe} \end{cases}$$

The parameters *frac<sub>xe</sub>* and *frac<sub>xm</sub>* are general, and *frac<sub>xe</sub>* is (minimum) mean river depth (m) where fractional river area = 1 and *frac<sub>xm</sub>* is mean river depth (m) where the slope of the fractional river area has its maximum (must be in the range between 0 and *frac<sub>xe</sub>*). *x* is the current mean river depth (based on full area extension).

### Pure delay

The delay in the watercourse (*transtime*) in days is determined by the length of the watercourse (*rivlen*) and the water's maximum velocity (*rivvel*). The maximum velocity is a general parameter with unit *m/s*. The delay in the river is dependent on subbasin land area if the default river length is used. The delay is a pure translation unless the parameter *damp* is set. The pure delay is divided into whole days (*ttday*) and parts of the day (*ttpart*) (if the time step is daily, other wise the division is made into whole and fractions of that time step). In the following description daily time step is assumed.

$$transtime = \frac{rivlen}{rivvel \times 8.64 \times 10^4}$$

$$ttday = INT(transtime)$$

$$ttpart = transtime - ttday$$

The inflow of the river is stored in two arrays (*riverq* and *riverc*) until it is time for it to flow out of the river stretch. The outflow is weighted by using the parts of the time step (*ttpart*) that are to flow out during the time step.

```
transq = (1-ttpart)*riverq(ttday) + ttpart*riverq(ttday+1)
IF(transq>0)THEN
  transc = ((1-ttpart)*riverq(ttday)*riverc(ttday) +
            ttpart*riverq(ttday+1)*riverc(ttday+1)) / transq
ELSE
  transc = 0.
ENDIF
```

After the calculation of outflow the values in the arrays are shifted forward one time step.

## Delay and attenuation

If the delay in the watercourse also includes a damping of the peaks then only part of the delay is considered as translation, while the rest of the delay lies in damping. The translation is calculated first and then the flow goes through a linear box that creates attenuation. The parameter *damp* indicates how much of the delay that will occur in the attenuation box (or damping box), and translation time is reduced accordingly. Otherwise the translation is calculated in the same manner as above.

$$totaltime = \frac{riverlen}{rivervel \times 8.64 \times 10^4}$$

$$transtime = (1 - damp) \times totaltime$$

The result from the translation of water (*transq* and *transc*) flows into the attenuation box, which is assumed to be completely mixed. The delay time in the box (*kt*, in timesteps) is determined from input and parameters.

$$kt = damp \times totaltime$$

The outflow equation depends on the attenuation box's volume before changes during the time step (*riverbox<sub>ini</sub>*) and on the net inflow during the timestep (*inflow*). The net inflow includes the outflow from the translation (local runoff, upstream inflow, point sources etc.), removal through irrigation or regional groundwater flow, and the flow to or from a main river floodplain. The equation has two coefficients:

$$riverrc1 = 1 - kt + kt \times e^{-1/kt}$$

$$riverrc2 = 1 - e^{-1/kt}$$

$$dampq = riverrc1 \times inflow + riverrc2 \times riverbox_{ini}$$

An earlier version of calculation of outflow from the box can be accessed through a model option. In this case the delay time of the attenuation box is recalculated to a recession coefficient (*riverrc*). The recession coefficient is used to calculate the outflow from the box (*dampq*) by the current volume in the box (*riverbox*).

$$riverrc = 1 - kt + kt \times e^{-1/kt}$$

$$dampq = riverrc \times riverbox$$

$$\text{where } riverbox = riverbox_{ini} + inflow$$

## Additional smoothing by dead volume

A dead volume (*deadriver*,  $m^3$ ) can be used to get additional equalization of concentrations in the watercourse. In this case, the initial value of the attenuation box is equal to the dead volume. Inflows into the box are mixed with the entire volume. At present this represents a dilution of the

concentration since the initial value of concentration in the box is zero. In HYPE the dead volume is estimated from the upstream area (including the area of the subbasin itself). This gives the dead volume:

$$deadriver = dead \times upparea \times rivlen$$

where the parameter *dead* (*deadl* or *deadm*,  $m^2 / km^2$ ) is general parameters, the upstream area *uparea* ( $km^2$ ) is calculated from input data, and the river length (*rivlen*, *m*) is estimated to be the root of the subbasin's area unless it is set among the input data. Dead volume can be used without suppression of water flow peak (*damp* = 0). When additional smoothing is used, the *riverbox* in the attenuation equations above is replaced by the volume of the attenuation box minus the dead volume.

In addition, the dead volume is used to calculate a minimum river width used for calculation of river surface area in the case the river area is not given as a class area. The width is assumed to be 10 times the depth and together with the dead volume and river length, the with is calculated as:

$$deadwidth = 10 \times \left( \frac{deadriver}{rivlen \times 10} \right)^{\frac{1}{2}}$$

### Derivation of the equation of recession in the attenuation box

The continuous-time equation for the change in volume *S* in a reservoir is:

$$\frac{dS}{dt} = i(t) - q(t)$$

where *i(t)* is influx and *q(t)* is outflow. If we assume that the outflow is proportional to the volume we get another expression for the change in *S*. *k* is in the unit of time.

$$q(t) = \frac{1}{k} S,$$

$$\frac{dS}{dt} = k \frac{dq}{dt}$$

Combining these two equations gives a first degree equation that can be solved using the method of "integrating factor":

$$\frac{dq}{dt} + \frac{1}{k} q = \frac{i}{k},$$

$$\frac{de^{t/k} q}{dt} = \frac{i}{k} e^{t/k},$$

$$q(t) = \frac{1}{k} \int i(\tau) e^{\tau-t/k} d\tau$$

In our case, we have a constant influx during a time steps of length 1 day. Assume that the influx

during a time steps are  $I$  and the outflow  $Q$ . These flows are thus the equivalent discrete flows.

Because  $I$  is constant during the time step ( $i(t) = I$ )  $Q$  can be calculated. Define  $y(t) = e^{\frac{t}{k}} q(t)$  and integrate the second equation above, from 0 to  $t$  ( $t = 1$ , one time step).

$$\int_{y(0)}^{y(t)} \frac{dy}{dt} d\tau = \int_0^t \frac{i(\tau)}{k} e^{\tau/k} d\tau,$$

$$y(t) - y(0) = \frac{I}{k} \int_0^t e^{\tau/k} d\tau = I \left( e^{t/k} - 1 \right)$$

$$e^{t/k} q(t) - q(0) = I \left( e^{t/k} - 1 \right)$$

$$q(t) = q(0) e^{-t/k} + I \left( 1 - e^{-t/k} \right)$$

Calculate  $Q$  for time step (the mean of  $q(t)$ ) and replace the outflow with the volume.

$$Q = \frac{\int_0^1 q(t) dt}{\int_0^1 dt} = \int_0^1 q(0) e^{-t/k} + I \left( 1 - e^{-t/k} \right) dt$$

$$= q(0) \int_0^1 e^{-t/k} dt + I \int_0^1 1 - e^{-t/k} dt$$

$$= q(0) \times \left( -k \left( e^{-1/k} - 1 \right) \right) + I(1-0) - I \left( -k \left( e^{-1/k} - 1 \right) \right)$$

$$= I + \left( e^{-1/k} - 1 \right) \left( kI - kq(0) \right)$$

$$Q = I + \left( e^{-1/k} - 1 \right) \left( kI - S(0) \right)$$

or separated for  $I$  and  $S$ .

$$Q = \left(1 - k + k \times e^{-1/k}\right) I + \left(1 - e^{-1/k}\right) S(0)$$

Common in HYPE is that outflows are calculated with a recession coefficient. Here the recession coefficient is defined as the proportion of outflow after an inflow pulse. It is therefore calculated as  $r = Q/I$  under the assumption that the starting volume can be neglected,  $S(0)=0$ .

$$r = \frac{Q}{I} = 1 + \left(e^{-1/k} - 1\right) \times k$$

For a real flow the volume can of course not be neglected in this way and this relationship is only approximate.

Alternatively the outflow may be calculated from the equation separated for  $I$  and  $S$ . For this option  $I$  is calculated as the sum of several different types of inflow and outflow to the river during the time step, e.g. precipitation and evaporation, in addition to the lateral inflow. In addition the initial value of the river volume ( $S(0)$ ) is saved here to be used for the outflow calculation.

### Bank-full flow

Bank-full flow is used for erosion of particulate phosphorus in rivers (see [Sedimentation/Resuspension](#)). It is approximated by the second highest daily flow during the last year.

### Local river

The local river has a length equal to the square root of the subbasin area, if not specified as input. Runoff from the land area of the subbasin forms the inflow to the local river (Figure 3). The flow in the local river is delayed and attenuated as described above. Of the resulting flow from the stream a constant percentage goes to the local lake (*icatch*), and the rest directly to the main river.

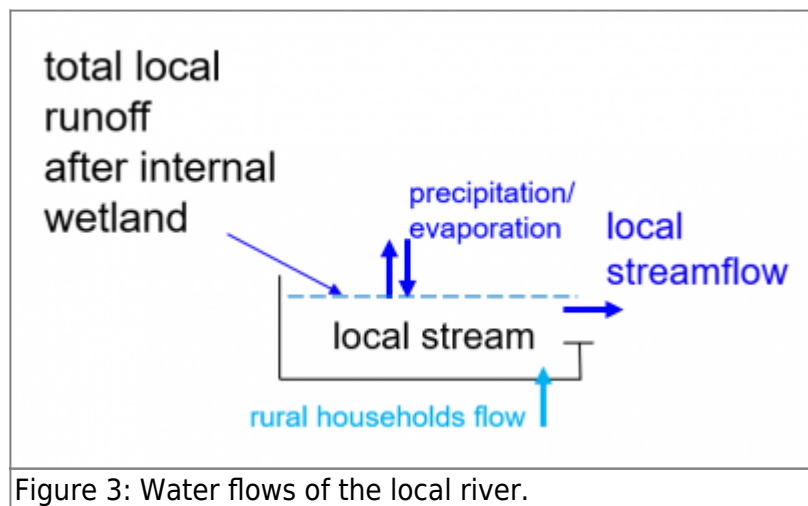


Figure 3: Water flows of the local river.

## Main river

A main river is present in all subbasins. The length is equal to the square root of the subbasin area, if it is not specified as input. In areas without incoming water from upstream there is still a main river, but it receives only local river flow after the local lake. In subbasins with upstream incoming water, the flow to the main river will be the sum of outflow from the local lake, the proportion of flow in the local river not flowing into the local lake and the water from upstream (Figure 4). For a subbasin with a (part of) a multi-basin lake, the main river transports the flow from land to the lakebasin of the subbasin.

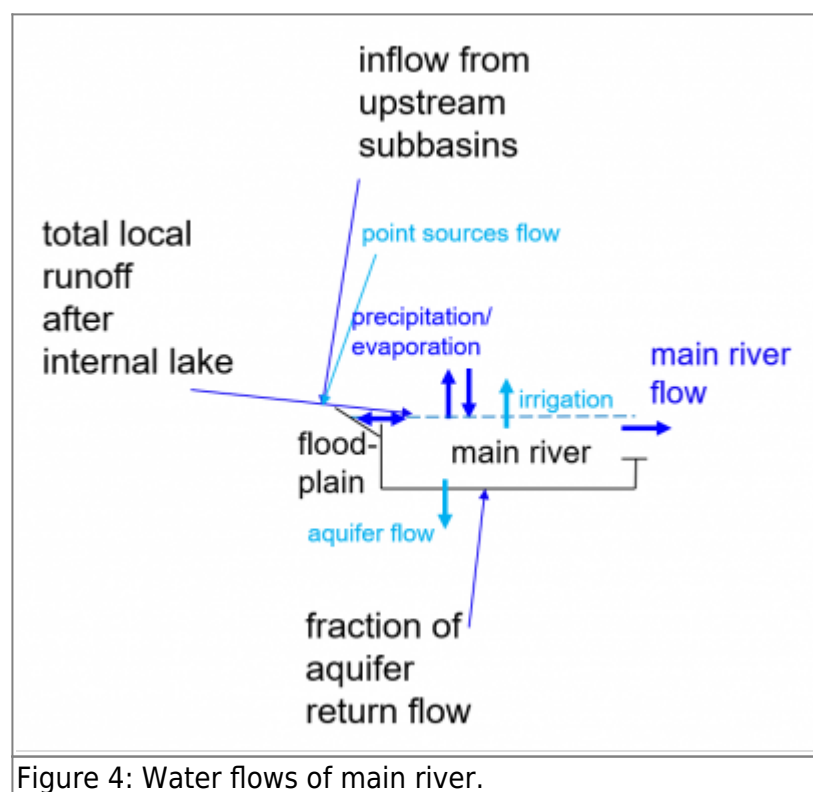


Figure 4: Water flows of main river.

The return flow from an aquifer is added to the inflow of the main river (Figure 4).

The flow in the main river is delayed and attenuated as described above. The resulting effluent from the river flows to the outlet lake or to the next subbasin if there is no outlet lake.

If the main river length is zero, the inflows to the main river is added to form the main river outflow.

## Inflow from upstream subbasins

In input files it is given to which subbasin(s) the outflow from each subbasin flows. The upstream flow enters the main river of the downstream subbasin. Inflow into the main river of a subbasin is calculated by adding outflows from upstream areas. Concentrations are flow-weighted by their relative share. The upstream flow enters the main river, except if the upstream subbasin has a lakebasin that is part of the downstream subbasin's lake. In this case the upstream lakebasin outflow goes directly into the downstream lakebasin.

## Hydraulic geometry to calculate water depth and velocity

River diagnostic variables can be calculated for the main river. A river water depth ( $d$ ) and river velocity ( $u$ ) can be calculated from hydraulic geometry for the main river and the calculated main river flow ( $q$ ). In the case of updating of flow in a subbasin with no outlet lake, the updated (main river) flow is used. The equation coefficients are general parameters ( $c, f, k, m$ ).

$$d = c \times q^f$$

$$u = k \times q^m$$

## Rating curve to calculate river water level

Given a continuous monotonous increasing equation for calculating water level from flow, the water level of a river ( $wr$ ) can be calculated from the flow by the inverse of this rating curve. This method is used to calculate main river water level in HYPE. Three different types of river rating curve formulations can be used (*TYPEofRRC*).

The first is a single rating curve with a threshold ( $w0$ ) and the rating curve coefficients are input. The rating curve

$$q = k \times (wr - w0)^p$$

give coefficients  $a$  and  $b$  for the inverse equation:

$$a = \frac{1}{k}^{\frac{1}{p}} \quad b = \frac{1}{p}$$

$$wr = a \times q^b + w0$$

The second type uses two rating curves with the same threshold ( $w0$ ); one for ice free condition and one for use when there is ice on the river. The transition between the rating curves is smoothed by scaling of the curves in a transition period. The scaling of rating curve parameters are determined by ice porosity and ice thickness (general model parameters). Between thresholds the influence of the ice curve increases linearly from zero to one. The minimum influence (of porosity and thickness) is used as the scaling weight of the winter curve during the transition period. If only parameters for the ice free curve are given as input, the same parameters is used for ice conditions except for the exponent ( $p$ ) which is then set to one.

The third type is for a sectorial rating curve. It has different rating curve coefficients for different sections of water level. The sections may not overlap or leave gaps in water level, even though small discrepancies is handled. The parameters and sections need to be chosen so that the whole curve is continuous. Since the rating curves are used inverted in the code, the sections are turned into sectors of flow. The average of the flow given by the minimum water level of one curve and the maximum water level of another curve is taken as the actual limit between the two curves. This type does not handle ice curves.

In addition to river water level described above, it is possible to calculate the water level in the scale





Modules (file)	Procedures	Sections
<a href="#">surfacewater_processes (sw_proc.f90)</a>	add_precipitation_to_river	precipitation
	calculate_river_evaporation	evaporation
	translation_in_river	pure delay
	calc_qbank	bank-full flow
	update_qbank	
	river_water_level	main river
	local_water_level	
	ice_on_river	

## Lakes

### Common lake processes

#### Precipitation

Daily precipitation is added to the lake and a new concentration is calculated. Lakes are assumed to be completely mixed.

#### Evaporation

The lake is assumed to evaporate at potential evaporation rate (see [Processes above ground - Evaporation](#)) and a new concentration calculated. The actual evaporating area is reduced if the lake is (partly) ice covered. Ice is calculated in case water temperature (substance T2) is simulated.

#### Outflow with rating curve

The rating curve is used for calculation of outflow in different circumstances. It is based on water level above a threshold.

$$q(t) = k \times \left( w(t) - w_0 \right)^p$$

In the program the equation is solved for the average flow during the timestep with linearization (Lindström, 2016). For rating curve exponent equal to one, the solution is exact, while for other exponent an approximation is used. The outflow of the lake is limited to the water above the threshold, in the case that the rating curve give a higher value.

#### Local lake (ilake)

The local lake has an initial volume which is determined by its area and threshold depth. The lake depth below the threshold is determined by model parameters (*gldepi* or *illdepth*).

The inflow to the lake is a percentage of the flow in the local stream (*icatch*). The percentage is determined by the percentage of the subbasin area that drains to the lake. This percentage can be given as a fraction in GeoData.txt for each subbasin with an internal lake, or be given by a regional parameter *ilicatch*, or be given by a general parameter *gicatch*. If not set at all the default value is 1, i.e. the local river runs through the local lake. The flow from the local river is added to the lake (Figure 6). The lake water is assumed completely mixed.

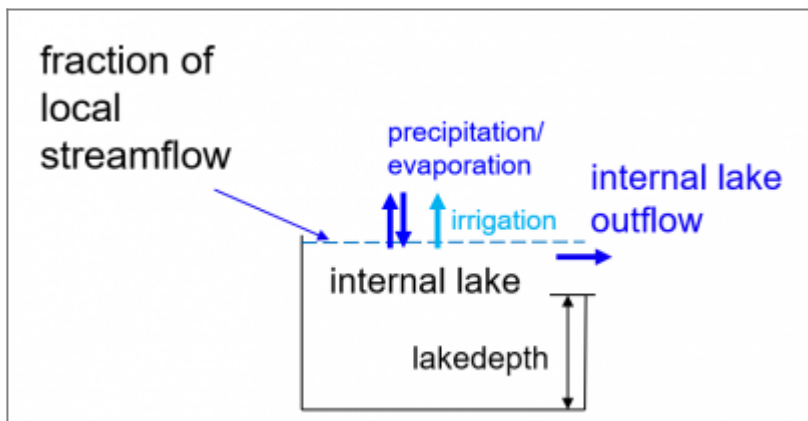


Figure 6: Water flows of internal lake.

Water outflow is calculated with the universal rating curve using general or region specific model parameters.

There is an alternative model for local lakes (a model option), the ilake connectivity model. The ilake connectivity model simulates the ilake as a chain of small connected lake sections (*lks*) with different height of their outflow threshold. Depending on water volume in the ilake one or several of the lake sections can be disconnected from the chain (zero outflow with water level below the threshold). The precipitation and evaporation are divided by the lake sections depending on their fraction of the ilake area, but the runoff from land area is divided by the lake sections depending on their fraction of the catchment area. In case the water level of a lake section is above its threshold, its outflow is calculated. The outflow of each lake section is calculated using the universal rating curve using general or region specific model parameters same as a simple ilake. The outflow of a lake section is of course also added to the receiving lake section before that one's outflow is calculated. Depending on how far upstream from the last lake section of the chain there is inflow from the above lake section, that fraction of the ilake area is the ilake's current connectivity. The catchment area of the lake sections above this border is the currently non-contributing catchment area.

### Simple outlet lake or dam (olake)

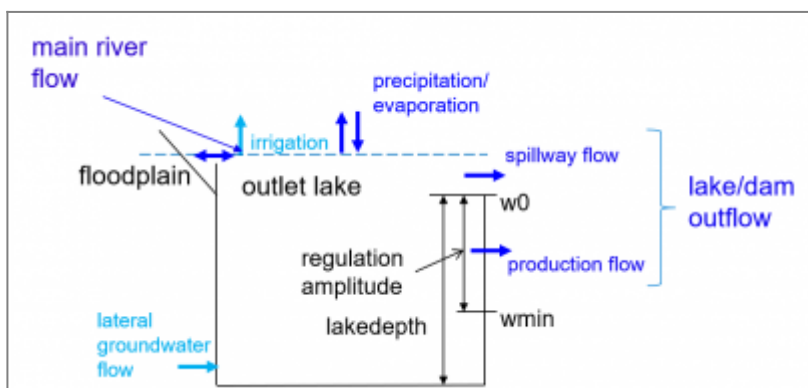


Figure 7: Water flows of outlet lake.

## Inflow

The outlet lake receives both local runoff and inflow from upstream areas via the main river. An outlet lake may receive inflow from regional groundwater flow (Figure 7).

## Outflow with universal rating curve

If the water level ( $wlm$ ) is higher than the threshold ( $lake\_depth$ ) the outflow is calculated with the following equation:

$$out\ flow = \begin{cases} 0 & ,\ for\ wlm < lake\_depth \\ gratk \times uparea^{grata} \times corr \times (wlm - lake\_depth)^{gratp} & ,\ for\ grata > 0 \\ gratk \times corr \times (wlm - lake\_depth)^{gratp} & ,\ for\ grata = 0 \end{cases}$$

$$corr = 1 + ratcorr$$

$gratk$ ,  $gratp$  and  $grata$  are general parameters that apply to all lakes in the model application.  $ratcorr$  is parameter region dependent model parameter for adjusting  $gratk$  to different regions. The upstream area ( $uparea$ ,  $km^2$ ) is included in the equation if parameter  $grata$  is  $>0$ . If the water level is below the threshold then the outflow is zero. If the flow calculated by the equation gives a larger outflow volume during a time step than the water volume above the threshold, the outflow is limited so that the water level do not drop below the threshold.

## Outflow with specific rating curve

In LakeData.txt a specific rating curve may be set instead of the general parameters. This is done by setting the  $rate$  and  $exp$  larger than zero in LakeData.txt. The outflow is also in this case limited by the threshold not to cause the water level to drop below it.

$$out\ flow = rate \times (wlm - lake\_depth)^{exp}$$

## Regulated lake

A simple regulation routine can be given in LakeData.txt for selected outlet lakes. There are two main versions of regulations. The first determines a (often constant) production flow, for water levels between the lake threshold and a lower water stage threshold, and a spill for higher water levels. The second determines the outflow by different rating curves for (two) different seasons.

## Production flow

The regulated outflow is independent of water level ( $wlm$  (m)) between the threshold ( $w0=lake\_depth$ ) and a minimum water level ( $wmin$ ). For this interval the production flow ( $qprod$ ) is used. Production flow may depend on time of year and the water level. The minimum water stage is not really a water stage, but is calculated from the regulation volume ( $regvol$ ) and the lake (surface) area. Thus it does not consider the area changing with depth.

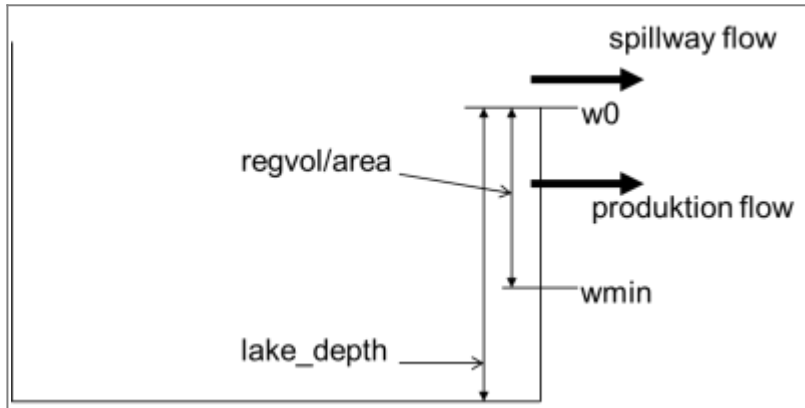


Figure 8: Illustration of regulation flow and variables.

For water levels above the threshold, the flow is calculated with a rating curve (typically flow through spillways) or all the water above the threshold is discharged. In this case though, the calculated flow may be increased so that the outflow is at least equal to the production flow ( $qprod$ ). Compiled together as one equation, the outflow from a regulated lake is:

$$out\ flow = \begin{cases} MAX \left( rate \times (wlm - w0)^{exp}, qprod \right) & , \text{ for } wlm > w0, rate > 0 \\ MAX \left( \frac{(wlm - w0) \times 1000}{qunit\ factor}, qprod \right) & , \text{ for } wlm > w0, rate \leq 0 \\ MIN \left( qprod, \frac{(wlm - wmin) \times 1000}{qunit\ factor} \right) & , \text{ for } wmin < wlm < w0 \\ 0 & , \text{ for } wlm < wmin \end{cases}$$

Input variables  $rate$ ,  $exp$ ,  $qprod$ ,  $regvol$ ,  $w0$ ,  $qamp$  and  $qpha$  can be found in LakeData.txt. The variable  $wmin$  is calculated by the program from  $regvol$  and lake area:

$$wmin = w0 - \frac{regvol \times 1000000}{area}$$

Production flow can have two different values during the year, which depends on the day of the year. This is determined by the input variables  $qprod1$ ,  $qprod2$ ,  $datum1$  and  $datum2$ . Regulation period 1 between  $datum1$  and  $datum2$  has production flow  $qprod1$ , while the rest of the year has production

flow  $qprod2$ . Not setting the dates gives the same production flow the whole year ( $qprod1$ ). Production flow can alternatively be made to vary sinusoidal over the years, with a peak in December, when power output is normally high, and a minimum in June. This is done with the input variable  $qamp$ . If you want a different seasonal variation set  $qpha$  (default = 102).

$$qprod = qprod \times \left( 1. + qamp \times \sin \left( \frac{2 \times \pi \times (dayno + qpha)}{365} \right) \right)$$

In addition, the production flow may be reduced when there is a low water level in the dam. The flow will then be reduced linearly from full production flow for that time of year when water level is over the limit ( $limqprod$  (percentage of volume)) to zero when no water is left in the dam over  $wmin$  level.

### Two rating curves

Regulated outflow can also be calculated from different rating curves for the regulation periods. The rating curve parameters are the specific rate and exponent for the lake, or if lacking, the general rate and exponent parameters.

$$q(t) = k \times (w(t) - w_0)^p$$

Rating curve threshold ( $w_0$ ) determined by input variable  $w0ref$  is used for regulation period 1, while the rating curve threshold is changed for regulation period 2. The shift is determined by input variable  $deltaw0$ :

$$w_0 = w0ref + deltaw_0$$

The shift between regulation period 1 and 2 can be made smoother by allowing the threshold to increase/decrease linearly over a few days. The number of days is given by the parameter  $w0adjdays$ .

### Dams of specific purpose

In addition to regulated lakes (described above), dams in HYPE can instead be regulated by a specific purpose. Dams are then governed by the main purpose of the dam, e.g. hydropower, flood control. The rules of calculating the outflow of the dam then depends on the specified purpose of each dam. The outlet lakes that are regulated by a specified purpose is given in DamData.txt. A dam without defined purpose may use the methods given in previous sections (and be defined in LakeData.txt).

Dams are characterised by their regulation volume ( $regvol$ ) between a minimum water stage ( $wmin$ ) and the spill threshold ( $w_0$ ). The minimum water stage ( $wmin$ ) is not a real a water stage, but is calculated from the regulation volume ( $regvol$ ) and the dam (surface) area. Thus it does not consider the area changing with depth. Below the minimum water level no outflow occurs. The spill threshold is defined from the lake's given depth ( $lake\_depth$ )

For dams a preferred production flow is calculated (see below). This is released if the dam is within the regulation volume, but no more water is released than to keep the water level above the minimum threshold. If instead the water level of the dam is above the threshold  $w_0$ , the spillway equation (rating curve) is used. The outflow is the maximum of the spillway equation and the preferred production flow.

Input variables *rate*, *exp*, *regvol*, *qamp* and *qpha* is given in DamData.txt.

### General preferred production flow

For all dams the general preferred production flow is calculated from input data on the dam. The flow that is preferred to be released from the dam today is then adjusted depending on purpose. The general production flow (*qprod*) may be constant or vary between two values per year. Typically it is estimated from the yearly average inflow of the dam, which is given as input data. It can alternatively be determined by input variables *qprod1*, *qprod2*, *datum1* and *datum2*, which then has to be estimated. Regulation period 1 between *datum1* and *datum2* has production flow *qprod1*, while the rest of the year has production flow *qprod2*. Not setting the dates gives the same production flow the whole year (*qprod1*). If *qprod1* is not given *qprod* is estimated from the yearly average inflow of the dam, which is given as input data.

The production flow may be reduced when the water level is low in the dam. The flow will be reduced linearly from full production flow for that time of year when water level is at the limit (*limqprod* (percentage of volume)) to zero when no water is left in the dam over *wmin* level.

### 1 - Irrigation dam and 2 - Water supply dam

Dams of these purposes use the general preferred production flow as it is.

### 3 - Flood control dam

The flood control dam will try to stay empty and ready to delay large inflows. The preferred production flow is calculated from the inflow of the day (*Qinf* today) for low inflow and water stages, but set higher than the inflow for high flows and water levels. The maximum flow (*qthresh*, the threshold flow) is determined from input data (*Qinf* is average inflow to dam for each month) and a general parameter (*kthr*), while the water level threshold (*wthresh*) is determined by a general parameter (*klow*) for the fraction of the regulation depth.

$$qthresh = kthr \times \text{MAX} (Qinf)$$

$$wthresh = w_0 - klow \times (w_0 - wmin)$$

If the current inflow is larger than the threshold inflow (*qthresh*), the maximum allowable discharge is released:

$$qprod = qthresh$$

If the current inflow is less than the threshold inflow (*qthresh*) and the current water level is lower

than the threshold level (*wthresh*), the inflow is released from the dam:

$$qprod = Qinftoday$$

If the water level is above the threshold (*wthresh*), more water is released than inflow in an attempt to empty the dam. The factor multiplied by inflow (*krel*) is a general parameter:

$$qprod = MIN \left( krel \times Qinftoday, qthresh \right)$$

#### 4 - Hydropower dam

Dams for hydropower uses the general preferred production flow, but the flow may be adjusted with a seasonal factor. The factor is sinusoidal over the year, with a peak in December, when power output is normally high (in northern Europe), and a minimum in June. For a different seasonal phase, set the input *qpha* to another value (default = 102). The size of the seasonal adjustment is set with the input variable *qamp* ( $0 < qamp < 1$ ) for each dam.

$$qprod = qprod \times \left( 1 + qamp \times \sin \left( \frac{2 \times \pi \times (daymo + qpha)}{365} \right) \right)$$

Alternatively the seasonal variation adjustment can be determined by the climate. If this option is used *qamp* is set to zero (or left out) in the input file, and the value used in the calculations is determined by the fraction of snow fall (*snowfrac*) for each subbasin/dam. If the snow fall fraction is above 0.35, a *qamp* of 0.71 is used.

### Outlet lake with two outlets

An outlet lake that is not part of a lake composed of lakebasins, may have two defined outlets in [LakeData.txt](#). The outlets can be divided into different outlet types depending on method for determining the outflow. HYPE will define an outlet's type based on variables present in [LakeData.txt](#). Below the defining (necessary) variables are given for each outlet type. Note: The threshold (*w0ref*) is given for first outlet. For the second outlet this variable instead is given relative to the threshold of outlet 1 (i.e usually zero).

#### Outlet types

1. An outlet with outflow as production flow (only). The production flow can be determined by constants for different seasons or with a sinus adjustment; see **Production flow** above. Defining variables in [LakeData.txt](#): *regvol*, (*rate*=0, *maxQprod*=0, *minflow*=0)
2. An outlet with outflow as production flow (same as type 1), but with an allowed (higher) maximum production flow. If the outflow from the other outlet is above zero, part of that flow will be sent through this outlet as part of production flow instead. Defining variables in [LakeData.txt](#): *regvol*,

*maxQprod*.

3. Rating curve determined outflow with constant threshold. Defining variables in LakeData.txt: *rate* (*deltaw0=0*, *regvol=0*, *maxQprod=0*)

4. Rating curve determined outflow with constant threshold relative to the threshold of the first outlet. This type is only used for the second outlet. Defining variables in LakeData.txt: *rate*, *w0ref* (*regvol=0*, *deltaw0=0*)

5. Flow determined by production flow below the threshold, and by a rating curve above the threshold. This could be production and overflow from a hydropower plant that goes too the same downstream subbasin. Defining variables in LakeData.txt: *regvol*, *rate* (*maxQprod=0*, *minflow=0*)

6. An outlet with outflow as production flow determined by different rating curves for different seasons; see **Two rating curves** above. Defining variables in LakeData.txt: *rate*, *deltaw0* (*regvol=0*, *maxQprod=0*)

7. An outlet with outflow as production flow determined by two rating curves (same as type 6), but with an allowed (higher) maximum production flow. If the outflow from the other outlet is above zero, part of that flow will be sent through this outlet as part of production flow instead. Defining variables in LakeData.txt: *rate*, *deltaw0*, *maxQprod* (*regvol=0*)

8. An outlet with outflow as minimum flow. The minimum flow is determined same as production flow by constants for different seasons or with a sinus adjustment. Defining variables in LakeData.txt: *regvol*, *minflow* (*rate=0*, *maxQprod=0*)

9. Minimum flow determined by a production flow below the threshold. In addition overflow is determined by a rating curve above the threshold. This could be a minimum environmental flow together with overflow. Defining variables in LakeData.txt: *regvol*, *rate*, *minflow* (*maxQprod=0*).

10. An outlet with outflow determined from a time series in Xobs (dwtr). This type is only used for the second outlet. Defining variables in LakeData.txt: *obsflow=1*

## Combinations of outlet types

Examples of common combinations of outlets:

- A hydropower plant with an estimated average production flow and a spill overflow at high water levels are simulated by combining production flow (outlet type 1) with rating curve determined overflow (outlet type 3). **Note** that different from a production flow and spill defined for one outlet (see Production flow above), the spill and produktion flow is here added together to form the total outflow. For one outlet the maximum of the two is used as total flow.
- Same as above a hydropower plant with production flow in one branch and overflow in another (outlet type 3), but in this case the maximum allowed production is assumed higher than the average production (outlet type 2). This means that if overflow is calculated due to high water level, part of that flow is lead to the powerplant and through production to the first branch.
- A hydropower plant where the production flow and spill goes to the same downstream subbasin (outlet type 5), but where a second spill outlet goes to another subbasin (outlet type 4).



- A hydropower plant with production flow in one branch (outlet type 1), and a minimum (environmental) flow plus overflow outlet in the old river course that goes to another subbasin (outlet type 9).

- A lake with two outlets with flow determined by a rating curve for each of them (may have different threshold also) are simulated by a combination of outlet type 3 and 4.

Many other combinations can be used, but not all outlet types are natural to combine. The following are **not allowed**:

Outlet 1	Outlet 2
1	2, 4
2	1, 2, 4, 7, 8, 10
3	5, 9
4	all
5	3, 9
6	4
7	2, 4, 7, 8, 10
8	2, 4, 7-9
9	3-5, 8, 9
10	all

### Change of branched flow after updating total flow

After calculating the outflow from the two outlets, which is done separately, the total outflow could be higher than the volume above the threshold. In this case the outflow will be reduced. When updating the total outflow from a lake with two outlets against observed total flow, the flow in respective branch has to be adjusted too. Depending on the outlet types this is done by different methods.

Lakes with a clear defined division between production flow and spill branch is handled so that production flow takes priority. For outlets with a maximum production flow, flow are diverted into this branch up to the maximum value and the rest into the other branch. For other lakes with a clear defined division between production flow and spill branch, the same method is applied but with the current production flow as the maximum value. For lakes with a minimum flow in one branch the minimum flow is given highest priority. Second priority is given to the production flow of the other branch (if any). At last the remaining flow is given to the branch for spill (outlet type 9) or the non minimum flow branch (outlet type 8). For the rest of the outlet type combinations the new flows are changed to be proportional to the old flows.

### Outlet lake as a multi-basin lake composed of lake basins

An lake can be part of a larger lake. It is then called a lake basin of the larger lake, which is called a multi-basin lake. A multi-basin lake is assumed to have equal water level in all lake basins. The lake basins can have different depths below the outflow threshold, and they all use the same threshold when the water level output is calculated (same reference).

## Inflow

A lake basin can receive local inflow, inflow from upstream areas (that are not part of the same multi-basin lake) and inflow from regional groundwater. Inflow works in the same way as for a simple lake (see above).

## Flow between lake basins

All lake basins of a multi-basin lake are connected by the main downstream path. They can in addition have branches within the multi-basin lake, where a fraction of the flow then goes. Flow through a lake divided into lake basins can occur in both direction according to the maindown (and possibly branchdown). The implementation prioritizes flow from nearby lake basins and flow in the downstream direction when calculating the flow between lake basins. The flow between the lake basins are calculated to reach an equal water level. No internal threshold between lake basins is considered when the flows are calculated.

The flow out of a lakebasin (to another lakebasin) is determined in several steps:

1. First the net outflow of each lakebasin is calculated. It is calculated as the difference between the current water level of the lakebasin and the water level of the whole lake after outflow from the multi-basin lake. First the average water level of the multi-basin lake is calculated, and from this the outflow(s) of the lake. Considering these outflows leaving the lake, a new average water level is calculated that the lake will have after all flows between lakebasins have been moved. This is the water level compared to the water level of the lakebasin in this step.
2. The flow out of a lakebasin is the netflow plus all inflows from other lakebasins. These inflows are outflows of other lakebasins. Starting upstream in the lake the inflows are added to the outflow to give the flow out of every lakebasin. Note that these flows can be negative.
3. The last step is to actually move the water. Preferably we want water to move from one lakebasin to the next, without mixing with water incoming from lakebasins further upstream. This we want so that substances do not travel all the way to the lake outlet on one time step. Therefore we try to remove all outflows from the lakebasins before we add the removed water into the downstream lakebasins. This might not be possible so we keep track of how much we can't move and repeat the third step if necessary.

## Outflow of lake composed of lake basins

The main outflow of the lake is assumed to be located in the last lake basin. Here outflow is estimated by a rating curve, either a specific equation or with the general equation, or by a regulation routine. The threshold may be changing over the year as described in Section Two rating curves above. If the water level is below the threshold then the outflow is zero, or if the production flow regulation is used, there is production flow. See Section Production flow above.

In addition the lake can have outflows out of the multi-basin lake from other lakebasins. These additional outflows are then the flow in the branch of these lakebasins. The outflow in a branch going out of the multi-basin lake is specified in LakeData for that lakebasin. It can be determined by a rating curve or by production flow. If it is a production flow it uses the same thresholds as those for the main outflow of the multi-basin lake, i.e.  $w0$  and  $wmin$ .

## Initialisation of lake volume

Unless a starting state is given from a file, the lakes start the simulation filled with water to their outflow threshold. That means for most lakes a water level equal to *lake\_depth*. Dams are filled to the dam's maximum elevation (equal to *lake\_depth*), except for flood control dams which are initialized with an empty regulation volume.

For lakes with outflow determined by a rating curve, the water level of the lake will often be higher than the outflow threshold level. The equilibrium level will depend on the size of the inflow and the outflow rating curve parameters. Depending on the residence time of water in the lake it may take time for this level to be established, and until then the outflow of the lake will be simulated lower than it should be. Thus a spin-up time is needed for a model simulation. For lakes in arid areas and for lakes with small catchments, the water level may be below the threshold and only occasionally have outflow.

## Constructing dams

It is possible to set at date from when a dam with regulation given in [LakeData](#) or [DamData](#) is supposed to be in place and regulate the flow. The same way it is possible to give a date when the dam is supposed to be removed. For the period before the dam is build and/or after the dam is removed the outflow of the lake (that is not a dam anymore) is calculated from the rating curve parameters used for spill during regulation period.

The removal of a dam is simulated by reducing the threshold of the lake (aka reducing the lake depth) to the *wmin*-level. Similarly the lake depth is increased when a dam is built. The lake depth given in the input file is assumed to be the lake depth of the dam (as it is for outlet lakes without dates for building/removing the dam). The lake depth is set to the *lake\_depth* given in input file minus the the distance calculated from regulation volume divided by lake surface area for the period without a dam. When during a simulation the date of the building or removing of the dam is passed, the lake depth used in the simulation is changed.

Removing of a filled dam may cause a flood wave travelling downstream. An option to have a smoother transition is to have an emptying phase with a gradually reducing threshold. This is implemented with a parameter specifying a number of days before the date the dam is removed that the threshold will start to decrease.

If shorter time steps are used, the dam is assumed to be build or removed on the hour 00.

## Links to file reference

Section	Symbol	Parameter/Data	File
Local lake (ilake)		<i>gldepi</i> , <i>illdepth</i> , <i>gratk</i> , <i>gratp</i> , <i>grata</i> , <i>ilratk</i> , <i>ilratp</i>	<a href="#">par.txt</a>
	<i>icatch</i>	<i>icatch</i>	<a href="#">GeoData.txt</a>
		or <i>gicatch</i> , <i>illicatch</i>	<a href="#">par.txt</a>
		<i>lks_num</i> , <i>lks_dp_N</i> , <i>lks_fi_N</i> , <i>lks_fa_N</i>	<a href="#">GeoData.txt</a>

Section	Symbol	Parameter/Data	File
Simple outlet lake or dam (olake)	$lake_{depth}$	$lake\_depth$	<a href="#">GeoData.txt</a> or <a href="#">LakeData.txt</a> or <a href="#">DamData.txt</a>
		$gldepo, ollddepth$	
		$ratcorr, olratk, olratp, kthr, klow, krel$	<a href="#">par.txt</a>
	$wmin$	calculated from $regvol$	
		$rate, exp, qamp, qpha$	<a href="#">LakeData.txt</a> or <a href="#">DamData.txt</a>
	$qprod$	$qprod1, qprod2, datum1, datum2$	
		$limqprod, w0adjdays$	<a href="#">LakeData.txt</a> or <a href="#">par.txt</a>
		$deltaw0$	<a href="#">LakeData.txt</a>
		$purpose, snowfrac$	
	$Qinf$	$qinfjan, qinf feb, \dots, qinfdec$	<a href="#">DamData.txt</a>
Outlet lake with two outlets		$ldtype=5$ or $6$	
		$w0ref, maxQprod, minflow, obsflow$	<a href="#">LakeData.txt</a>
Constructing dams		$bulddam, removedam$	<a href="#">LakeData.txt</a> or <a href="#">DamData.txt</a>
		$remdamdays$	<a href="#">par.txt</a>

## Links to relevant modules in the code

Modules (file)	Procedures	Sections
<a href="#">surfacewater_processes (sw_proc.f90)</a>	$calculate\_potential\_evaporation$	evaporation
	$calculate\_actual\_lake\_evaporation$	
	$average\_flow\_rating\_curve$	ouflow with rating curve
	$calculate\_ilake\_outflow$	local lake
	$calculate\_ilakesection\_outflow$	
	$set\_general\_rating\_k$	simple outlet lake
	$calculate\_outflow\_from\_outlet\_lake$	simple outlet lake
	$calculate\_branched\_flow$	outlet lake with two outlets
	$calculate\_lake\_outlet\_outflow$	outlet lake with two outlets
	$calculate\_lakebasin\_average\_waterstage$	outlet lake as multi-basin lake
	$calculate\_outflow\_from\_lakebasin\_lake$	
	$calculate\_branched\_flow$	
	$change\_current\_dam\_status$	constructing dams
<a href="#">modelmodule (model_hype.f90)</a>	$initiate\_model$	local lake
	$calculate\_local\_lake\_model$	
	$calculate\_flow\_from\_undivided\_lake$	simple outlet lake
	$calculate\_flow\_for\_lakebasin\_lake$	outlet lake with two outlets
		outlet lake as multi-basin lake

## References

Lindström, G., 2016. Lake water level for calibration of the S-HYPE model, Hydrology Research 47.4:672-682. doi: 10.2166/nh.2016.019.

## Floodplains

Floodplains may be simulated adjacent to a main river or outlet lake. Floodplain is a land area close to the surface water that may be temporary flooded. Water may flow from the surface water to the floodplain and back. The floodplain is a part (*fpf*) of the classarea of the main river or the lake. The lake/river surface area is the rest of the classarea. The floodplain is flooded when the water in the river or lake reaches above a certain threshold. The water on the floodplain may return to the river lake or may be trapped when the water level of the floodplain sink.

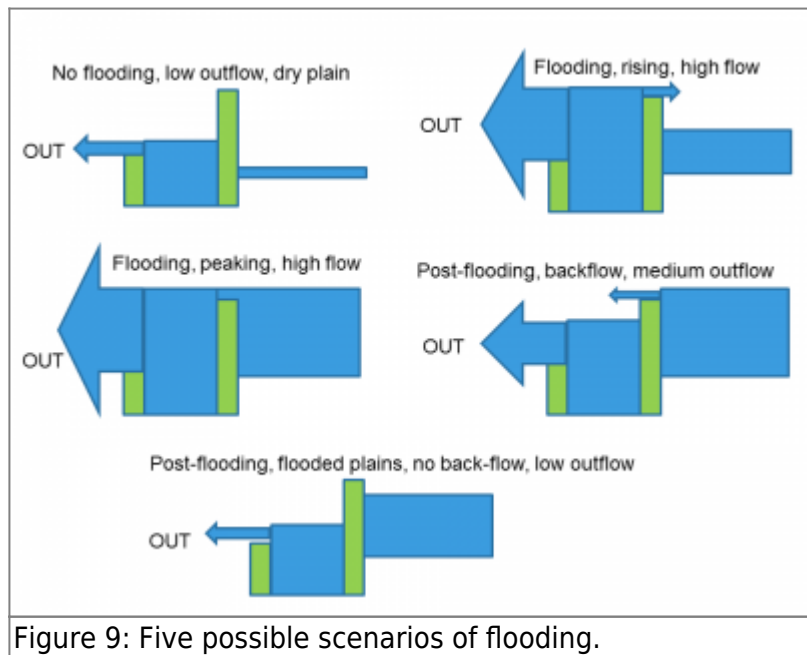


Figure 9: Five possible scenarios of flooding.

### Common floodplain processes

The water level of the floodplain and the extent of the flooded area ( $a_{\text{plain}}$ ,  $\text{m}^2$ ) are calculated from the volume ( $vol_{\text{plain}}$ ,  $\text{m}^3$ ), maximum area ( $a_{\text{max}}$ ) and water level at maximum areal extent ( $fym$ ,  $\text{m}$ ). The actual water level will be compared to the equilibrium water level to determine the water exchange. The equilibrium water level is the water level we would have if water was evenly distributed in the lake and the floodplain. It is solved from a second degree equation.

$$wl_{\text{plain}} = \sqrt{\frac{vol_{\text{plain}} \times fym \times 2}{a_{\text{max}}}}$$

$$a_{\text{plain}} = \frac{wl_{\text{plain}} \times a_{\text{max}}}{fym}$$

$$a_{\text{max}} = fpf \times \text{classarea}$$

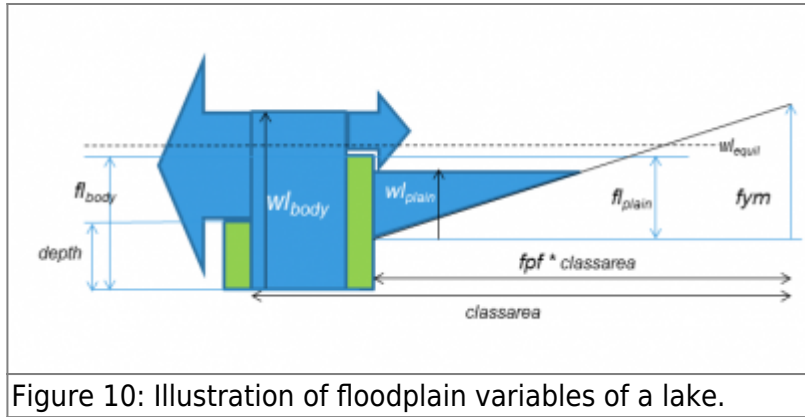


Figure 10: Illustration of floodplain variables of a lake.

The exchange of water between lake (or river) and floodplain is determined by the respective water levels ( $wl_{body}$  and  $wl_{plain}$ ), the equilibrium water level ( $wl_{equil}$ ) and the thresholds between the water bodies ( $fl_{body}/fl_{plain}$ ). If the water level is over the threshold and higher than the water level of the floodplain the flow from water body to floodplain is:

$$q = rc_{bodyToPlain} \times MIN \left( (wl_{lake} - fl_{body}) - wl_{equil}, (wl_{lake} - fl_{body}) \right)$$

If the floodplain water level is over the threshold and higher than the water level of the water body the flow from floodplain to body is similarly:

$$q = rc_{plainToBody} \times MIN \left( (wl_{plain} - fl_{plain}) - wl_{equil}, (wl_{plain} - fl_{plain}) \right)$$

Floodplain water level ( $wl_{plain}$ ) and degree of flooded area ( $\frac{a_{plain}}{a_{max}}$ ) are possible to print out from a simulation with an outlet lake or main river with floodplain.

## Simple floodplain model

The simple floodplain model simulates precipitation on the floodplain and evaporation from the flooded water, in addition to the exchange of water between the river or lake and the flood plain. All precipitation on the floodplain is contributing to flooded water.

## Floodplain model with soil routines

The floodplain model with soil routines is calculating all soil flows for the non-flooded part of the floodplain. The runoff from the floodplain soil goes to the flooded water, and not to the local stream as it does for ordinary land classes. Precipitation and evapotranspiration is calculated for the flooded water and the non-flooded soil. For the soil part, snow, infiltration and surface runoff, percolation, tile drainage and groundwater runoff is calculated as for an ordinary soil. Infiltration from the flooded water to the soil may occur. Regional groundwater percolation is only calculated for the flooded part together with percolation from the river.

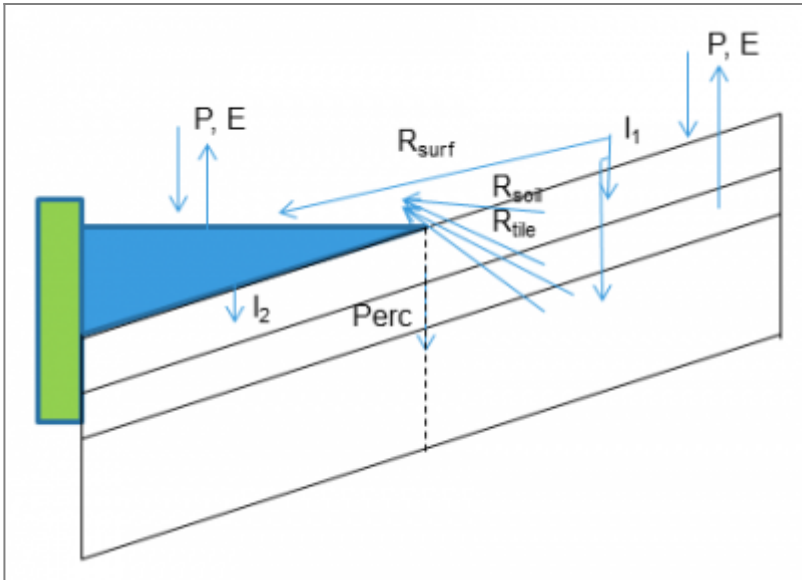


Figure 11: Illustration of floodplain soil flows.

## Connected floodplains

It is possible for flooded floodplains to limit the water flow from upstream rivers and lakes. This is determined based on the water levels in the floodplains, a higher water level of a downstream floodplain compared to an upstream floodplain may reduce the flow from the upstream compartment (main river or outlet lake) to the downstream compartment (main river or outlet lake). The damming flow is calculated as the flow to reach equilibrium water level between the two floodplains. The relative level of the two floodplains location is determined based on the classes average elevation and thresholds or given as indata separately. The function of connecting floodplains works within a subbasin or between connected subbasins. This floodplain features may be used together with the **Floodplain model with soil routines**.

## Links to file reference

Section	Symbol	Parameter/Data	File
	<i>fpf</i>	<i>fpfol</i> , <i>fpfmr</i>	<a href="#">FloodData.txt</a>
Common floodplain processes	<i>classarea</i>	<i>slc_nn</i> , <i>area</i>	<a href="#">GeoData.txt</a>
	<i>fym</i>	<i>fymol</i> , <i>fymmr</i>	<a href="#">FloodData.txt</a>
	<i>rc<sub>bodyToPlain</sub></i>	<i>rclfp</i> , <i>rcrfp</i>	
	<i>rc<sub>plainToBody</sub></i>	<i>rcfpl</i> , <i>rcfpr</i>	
	<i>fl<sub>body</sub></i>	<i>floll</i> , <i>flmrr</i>	
	<i>fl<sub>plain</sub></i>	<i>flolp</i> , <i>flmrp</i>	
Connected floodplains		<i>hrefl</i> , <i>hrefr</i>	

## Links to relevant modules in the code

Modules (file)	Procedures	Sections
surfacewater_processes (sw_proc.f90)	add_precipitation_to_floodplain	common floodplain processes
	calculate_floodplain_evaporation	
	calculate_floodplain_waterlevel	
	calculate_waterbody_floodplain_interflow	
	calculate_floodplain_volume	
	calculate_floodplain_equilibriumlevel	
	calculate_regional_floodplain_flows	connected floodplains
	calculate_two_floodplain_equilibriumlevel_dp	
	calculate_equilibrium_floodplain_level_eq1dp	
	calculate_equilibrium_floodplain_level_eq2dp	
	calculate_equilibrium_floodplain_level_eq3dp	
	calculate_interflow_between_floodplains2	
floodplain soilmodel (soilmodel4.f90)	soilmodel_4	simple floodplain model
		floodplain model with soil routines
		connected floodplains

## Bifurcations

Outflow from a subbasin may flow in one or two directions. The main flow follows the main channel, which is the downstream subbasin given by the path in [GeoData.txt](#). If there is a bifurcation, the branch flow goes to another downstream subbasin (with subid *branchid*). Any of the flows may go outside the model set-ups area, they are then no longer a concern of the model.

There are three ways to determine the flow in the different channels. 1) The division of the total outflow is determined in [BranchData.txt](#). 2) The outflow is calculated for two outlets separately and then divided into the channels. For the second alternative see [Outlet lake with two outlets](#) section above. 3) The demanded flow in the branch is prescribed ( $Q_{branch}$ ). If the total flow is not enough to fulfil the need, less water goes into the branch. The main channel gets the rest of the total flow.

For the first method the total outflow has been calculated the ordinary way. The flows division into the two channels is the determined by four parameters;  $mainpart$ ,  $maxQ_{main}$ ,  $minQ_{main}$  and  $maxQ_{branch}$ , which are set in [BranchData.txt](#). Zero values of the input data mean they are not used. The main flow ( $mainflow$ ) is calculated from the totalflow ( $q$ ) as:

$$mainflow = \begin{cases} q & q \leq minQ_{main} \\ mainpart \times (q - minQ_{main}) + minQ_{main} & minQ_{main} < q \leq q_{thresh} \\ maxQ_{main} & q > q_{thresh}, q_{thresh} = \frac{maxQ_{main} - minQ_{main}}{mainpart} + minQ_{main} \\ q - maxQ_{branch} & q > q_{thresh}, q_{thresh} = \frac{maxQ_{branch}}{1 - mainpart} + minQ_{main} \end{cases}$$



$$q_{thresh} = \text{MIN} \left( \frac{\max Q_{main} - \min Q_{main}}{\text{mainpart}} + \min Q_{main}, \frac{\max Q_{branch}}{1 - \text{mainpart}} + \min Q_{main}, 0 \right)$$

The rest of the total flow goes in the branch. The following figures give some examples of parameter combinations.

- Serie 1:  $\text{mainpart} = 0.5$ ,  $\min Q_{main} = 2$ ,  $\max Q_{main} = 10$ ,  $\max Q_{branch} = 0$
- Serie 2:  $\text{mainpart} = 0.5$ ,  $\min Q_{main} = 2$ ,  $\max Q_{main} = 0$ ,  $\max Q_{branch} = 5$
- Serie 3:  $\text{mainpart} = 0.5$ ,  $\min Q_{main} = 0$ ,  $\max Q_{main} = 10$ ,  $\max Q_{branch} = 0$
- Serie 4:  $\text{mainpart} = 0$ ,  $\min Q_{main} = 2$ ,  $\max Q_{main} = 0$ ,  $\max Q_{branch} = 5$

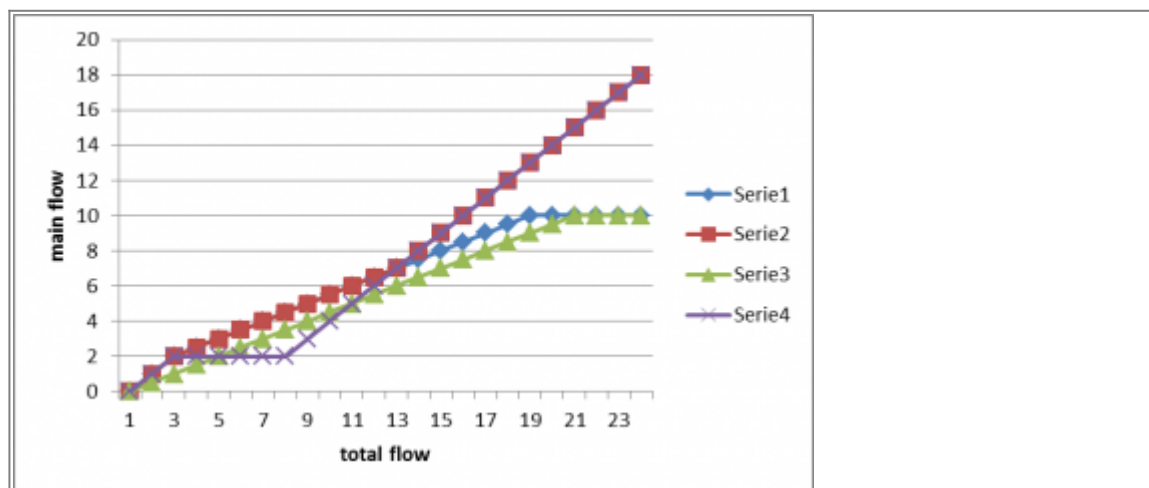


Figure 12: Main flow in relation to total flow for four different parameter combinations.

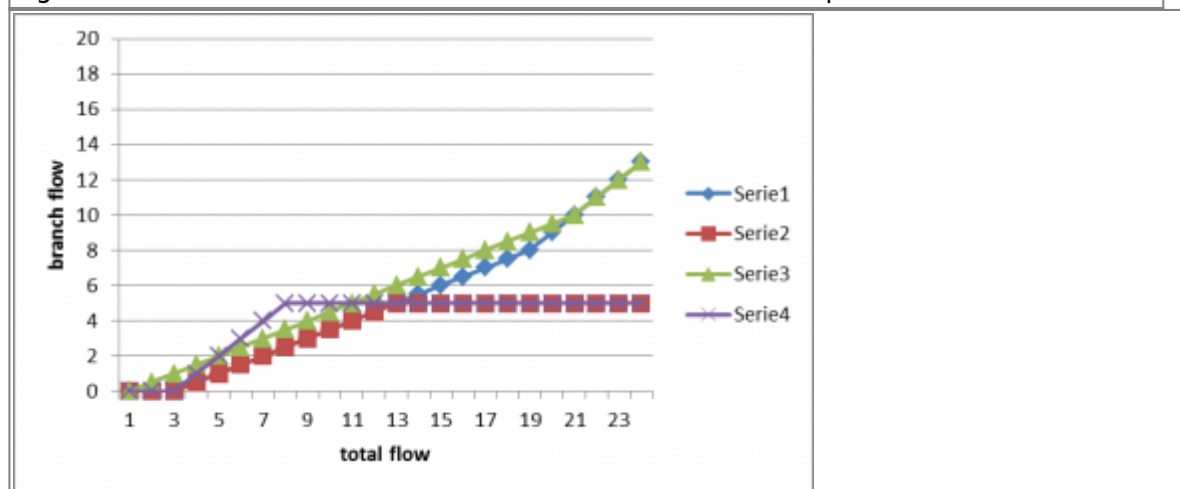


Figure 13: Branch flow in relation to total flow for four different parameter combinations.

## Links to file reference

Symbol	Parameter/Data	File
$\text{branchid}$ , $Q_{branch}$ , $\text{mainpart}$	$\text{branchid}$ , $Q_{branch}$ , $\text{mainpart}$	BranchData.txt
$\max Q_{main}$ , $\min Q_{main}$ , $\max Q_{branch}$	$\max Q_{main}$ , $\min Q_{main}$ , $\max Q_{branch}$	

## Links to relevant modules in the code

Modules (file)	Procedures	Sections
<a href="#">surfacewater_processes (sw_proc.f90)</a>	calculate_branched_flow	-
	calculate_branched_flow_new	-



# Nitrogen and phosphorus in land routines

HYPE simulate nitrogen and phosphorus in soil both as immobile soil pools and as fractions dissolved in soil water. The pools are changed by outside sources and internal soil processes. The following fractions are simulated:

- **fastN** - immobile pool of organic nitrogen in the soil with rapid turnover ( $\text{kg}/\text{km}^2 = \text{mg}/\text{m}^2$ )
- **fastP** - immobile pool of organic phosphorus in the soil with rapid turnover ( $\text{kg}/\text{km}^2 = \text{mg}/\text{m}^2$ )
- **humusN** - immobile pool of organic nitrogen in the soil with slow turnover ( $\text{kg}/\text{km}^2 = \text{mg}/\text{m}^2$ )
- **humusP** - immobile pool of organic phosphorus in the soil with slow turnover ( $\text{kg}/\text{km}^2 = \text{mg}/\text{m}^2$ )
- **IN** - concentration of inorganic nitrogen in soil water ( $\text{mg}/\text{L}$ )
- **ON** - the concentration of (dissolved) organic nitrogen in soil water ( $\text{mg}/\text{L}$ )
- **partP** - immobile pool of inorganic phosphorus adsorbed to soil particles ( $\text{kg}/\text{km}^2 = \text{mg}/\text{m}^2$ )
- **PP** - concentration of particulate phosphorus (of organic origin) in soil water ( $\text{mg}/\text{L}$ )
- **SP** - concentration of soluble (reactive) phosphorus in the soil water ( $\text{mg}/\text{L}$ )

The figure below give an overview of the nitrogen and phosphorus in soil in HYPE. The internal processes are shown with broad arrows and described in the section [Nutrient soil processes](#) below. Sources and sinks are shown with arrows and text, and are further described in sections [Nutrient sources](#) and [Vegetation and soil surface processes](#)

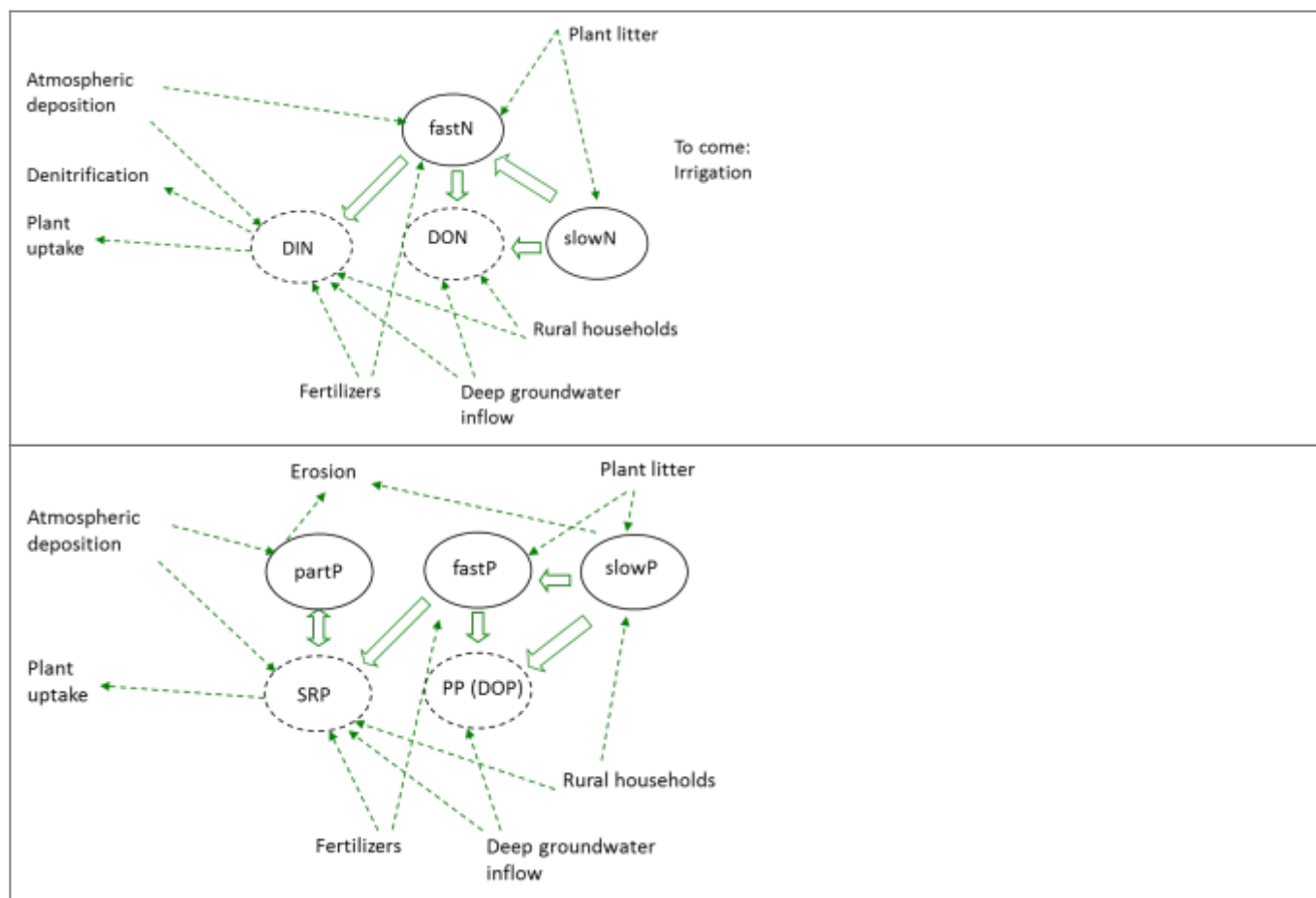


Figure 1: Sources and sinks of nitrogen and phosphorus in the soil. The humusN and humusP pools are called slowN and slowP in this figure.

# Nutrient Sources

## Fertilizer

Fertilization can be specified to be applied up to two times per year for each of the two fertilization forms, (commercial) fertilizer (*fert*) and manure (*man*). For each fertilization event the application date is specified, the amount of N and P (*fert(inut,ife)*, *man(inut,ife)*, where *inut*=N,P and *ife*=1,2 the fertilization event) to be added, and a management indicator (*fdown(ife)*, *mdown(ife)*). The last information indicates how much of the fertiliser is to be added to the second soil layer (the rest is added to the first soil layer). These parameters are given in [CropData.txt](#) and are crop and region dependent. For manure 50% of the nutrients are assumed to be in organic form (*inorgpart* = 0.5).

Fertilizer and the inorganic part of the manure are added to the IN and SP pools of the soil water. The total amount is spread out over a specified period (given by the general parameter *fertdays*) after the date of fertilization. An equal amount is applied each day for the specified period. The organic part of the manure is added to the fastN and fastP pools.

Each class can have two crops (*icrop*=1,2), the main crop and the secondary crop. The secondary crop do not have to cover (*a*) the whole class, e.g. if only 50% of the spring wheat (main crop) area have catch crop (second crop) (*a*(1)=1, *a*(2)=0.5). The total amount of added nutrients (*common<sub>add</sub>*) is calculated as the sum of fertiliser additions from the main and secondary crops.

Inorganic nutrients to be added to IN and SP pools:

$$common_{add}(inut, layer=1) = \sum_{icrop=1}^2 \sum_{ife=1}^2 a(icrop) \times \left( \frac{fert(inut, ife, icrop) \times (1 - fdown(ife, icrop))}{fertdays} + \frac{man(inut, ife, icrop) \times (1 - mdown(ife, icrop)) \times inorgpart}{fertdays} \right)$$

$$common_{add}(inut, layer=2) = \sum_{icrop=1}^2 \sum_{ife=1}^2 a(icrop) \times \left( \frac{fert(inut, ife, icrop) \times fdown(ife, icrop)}{fertdays} + \frac{man(inut, ife, icrop) \times mdown(ife, icrop) \times inorgpart}{fertdays} \right)$$

Organic nutrients to be added to fastN and fastP pools:

$$common_{add}(inut, layer=1) = \sum_{icrop=1}^2 \sum_{ife=1}^2 a(icrop) \times \frac{man(inut, ife, icrop) \times (1 - mdown(ife, icrop)) \times (1 - inorgpart)}{fertdays}$$

$$common_{add}(inut, layer=2) = \sum_{icrop=1}^2 \sum_{ife=1}^2 a(icrop) \times \frac{man(inut, ife, icrop) \times mdown(ife, icrop) \times (1 - inorgpart)}{fertdays}$$

Note that the terms for application of fertilizer and manure only are added during the active time, which is from the application date and for *fertdays* number of days thereafter.

## Plant residues

Plant residues are added at a specified date each year. They are added to the immobile soil pools (fastN, fastP, humusN, and humusP) to get a delay of the leakage and an accumulation of humus in the soil. The amount of crop residues (*resamount(inut)*, *inut*=N,P) and the date they are added on depends on crop and region and is given in the [CropData.txt](#) file. Part of the plant residues are added to the fast pool (*resfast*), and part to the humus pool. The plant residues are added to the upper two soil layers with a distribution (*resdown*) also specified in CropData. For each class the plant residues are weighted ( $\frac{common_{res}}{common}$ ) for main and secondary crop together every day with the help of share for each crop (*a(icrop)*, where *icrop*=1,2).

$$common_{res}(fast-inut, layer=1) = \sum_{icrop=1}^2 a(icrop) \times resfast(icrop) \times resamount(inut, icrop) \times (1 - resdown(icrop))$$

$$common_{res}(fast-inut, layer=2) = \sum_{icrop=1}^2 a(icrop) \times resfast(icrop) \times resamount(inut, icrop) \times resdown(icrop)$$

$$common_{res}(humus-inut, layer=1) = \sum_{icrop=1}^2 a(icrop) \times (1 - resfast(icrop)) \times resamount(inut, icrop) \times (1 - resdown(icrop))$$

$$common_{res}(humus-inut, layer=2) = \sum_{icrop=1}^2 a(icrop) \times (1 - resfast(icrop)) \times resamount(inut, icrop) \times resdown(icrop)$$

All crop residues are transferred into the soil at a given date as determined by the user in [CropData.txt](#). Note that the terms for application of residues only are added that day.

## Rural household diffuse source

Information on rural household diffuse source (private sewers) is given per subbasin. The source is defined by flow and concentrations of total nitrogen and phosphorus and fraction of IN and SP. The diffuse source may be divided into two parts with separate application point. One part is added directly to the local river, while the other part is added to the soil. The division between the two parts is determined by a general parameter (*locsoil*) or separately for each subbasin (*loc\_soil* in GeoData.txt). The default is to add all diffuse source to the local river. The soil part is added to soil water (for IN, ON, and SP) or solid pool (PP to fastP) in the bottom soil layers of all land classes in the

subbasin with a distribution proportional to the classes' area. Additional substance concentration can be used for the diffuse source.

An alternative method (model option) for adding a local diffuse source may be used. In this case the source is defined as a load to be added to the third layer of the soil of the land classes that have three soil layers (not including wetlands). Classes simulated with the traveltime soil model are also excluded. The load is dissolved in the soil water of the third layer of the remaining classes. In addition concentration and flow may be given, and is then added to the local river (*locsoil* parameter is not used).

## Links to file reference

Section	Symbol	Parameter/Data	File
Fertilizer	<i>fert</i>	<i>fn1, fp1, fn2, fp2</i>	CropData.txt
	<i>man</i>	<i>mn1, mp1, mn2, mp2</i>	
	<i>fdown</i>	<i>mdown1, fdown2</i>	
	<i>mdown</i>	<i>mdown1, mdown2</i>	
		<i>reg, cropid, fday1, fday2, mday1, mday2</i>	par.txt
		<i>fertdays</i>	GeoClass.txt
		<i>cropid, cropid2</i>	GeoData.txt
Plant residues		<i>src_nn, slc_nn</i>	
	<i>resamount</i>	<i>resn, resp</i>	CropData.txt
Rural household diffuse source		<i>resday, resfast, resdown</i>	
		<i>modeloption diffusesource</i>	info.txt
		<i>locsoil</i>	par.txt
		<i>loc_tp, loc_tn, loc_ts, loc_t1, loc_t2, loc_vol, loc_sp, loc_in, loc_ss, loc_si, loc_ds, loc_oc, loc_soil</i>	GeoData.txt
		<i>loc_ltp, loc_ltn, loc_lts, loc_lt1, loc_lt2, loc_lsi, loc_loc, loc_lsp, loc_lin, loc_lss, loc_lds</i>	

## Links to relevant procedures in the code

Modules (file)	Procedures	Sections
<a href="#">npc_soil_processes (npc_soil_proc.f90)</a>	<i>crop_sources</i>	fertilizer plant residues
	<i>local_diffuse_source</i>	rural household diffuse source
<a href="#">npc_surfacewater_processes (npc_sw_proc.f90)</a>	<i>add_diffuse_source_to_local_river</i>	rural household diffuse source

## Vegetation and soil surface processes

The CropData.txt file contains information on the various crops that are simulated. Each soil type-land use class (SLC-class) can have a main crop and a secondary crop. To reduce the number of SLC-classes in a simulation the individual crops with similar properties are brought together in a grouped crop (e.g. row crops). In CropData.txt there is information such as the amount of fertilizer, manure and crop residues introduced into the soil for different crops in different regions. The file also provides coefficients for nutrient uptake capabilities.

## Potential vegetation uptake of nitrogen

The potential uptake is used to calculate the absorption of IN and SP from the soil water (see section Vegetation nutrient uptake). The actual uptake will be limited by the nutrients available. Potential uptake of nitrogen by plants is based on a three-parameter equation (logistic growth) from SOILN (Eckersten et al., 1994). Nitrogen uptake (*uptake*) is calculated for the main and secondary crop and added to the common potential absorption function (*common\_uptake*, mg/m<sup>2</sup>/d) using the percentage of the area of the main crop (*part* = 1.0) and secondary crop (*part* < 1). The potential uptake is divided between the upper soil layer and second soil layer by a crop dependent input variable for the fraction of the upper soil layer. If the crop is growing in a class of only one soil layer, only the upper soil layer fraction of the potential uptake will be used in the calculations. The growing period with simulated nutrient uptake is determined by the sowing date (*bd2*) and continues until harvest (*bd3*). Potential uptake depends on three parameters (*up1*, *up2* and *up3*) and the number of days after sowing date. The uptake for a specific crop is calculated with the parameter values for that crop as

$$help = (up1 - up2) \times e^{-up3 \times (dayno - bd2)}$$

$$uptake = \frac{up1 \times up2 \times up3 \times help}{\left( (up2 + help) \times (up2 + help) \right)}$$

for the growing period. Outside the growing period the uptake is assumed to be zero.

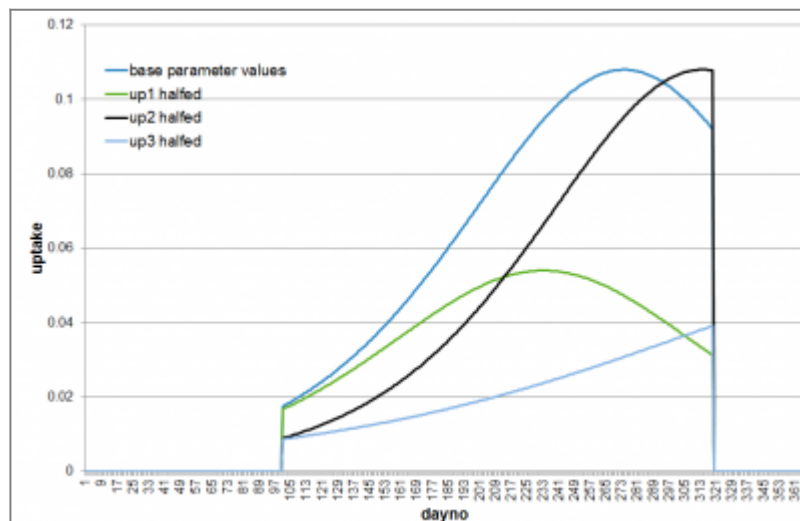


Figure 2 Uptake function over a year with growing period *bd2*=100 to *bd3*=230. Effect of changing parameters *up1*-*up3*.

Autumn-sown crops may take up IN and SP for a while after sowing in autumn. The same potential uptake of nitrogen as the main growing season are used, but uptake is limited by a temperature function. This uptake will run from the autumn sowing date (*bd5*) to the mid winter (31 December or 30 June depending on the autumn sowing date).

$$help = (up1 - up2) \times e^{-up3 \times \left( dayno - (bd5 + 25) \right)}$$

$$uptake = \frac{\frac{tmpfcn \times up1 \times up2 \times up3 \times help}{(up2 + help)}}{(up2 + help)}$$

where the temperature function (*tmpfcn*) is calculated as

$$tmpfcn = delim \begin{cases} MIN \left[ 1, \frac{(T-5)}{20} \right] & T > 5 \\ 0 & T < 5 \end{cases}$$

The growing period will then continue next season from *bd2* as described above.

The sowing date (*bd2*) may be given as a constant, or calculated dynamically depending on temperature. If it is dynamically determined it is set to the first day of the year which has a degree-day sum (*GDD*) above a threshold (*gddsow*). The degree-day sum (*GDD*) is calculated as

$$GDD(d+1) = GDD(d) + MAX(0, T - basetemp)$$

where *d* is day of year, *T* is air temperature (degree Celcius), *basetemp* is a temperature threshold. The GDD is accumulated for each day after *firstday* with day length larger than *daylength*. The GDD is zeroed at *firstday*.

## Soil erosion

The erosion of soil particles is calculated by HYPE for transport of particulate phosphorus (PP) or for the simulation of sediment material (SS). For transport of PP an erosion model depending on mobilization of particles by rain and by surface runoff is often used. This formulation is the default model for sediment transport, but another erosion model based on catchment erosion index can also be used.

In the first model PP can leave the land by two means, by surface runoff transport and by macropore flow through drainage pipes. The calculation of PP transport is done in three steps: first the erosion (mobilization) of soil particles from the land surface is calculated, secondly how much of the mobilized particles that are leaving the field is calculated, finally the amount of soil particles is converted to phosphorus.

### Step 1: Mobilization of particles from soil

Mobilization (*MobilisedRain* and *MobilisedSR*) occurs due to energy from falling raindrops (*Rain fall energy*) and/or from surface runoff. The kinetic energy in rainfall is calculated as a function of rainfall and day of the year (*dayno*). If the precipitation falls as snow, or if it falls on snow-covered ground or if it is smaller than 5 mm/day no mobilization occurs in the model. Some of the rain drop's energy can be absorbed by vegetation. Cropcover is defined as the portion of land that is sheltered from raindrops; for a description of how this is calculated, see [next chapter](#). The factor *common cropcover* is the total fraction of cover that the main and secondary crop give together. It



varies over the year due to crop growth and management. The mobilization is also influenced by soil erodibility (soil dependent parameter *soilerod* (g/J)).

$$Rainfall_{energy} = prec \times \left( 8.95 + 8.44 \times LOG_{10} \left( prec \times 2 \times \left( 0.257 + 0.09 \times \sin \left( 2\pi \times (dayno - 70) / 365 \right) \right) \right) \right)$$

$$MobilisedRain = Rainfall_{energy} \times (1 - common_{cropcover}) \times soilerod$$

When surface runoff occurs, soil particles are eroded and carried away as the soil surface is exposed to shear forces. The mobilization is calculated from the surface runoff (*sflow*), land slope, a parameter for soil cohesion (*soilcoh* (kPa) soil type dependent), and a general parameter (*sreroexp*). This type of erosion can be mitigated by protective vegetation or vegetation residues that are in contact with the ground. The calculation of this factor (*groundcover*) is described in [next chapter](#). The variable *common\_{groundcover}* is the total fraction of cover that the main and secondary crop give.

$$MobilisedSR = \frac{(sflow \times 365)^{sreroexp} \times (1 - common_{groundcover}) \times \frac{1}{0.5 \times soilcoh} \times \sin \left( \frac{slope}{100} \right)}{365}$$

All mobilised particles is not removed from the field, because sometimes the transport capacity of the particle-bearing water (*eflow*) will not suffice for the task. A *transportfactor* will reduce the particle amount:

$$transport\ factor = MIN \left( 1.0, (eflow / 4)^{1.3} \right)$$

Mobilised sediment (kg/km<sup>2</sup>) is calculated as:

$$mobilSed = 1000 \times (MobilisedRain + MobilisedSR) \times transport\ factor$$

The alternative erosion model calculates mobilised sediment (*mobilSed* (kg/km<sup>2</sup>)) based on rainfall (*rain*) and a number of model parameters and subbasin input data.

$$mobilSed = 1000 \times \left( \frac{slope}{5} \right)^{erodslope} \times erodluse \times erodsoil \times \frac{EI}{erodindex} \times rain^{erodexp}$$

The parameters *erodslope*, *erodexp* and *erodindex* are general. The parameters *erodluse* and *erodsoil* are land-use and soil type dependent. Subbasin input is needed on *slope*, the subbasins average slope, and an erosion index, *EI*. In addition an adjustment can be made with a monthly model parameter *erodmon* to help match seasonality of erosion.

$$mobilSed = mobilSed \times erodmon(month)$$

where *month* is the current month.

A selective process is affecting the soil erosion of phosphorus. Smaller and lighter particles are eroded easier than larger ones. The tiny particles contain more P per unit weight than the average particle of the soil. Therefore an enrichment factor (*enrichment*) is used. The enrichment factor is calculated from three parameters (*ppenrmax*, *ppenrstab*, *ppenrflow*), one of which is soil type dependent (*ppenrmax*), and the particle bearing flow. Typical values of the parameters, here called max, stab and flowstab, are given in the example in Figure 3.

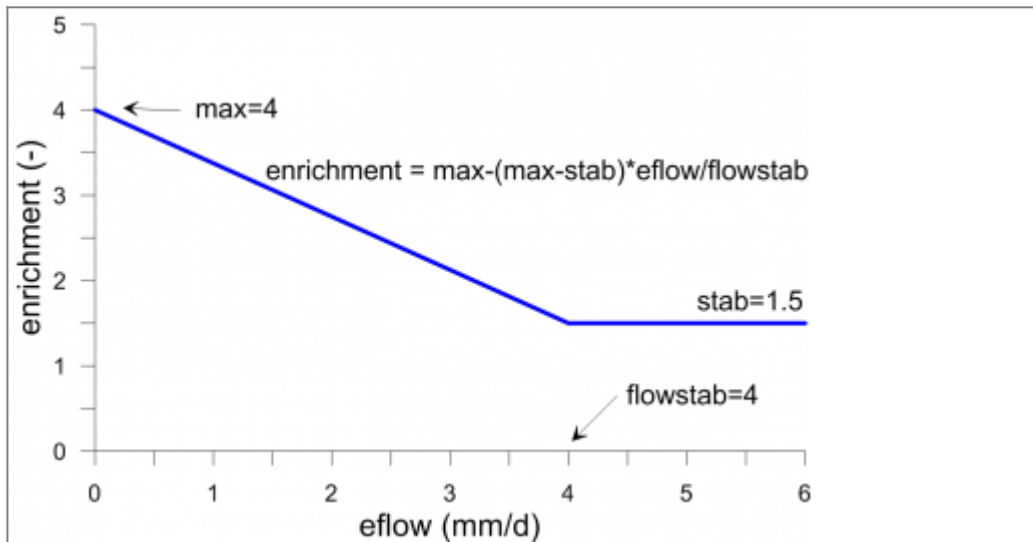


Figure 3 The enrichment factor for particulate phosphorus during soil erosion.

Eventually the mobilised PP (kg/km<sup>2</sup>) is calculated as:

$$mobilP = 0.000001 \times mobilSed \times \frac{PartP + HumusP}{thickness \times bulkdensity} \times enrichment$$

The calculated mobilised phosphorus (*mobilP*) is now available to add to the particulate phosphorus (PP) of surface runoff.

## Step 2: Transport of eroded phosphorus off the field

A portion of PP in surface runoff (*sflow*) is filtered out (for example by buffer zones). Filtering (*srfilt*) is determined by land use dependent parameters (*bufferfilt*, *innerfilt*, *otherfilt*), percentage of agricultural land close to a watercourse (*alfa*), and proportion of agricultural land near the rivers which have a protective buffer zone (*bufferpart*).

$$srfilt = otherfilt + alfa \times \left( 1 + bufferpart \times (bufferfilt - 1) \right) + innerfilt \times (1 - alfa)$$

Similarly, part of the PP which is transported by macropore flow (*mflow*) is filtered away between the soil surface and drainage pipes. The parameter that determines this effect (*macrofilt*) is soil dependent. The PP transported (*erodedP*) by surface runoff and macropore flow is collected in a temporary storage pool (*relpool* (kg/km<sup>2</sup>)) together with PP in tile runoff (*tilep*).

$$erodedP = \frac{(srfilt \times sflow + macrofilt \times mflow) \times mobilP}{eflow}$$

$$eflow = sflow + mflow$$

$$relpool = relpool + erodedP + tilep$$

### Step 3: Particles reaching the local stream

From the temporary pool phosphorus is released (*release* (kg/km<sup>2</sup>)) and then transported to the local river depending on the size of the total runoff (*runoff* (mm)). The parameters *pprelmax* and *pprelexp* are general.

$$release = MIN \left( relpool, relpool \times \left( runoff / pprelmax \right)^{pprelexp} \right)$$

The sediment may be removed from the temporary pool between the land surface and the local stream if no erosion occurs. A 1st order decay, with general parameter *eroddecay*, is used so that sediment doesn't remain detached forever.

### Crop cover and ground cover

Erosion can be mitigated by protective vegetation or vegetation residues that are in contact with the ground. Crop cover and ground cover reduce erosion by rain and surface runoff for particulate phosphorus (PP) and the default sediment transport model (SS). Each crop covers a fraction of the ground, thus for simultaneous crops (1st and 2nd crop) their respective crop/ground cover is combined to a common crop/ground cover.

Harvested crops have seasonally varying ground and crop cover, while permanent vegetation (e.g. forest) has constant values for these parameters. The input data needed for calculations (*ccmax1*, *ccmax2*, *gcmx1*, *gcmx2*) are given in [CropData.txt](#). Parameters *ccmax1* and *gcmx1* describe the maximum crop and ground cover during spring-summer growing season, parameters *ccmax2* and *gcmx2* are corresponding maxima for winter crop's growth. These maximum ratios are reached at *maxday1* and *maxday2*, which are defined as halfway between planting and harvest, and halfway between autumn planting and midwinter (1 January or 30 June depending on autumn planting date), respectively. From sowing the coverage fractions increase linearly up to their maximum values. After these dates maximum coverage is maintained to the next ploughing, harvest, or until the growing season starts again in the spring (for winter crops) (Figure 4). During the period between harvesting and ploughing, crop cover is equal to ground cover (*gcmx1*). At ploughing, ground and crop cover are reduced to zero. Parameters *bd1* and *bd4* describe the dates of spring and autumn ploughing. In the case of spring sowing, when no winter crop is growing, either one of the ploughing parameters can be set for ploughing date.

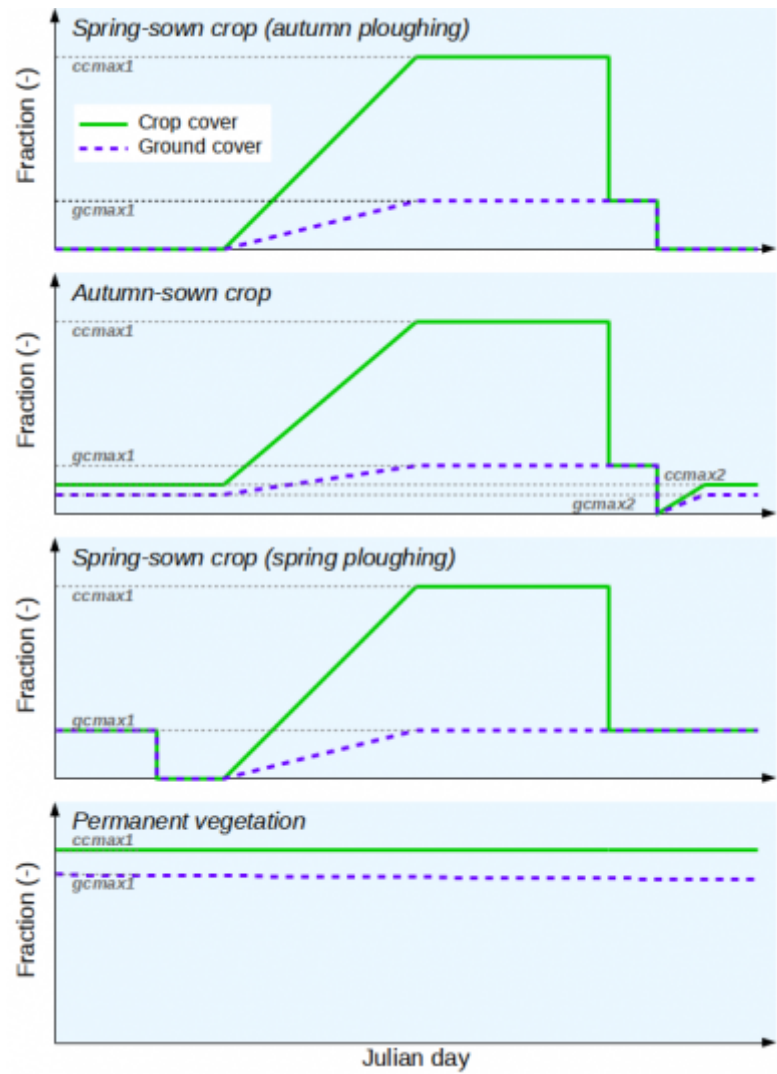


Figure 4: Crop cover and ground cover development for four different crop configurations.

Transformation of nitrogen from atmospheric deposition

The parameter *ponatm* indicates that some of the nitrogen deposition will be organic instead of only inorganic. For wet deposition and deposition on snow, this is adjusted when there is infiltration of rain and melted snow into the soil. A portion (*ponatm*) of IN in the infiltration are added to fastN in the top layer instead of following the water to the soil water.

Links to file reference

Section	Symbol	Parameter/Data	File
Potential vegetation uptake of nitrogen	<i>part</i>	<i>scr_nn</i>	<a href="#">GeoData.txt</a>
		<i>upupper, pnupr, bd2, bd3, bd5, up1, up2, up3, gddsow, basetemp, firstday, daylength</i>	<a href="#">CropData.txt</a>
	<i>T</i>	calculated from	<a href="#">Tobs.txt</a>

Section	Symbol	Parameter/Data	File
Soil erosion		<i>soilerod, soilcoh, seroexp, erodslope, erodluse, erodsoil, erodindex, erodexp, erodmon, bufferfilt, innerfilt, otherfilt, macrofilt, pprelmax, pprelexp, eroddecay</i>	<a href="#">par.txt</a>
	<i>slope</i>	<i>slope_mean</i>	<a href="#">GeoData.txt</a>
	<i>El</i>	<i>eroindex</i>	
	<i>alfa</i>	<i>close_w</i>	
	<i>bufferpart</i>	<i>buffer</i>	
Crop cover and ground cover		<i>ccmax1, ccmax2, gcmax1, gcmax2, bd1, bd4</i>	<a href="#">CropData.txt</a>
	<i>maxday1, maxday2</i>	calculated from <i>bd2, bd3, bd5</i>	
Transformation of nitrogen from atmospheric deposition		<i>ponatm</i>	<a href="#">par.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures	Sections
<a href="#">npc_soil_processes</a> ( <a href="#">npc_soil_proc.f90</a> )	<i>calculate_plant</i>	potential vegetation uptake of nutrients
	<i>particle_processes_for_runoff</i>	erosion calculations
	<i>calculate_pp_transport</i>	
	<i>calculate_hbvsed_erosion</i>	
	<i>calculate_erosion</i>	erosion calculations
		crop cover and ground cover
	<i>atmdep_in_loss</i>	transformation of nitrogen from atmospheric deposition

## Nutrient soil processes

### Soil pools - initial values

The initial pools of immobile soil nutrients are dependent on land use and is determined by the user and very important because they are changing slowly. The initial pools of the soil layers are calculated from a parameters. The parameters (*humusn0, humusp0, partp0, fastn0, fastp0*) are representative for the uppermost soil layer; the unit for these parameters is  $mg/m^3$ . The initial fastN, humusN, fastP, humusP and partP-pools decreases with depth, with half depth determined by land use dependent parameters (*hnhalf, hphalf* and *pphalf*).

The equation for a depth dependent soil nutrient is:

$$nut(d) = nut_0 \times e^{-par \times d}$$

where *d* is depth below zero-level, *nut0* is nutrient at zero-level (i.e. model parameter for uppermost soil layer) and *par* is calculated from respective nutrient half depth model parameter (*NNhalf*) as:

$$par = \frac{\log 2}{NNhalf}$$

As the model parameters that represent zero-level nutrient (*nut0*) is defined for the uppermost soil layer, the zero-level is the middle of uppermost soil layer. The following depths (*d*) represent the soil layers (1-3):

$$d(1) = 0$$

$$d(2) = \frac{soilthick(1)}{2} + \frac{soilthick(2)}{2}$$

$$d(3) = \frac{soilthick(1)}{2} + soilthick(2) + \frac{soilthick(3)}{2}$$

The nutrient is transformed into model soil nutrient pools (unit  $kg/km^2$ ) by taking into account the thickness of the soil layers.

$$pool(k) = nut(d(k)) \times soilthick(k), \quad k=1..3$$

The nutrients dissolved in soil water is initialized independent of depth. The initial IN, ON, SP and PP concentration are determined by land use dependent parameters (*inconc0*, *onconc0*, *spconc0* and *ppconc0*, mg/L) - valid for all three soil layers.

## Common functions

Many soil processes depend on temperature and soil moisture. The following equations are used in these cases. The temperature function (figure 5) depends on the estimated soil layer temperature (*soiltemp*). The soil temperature requires some parameters to be simulated, see Section [Soil temperature and snow depth](#).

```
tmpfcn = 2*((soiltemp - 20.0) / 10.0)
IF(temp < 5.0) tmpfcn = tmpfcn * (soiltemp / 5.0)
IF(temp < 0.0) tmpfcn = 0.0
```

The humidity function (figure 6) depends on soil moisture (*soil*) in the soil layer and the parameters of wilting limit (*wp*), field capacity (*fc*) and effective porosity (*ep*) transformed to unit *mm*. All these humidities are specified as percentages. The function includes coefficients *thetaupp* = 0.12, *thetalow* = 0.08, *thetapow* = 1.0 and *satact* = 0.6. Note that another function is used in the calculation of denitrification. For soil layers *k* = 1..3 the equation is:

```
IF(soil >= wp + fc + ep) THEN
  smfcn = satact
ELSE
  smfcn = min(1., (1-satact)*((wp+fc+ep-soil) / (thetaupp*soilthick))
  **thetapow + satact, ((soil-wp)/(thetalow*soilthick) **thetapow)
ENDIF
IF(soil < wp) THEN
  smfcn=0.0
```

ENDIF

The humidity function (figure 6) is always less than or equal to one while the temperature function may be greater than one when the temperature exceeds 20 degrees.

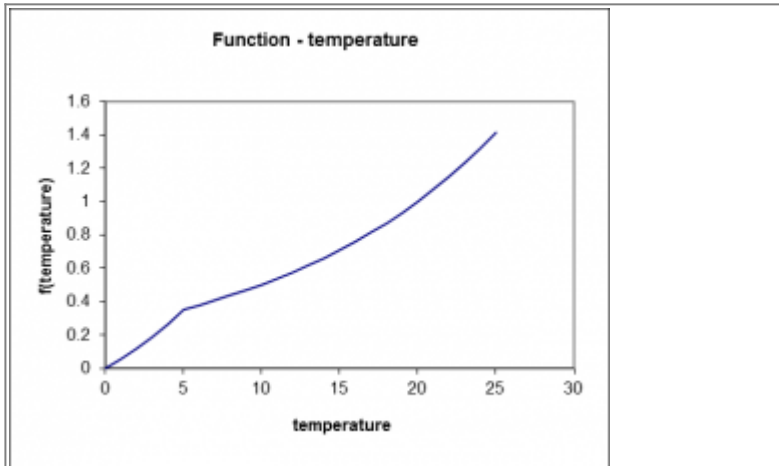


Figure 5: Common temperature function for soil processes

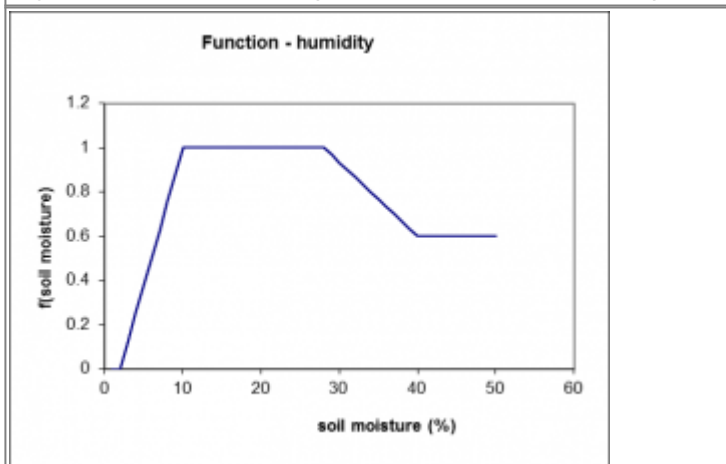


Figure 6: Common humidity function for soil processes

## Vegetation nutrient uptake

Vegetation nutrient uptake is a sink of the IN and SP concentrations in the two upper soil layers. The uptake ( $uptake / uptakeP$ ,  $mg/m^2/d$ ) depends on the potential uptake of nitrogen ( $common_{uptake}$ ), calculated for the combination of main crop and any secondary crop of the specified class. Vegetation nutrient uptake also depends on  $uptsoil1$  which gives the distribution of uptake between the two layers for a crop, the pool of IN ( $INpool$ ) and SP ( $SPpool$ ) and for phosphorus by P / N ratio ( $pnratio$ ). The pool of dissolved nutrients are calculated as the concentration multiplied by the water in the soil layer (in  $mm$ ). The main and secondary crop of a class is specified in `GeoClass.txt`.

$$\begin{aligned}
 uptakeN(1) &= MIN \left( uptsoil_1 \times common_{uptake}, a \times INpool(1) \right) \\
 uptakeN(2) &= MIN \left( (1 - uptsoil_1) \times common_{uptake}, a \times INpool(2) \right)
 \end{aligned}$$

$$\begin{aligned}
 uptakeP(1) &= MIN \left( pnratio \times uptsoil_1 \times common_{uptake}, a \times SP_{pool}(1) \right) \\
 uptakeP(2) &= MIN \left( pnratio \times (1 - uptsoil_1) \times common_{uptake}, a \times SP_{pool}(2) \right) \\
 a &= \frac{(soil - wp)}{soil}
 \end{aligned}$$

The plants can take up a maximum factor  $a$  of the pool. This factor is calculated as a percentage of soil water over the wilting point. The equations for the potential uptake of nitrogen ( $common_{uptake}$ ) are described under that section above. The constants  $uptsoil_1$  (percentage of plant nutrients absorbed top layer) and  $pnratio$  (ratio of N and P uptake) are given for each crop.

## Denitrification

Denitrification is a sink for IN in all three soil layers. Denitrification ( $denitr, mg/m^2/d$ ) depends on the denitrification rate and the pool of IN ( $IN_{pool}$ ) in each soil layer ( $k = 1, 2, 3$ ). The denitrification rate depends on a rate coefficient ( $drate$ ), soil temperature, IN concentration and soil moisture.

$$denitr(k) = drate \times IN_{pool}(k) \times tmpfcn(k) \times smfcn(k) \times concfcn(k)$$

The coefficient  $drate$  is determined by land use dependent model parameters  $denitr_{lu}$  and  $denitr_{lu3}$ . The temperature dependence ( $tmpfcn$ ) is described above. The soil moisture function (figure 7) is exponential and thus different from the general soil moisture function.

$$smfcn = \begin{cases} 0 & soil < pw \times lim \\ \left( \frac{\frac{soil}{pw} - dlim}{1 - dlim} \right)^{exp} & else \end{cases}$$

where  $pw = wp + fc + ep$

The function depends on soil moisture ( $soil$ ) and pore volume ( $pw$ ). It also depends on two constants; the limit where moisture is high enough to allow denitrification to occur ( $dlim = 0.7$ ) and the exponent ( $exp = 2.5$ ). These cannot currently be changed. The dependence of the denitrification rate on the IN concentration is described by a function with a half-saturation concentration (general parameter  $hsatINs$  was in earlier HYPE versions a constant equal to 1 mg/L) (Figure 8).

$$concfcn = \frac{conc}{conc + hsatINs}$$



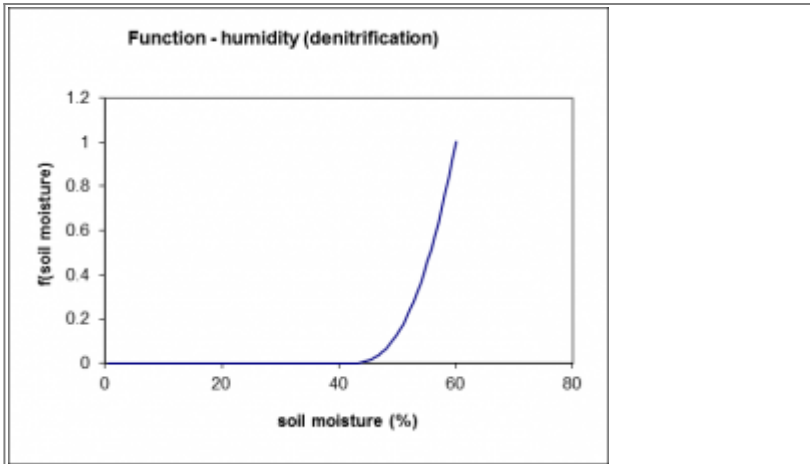


Figure 7: Soil moisture function in the denitrification process.

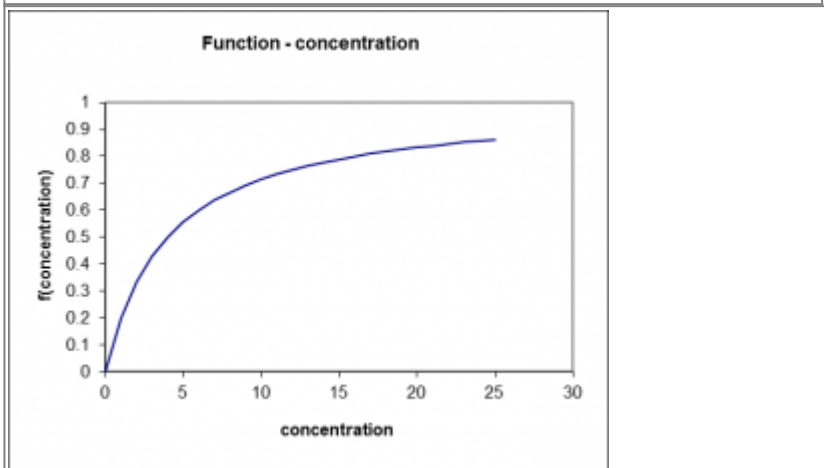


Figure 8: Concentration function in the denitrification process.

## Immobile soil nutrient pool transformations

### Turnover of fastN

Turnover of fastN is a sink for fastN and a source of IN in all three soil layers ( $k = 1..3$ ). Turnover ( $transfN$ , mg/m<sup>2</sup>/d) depends on a general parameter ( $minerfn$ ), temperature function ( $tmpfcn$ ), humidity function ( $smfcn$ ) and the pool of fastN ( $fastN$ ).

$$transfN(k) = minerfn \times tmpfcn(k) \times smfcn(k) \times fastN(k)$$

### Turnover of fastP

Turnover of fastP is a sink for fastP and a source of SP in all three soil layers ( $k = 1..3$ ). Turnover ( $transfP$ , mg/m<sup>2</sup>/d) depends on a general parameter ( $minerfp$ ), temperature function ( $tmpfcn$ ), humidity function ( $smfcn$ ) and the pool of fastP ( $fastP$ ).

$$transfP(k) = minerfp \times tmpfcn(k) \times smfcn(k) \times fastP(k)$$

## Turnover of humusN

Turnover of humusN is a sink for humusN and a source of fastN in all three soil layers ( $k = 1..3$ ). It is supposed to be a very slow process. Turnover ( $degradhN$ ,  $mg/m^2/d$ ) depends on a general parameter ( $degradhn$ ), temperature function ( $tmpfcn$ ), humidity function ( $smfcn$ ) and the pool of humusN ( $humusN$ ).

$$degradhN(k) = degradhn \times tmpfcn(k) \times smfcn(k) \times humusN(k)$$

## Turnover of humusP

Turnover of humusP is a sink for humusP and a source of fastP all three soil layers ( $k = 1..3$ ). It is supposed to be a very slow process. Turnover ( $degradhP$ ,  $mg/m^2/d$ ) depends on a general parameter ( $degradhp$ ), temperature function ( $tmpfcn$ ), humidity function ( $smfcn$ ) and the pool of humusP ( $humusP$ ).

$$degradhP(k) = degradhp \times tmpfcn(k) \times smfcn(k) \times humusP(k)$$

## Dissolution of soil organic pools

All organic soil pools of nutrient (humusN, fastN, humusP and fastP) contribute to dissolved organic nutrients in soil water (i.e. ON and PP). This process is a sink of the soil pools and a source of ON and PP in all three soil layers ( $k = 1..3$ ). Dissolution ( $mg/m^2/d$ ) depends on a model parameter (one for each soil pool,  $dispar$ ), temperature function ( $tmpfcn$ ), humidity function ( $smfcn$ ) and the size of the soil pool ( $soilpool$ ).

$$dissolution(k) = dispar \times tmpfcn(k) \times smfcn(k) \times soilpool(k)$$

The dissolution model parameters ( $dispar$ ) is land use dependent and named  $dissolfp$ ,  $dissolfn$ ,  $dissolhp$ , and  $dissolhn$  for dissolution of nutrient pools fastN, fastP, humusN and humusP respectively.

## Percolation

ON and PP concentration are reduced as the water percolates down through the soil layers. The decrease in concentration depends on land use dependent model parameters ( $par$ ).

$$conc = conc \times (1 - par)$$

## Balance SP - PartP

SP is in a dynamic equilibrium with P adsorbed to soil particles (partP). The equilibrium is described by the Freundlich equation. From the total amount of P (SP + partP) in the soil an equilibrium concentration of SP is calculated in the soil solution using the iterative Newton-Raphson method.

```

coeff = Kfr * bulkdensity * soilthick
DO WHILE
  fxn = xn * vol + coeff * (xn ** Nfr) - totalP
  fprimxn = vol + Nfr * coeff * ( xn **(Nfr-1))
  xn_1 = xn - fxn / fprimxn
  xn = xn_1
ENDDO

```

The coefficients *Kfr*, and *Nfr* are soil dependent, *bulkdensity* is the dry density of soil (= 1300 kg/m<sup>3</sup>) and *soilthick* is the soil layer thickness. *fxn* is function value for the concentration *xn*, *fprimx* is the function value of the derivative of *fxn*, and  $xn_1$  is the concentration of the next iteration. The resulting SP equilibrium concentration ( $xn = xn_1$ ) is used to calculate the PP equilibrium concentration ( $PP_{equi\_conc}$ ). A new SP concentration ( $SP_{conc}$ ) is calculated from the previous concentration ( $SP_{conc\_old}$ ) and the equilibrium concentration, taking into account the balance is not reached during the time step.

$$PP_{equi\_conc} = Kfr \times \left( xn_1^{Nfr} \right)$$

$$Adsdes = \left( PP_{equi\_conc} - concsolid \right) \times \left( 1 - e^{-Kadsdes} \right)$$

$$SP_{conc} = SP_{conc\_old} - \frac{\left( Adsdes \times bulkdensity \times soilthick \right)}{vol}$$

*Adsdes* is the amount of adsorbed/desorbed P, *Kadsdes* is a soil dependent rate constant, *equi\_concsolid* is the solid phase equilibrium concentration, and *concsolid* is the initial concentration the solid phase.

## Crop rotation effect on soil pools

The soil pools of nutrients are shared by crops on rotation on a field. This is modelled by averaging the soil pools once per year for the classes which are in the same crop rotation group.

## Links to file reference

Section	Symbol	Parameter/Data	File
Soil pools - initial values	<i>nut0</i>	<i>humusn0</i> , <i>humusp0</i> , <i>partp0</i> , <i>fastn0</i> , <i>fastp0</i>	<a href="#">par.txt</a>
	<i>NNhalf</i>	<i>hnhalf</i> , <i>hphalf</i> , <i>pphalf</i>	
		<i>onconc0</i> , <i>ppconc0</i>	<a href="#">GeoClass.txt</a>
	<i>soilthick</i>	calculated from <i>soillayerdepth</i>	
Common functions	<i>wp</i> , <i>fc</i> , <i>ep</i>	calculated from <i>wcwp</i> , <i>wcwp1-3</i> , <i>wcfc</i> , <i>wcfc1-3</i> , <i>wcep</i> , <i>wcep1-3</i>	<a href="#">par.txt</a>

Section	Symbol	Parameter/Data	File
Vegetation nutrient uptake		<i>cropid</i> , <i>cropid2</i>	<a href="#">GeoClass.txt</a>
	<i>uptsoil1</i>	<i>upupper</i>	<a href="#">CropData.txt</a>
	<i>pnratio</i>	<i>pnupr</i>	
Denitrification		<i>denitr1u</i> , <i>denitr1u3</i> , <i>hsat1Ns</i>	<a href="#">par.txt</a>
Immobile soil nutrient pool transformations		<i>minerfn</i> , <i>minerfp</i> , <i>degradhn</i> , <i>degradhp</i>	<a href="#">par.txt</a>
Dissolution of soil organic pools	<i>dispar</i>	<i>dissolfp</i> , <i>dissolfn</i> , <i>dissolhp</i> , <i>dissolhn</i>	<a href="#">par.txt</a>
Percolation	<i>par</i>	<i>onpercred</i> , <i>pppercred</i>	<a href="#">par.txt</a>
Balance SP - PartP	<i>Kfr</i>	<i>freuc</i>	<a href="#">par.txt</a>
	<i>Nfr</i>	<i>freuexp</i>	
	<i>Kadsdes</i>	<i>freurate</i>	
Crop rotation effect on soil pools		<i>crop rotation</i>	<a href="#">GeoClass.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures	Sections
<a href="#">npc_soil_processes</a> ( <a href="#">npc_soil_proc.f90</a> )	<i>soil_substance_processes</i>	
	<i>initiate_soil_npc_state</i>	soil pools - initial values
	<i>plant_uptake</i>	vegetation nutrient uptake
	<i>soil_denitrification</i>	denitrification
	<i>soil_pool_transformations</i>	immobile soil pool transformation
		dissolution
	<i>onpp_percolation_reduction</i>	percolation
	<i>balance_spsoil</i>	balance SP-partP
	<i>freundlich</i>	
	<i>croprotation_soilpoolaverage</i>	crop rotation
<a href="#">general_functions</a> ( <a href="#">general_func.f90</a> )	<i>tempfactor</i>	common functions
	<i>moisturefactor</i>	
	<i>halfsatconcfactor</i>	denitrification
	<i>exponential_moisturefactor</i>	

## Transport with soil water

IN, ON, SP and PP comes with water when it flows through the soil, but remain when water evaporates. Different flow paths contribute to the runoff of water and concentration of IN, ON, SP and PP from the corresponding soil layer. The flow paths possible today are: surface runoff (soil layer 1), runoff (soil layer 1, 2 and 3) and runoff through drainage pipe (soil layer 1, 2 or 3).

## Diagnostic output variables of soil nutrients

Some additional output variables are calculated from the soil variables. Gross and net soil load is calculated for the “root zone”, e.g. soil layer one and two together, and for soil layer three separate. A third option for soil load is calculated not strictly based on the soil layers but as an approximation for “groundwater” based on third soil layer but manipulated with tile drainage.

## Gross soil loads

Gross load to soil layer one and two (root zone). Components included: atmospheric deposition (wet and dry), fertilizers, crop residues, rural load if number of soil layers is less than three. Note that load from irrigation is not included. HYPE variable IDs:  $sl01=IN$ ,  $sl03=ON$ ,  $sl05=sl01+sl03=TN$ ,  $sl07=SP$ ,  $sl09=PP$ ,  $sl11=sl07+sl09=TP$

Gross load to soil layer three (lower soil). Components included: percolation from soil layer two to three, rural load if number of soil layers is three. HYPE variable IDs:  $sl13=IN$ ,  $sl15=ON$ ,  $sl17=sl13+sl15=TN$ ,  $sl19=SP$ ,  $sl21=PP$ ,  $sl23=sl19+sl21=TP$

Gross load to “groundwater” (modified soil layer three). Components included: percolation from soil layer two to three, rural load if number of soil layers is three, tile drain runoff if tile is in soil layer one or two. HYPE variable IDs:  $sl25=IN$ ,  $sl27=ON$ ,  $sl29=sl25+sl27=TN$ ,  $sl31=SP$ ,  $sl33=PP$ ,  $sl35=sl31+sl33=TP$

## Net soil loads

Net load from soil layer one and two (root zone). Components included: percolation from soil layer two to three, runoff from soil layer one and two, tile drain runoff if tile is in soil layer one or two, surface runoff. HYPE variable IDs:  $sl02=IN$ ,  $sl04=ON$ ,  $sl06=sl02+sl04=TN$ ,  $sl08=SP$ ,  $sl10=PP$ ,  $sl12=sl08+sl10=TP$

Net load from soil layer three (lower soil). Components included: runoff from soil layer three, tile drain runoff if tile is in soil layer three. HYPE variable IDs:  $sl14=IN$ ,  $sl16=ON$ ,  $sl18=sl14+sl16=TN$ ,  $sl20=SP$ ,  $sl22=PP$ ,  $sl24=sl20+sl22=TP$

Net load from “groundwater” (modified soil layer three). Components included: runoff from soil layer three, tile drain runoff if tile is in any soil layer. HYPE variable IDs:  $sl26=IN$ ,  $sl28=ON$ ,  $sl30=sl26+sl28=TN$ ,  $sl32=SP$ ,  $sl34=PP$ ,  $sl36=sl32+sl34=TP$

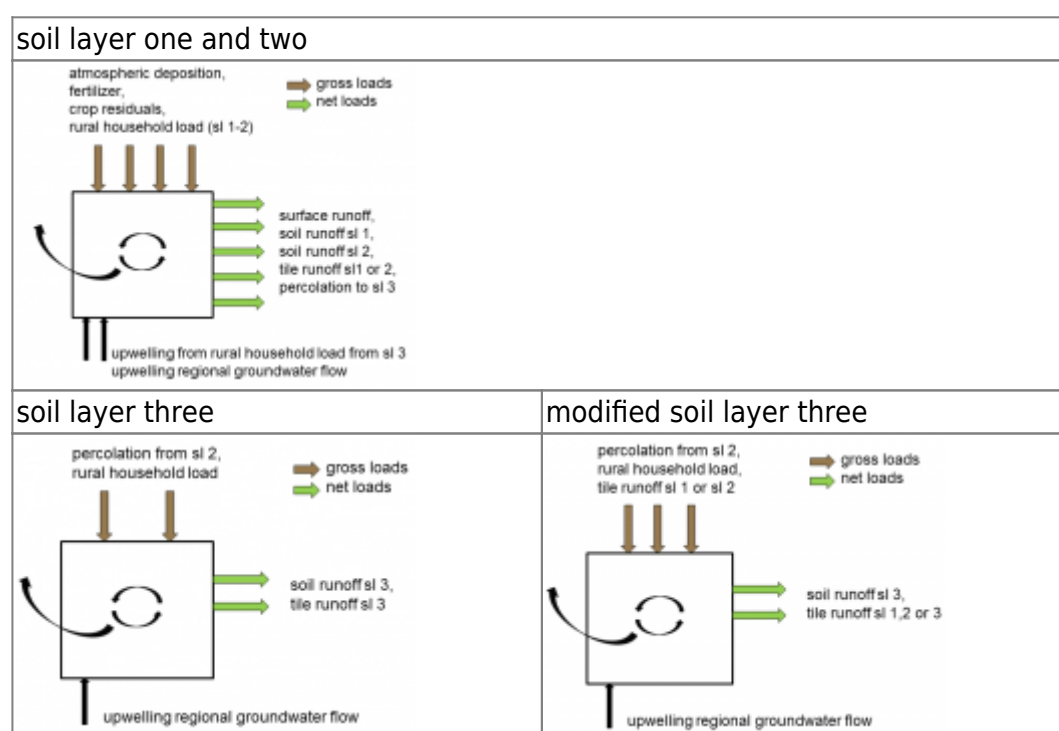


Figure 9: Components of calculated gross (brown) and net (green) loads of soil.

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">modelmodule (model_hype.f90)</a>	model
	define_output_variables

## Nutrient soil leakage from outer source

There is a model option in HYPE to not calculate nutrients as described above and instead use nutrient soil leakage from an outer source. In this case the calculations above is skipped, while soil water and runoff processes is calculated as usual. HYPE output variables related to soil nutrients are set to missing values.

The implemented soil leakage models are:

- 1. monthly typical concentrations for each subbasin
- 2. typical monthly load of excess nutrients for each class
- 3. monthly load of excess nutrients for each class
- 4. constant load and/or leakage for each class
- 5. monthly load and/or leakage for each class

The default is to use HYPE calculated runoff concentrations with the processes described in the sections above (modeloption soilleakage 0).

### 1. Monthly typical concentrations for each subbasin

The soil leakage is introduced into the model as inorganic and organic nitrogen, and soluble reactive phosphorus and particulate phosphorus concentrations of the total runoff. Different possibilities exist on how to introduce these regarding time variations (monthly seasonal variation, daily), spatial variations (subbasin, regions) and land use dependence (class groups). So far the only implemented alternative is the monthly seasonal variation.

The nutrient concentration of total runoff (HYPE variable ID c<sub>run</sub>) is set, and then used for all further calculations in the streams and lakes.

### 2. Typical monthly load of excess nutrients for each class

Soil leakage can be applied as monthly load of the excess nutrients and other substances from the soil. This load is given for each land class as typical monthly values. The load added to the class are transformed with processes depending on travel time to concentrations of the water leaving the soil as soil runoff or surface runoff or through other pathway. Simulation of water temperature (substance T2) can be combined with monthly load, but will not require load as input. The runoff T2 will be calculated from soil temperature as usual.

The soil leakage as monthly load is not compatible with other soil models than the travelttime soilmodel, i.e. special class code 5. Lake and river classes can be used by the same model set-up, but

not combined with floodplains. Special class code 3=glaciers can no be used.

### 3. Monthly load of excess nutrients for each class

Soil leakage can be applied as monthly load of the excess nutrients and other substances from the soil. This load is given for each land class as monthly time series. The load added to the class are transformed with processes depending on travel time to concentrations of the water leaving the soil as soil runoff or surface runoff or through other pathway. Simulation of water temperature (substance T2) can be combined with monthly load, but will not demand a monthly load. The runoff T2 will be calculated from soil temperature as usual.

The soil leakage as monthly load is not compatible with other soil models than the traveltime soilmodel, i.e. special class code 5. Lake and river classes can be used by the same model set-up, but not combined with floodplains. Special class code 3=glaciers can no be used.

### 4. Constant load and/or leakage for each class

Constant load and/or leakage can be used for individual classes together with ordinary soil model classes. Depending on classmodel defined for each class, load- or a leak-in data files checked for and read for that class. Special classmodel 5 and 6 can be used for this method. It is also possible to use the classmodel for only a selection of substances simulated. The classmodels' soil leakage input and processes will then be used for the substances files are given for, while the rest are calculated with the ordinary model.

### 5. Monthly load and/or leakage for each class

Monthly load and/or leakage can be used for individual classes together with ordinary soil model classes. It has the same construction as constant load and/or leakage for each class (method 4). The only difference is the columns with monthly values in the input files.

## The traveltime soilmodel

The travel time soilmodel calculate the concentration of runoff from soil from input of monthly load of substances to the land surface and a decay model. This soil model is used instead of the ordinary HYPE equations for substance processes in the soil.

The traveltime soilmodel (Hankin et al., 2019) will as default add the daily load to a pool on the surface of the land. The model will generate a release of dissolved substance from that pool.

$$release = pool \times \left( 1 - e^{-par \times flow} \right)$$

where *flow* is the current precipitation/snow melt (mm/ts). The general parameter *par* is different for the substances, e.g. in *release*. The released substances will be added to the the surface runoff and to the soil water of the uppermost soil layer with a distribution based on the total surface runoff and

infiltration flows.

Alternatively the daily load will be distributed among the surface pool, and the dissolved pools in the soil layers. This distribution is determined by parameters, e.g. *inloadf1*, *inloadf2*, etc. The parameters define the fraction for each soil layer and the rest is put in the surface pool. The load to the soil layers are directly put in the soil water.

The model can be combined with diffuse rural load to the soil, but not with atmospheric deposition load or wet concentration. Atmospheric deposition input will be zeroed.

Substances may be given a retention while in the soil. To simulate this an exponential decay equation is used. The process is governed by a decay model parameter *decay*, which denote the half life of a substance having a certain travel time to the stream, that is the reference time of travel (*totref*). As default *totref* is one day. Exponential decay is applied to substances in soil water and in the soil surface pool (*subst*). The decay can be sped up or slowed down relative to the reference. The half life can be adjusted with other parameters relating to land use's time of travel (*tot1*), class dependent time of travel (*tot2*), and soil layer/soil pool (*exp*, e.g. *totexp1*, *totexp0*). For instance could classes be divided according to distance from stream and the distance used to adjust the half life..

$$halflife = decay \times \left( \frac{totref}{tot} \right)^{exp}$$

$$subst = subst \times 2^{-\frac{ts}{halflife}}$$

where *ts* is length of time step in days. *subst* is IN=inorganic nitrogen, ON=organic nitrogen, SP=soluble phosphorus, PP=particulate phosphorus, OC=organic carbon, SS=suspended substances, AE=algae nitrogen, DS=dissolved silica, AS=algae silica or T1=tracer.

## Rootzone leakage soilmodel

The rootzone leakage soil model requires three layers of the soil. The upper two soil layers are defined as the rootzone for this model. The rootzone leakage soil model accept the rootzone leakage concentration from input and simulates substance processes only for in the third soil layer. For the upper two soil layers, water processes are calculated. The input rootzone concentration are added to the flow leaving the rootzone, i.e. surface runoff, soil layer one and two groundwater runoff, and percolation from soil layer two to soil layer three. In addition the concentration of the soil layer one and two is set to the root zone leakage concentration. In the third soil layer a simpler formulation of the soil substance processes is used than for the standard HYPE. Here an exponential decay of the dissolved substances are assumed, with different decay rates for the substances.

$$conc = conc \times 2^{-\frac{ts}{halflife}}$$

where *conc* is the concentration in the third soil layer, *halflife* is a general parameter (s) and *ts* is the timestep length (s).

The rootzone leakage concentrations can be given as a constant for the class or as monthly varying values. The concentrations are given for each substance and class simulated with the model. No atmospheric deposition is added to this model, but a diffuse source as rural households are accepted. Irrigation can be used, but will not affect the substances.



## Links to file reference

Section	Symbol	Parameter/Data	File
1. Monthly concentration		-	<a href="#">LeakageData.txt</a>
2. and 3. Monthly load or 4. and 5. Constant of monthly load		-	<a href="#">LoadNN_SLCNNN.txt</a>
	<i>par</i>	<i>inrelease, onrelease, sprelease, pprelease, ocrelease, ssrelease, aerelease, dsrelease, asrelease, t1release</i>	<a href="#">par.txt</a>
		<i>inload1f, inload2f, inload3f, onload1f, ..., t1load3f</i>	
	<i>decay</i>	<i>indecay, ondecay, spdecay, ppdecay, ocdecay, ssdecay, aedecay, dsdecay, asdecay, t1decay</i>	
	<i>totref</i>	<i>totref</i>	
	<i>tot</i>	<i>tot1 or tot2</i>	
	<i>tot1</i>	<i>intimeot, ontimeot, sptimeot, pptimeot, octimeot, sstimeot, aetimeot, dstimeot, astimeot, t1timeot</i>	<a href="#">par.txt</a>
	<i>tot2</i>	<i>traveltime</i>	<a href="#">classdata.txt</a>
4. and 5. Constant or monthly concentration	<i>exp</i>	<i>totexp0, totexpsl1, totexpsl2, totexpsl3</i>	<a href="#">par.txt</a>
		-	<a href="#">LeakNN_SLCNNN.txt</a>
	<i>halflife</i>	<i>indec3l, ondec3l, spdec3l, ppdec3l, ocdec3l, ssdec3l, aedec3l, dsdec3l, asdec3l, t1dec3l</i>	<a href="#">par.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">modelmodule (model_hype.f90)</a>	<a href="#">model</a>
<a href="#">datamodule (data.f90)</a>	<a href="#">load_soilleakage_concentrations</a>
	<a href="#">load_permanent_soilleakage</a>
<a href="#">modvar (modvar.f90)</a>	<a href="#">get_current_soilleakage</a>
<a href="#">default_soilmodel (soilmodel0.f90)</a>	<a href="#">soilmodel_0</a>
	<a href="#">update_rootzone_concentration</a>
	<a href="#">update_rootzone_model_concentration</a>
	<a href="#">set_model_for_substances</a>
	<a href="#">zero_rootzone_model_values</a>
<a href="#">soilmodel5 (soilmodel5.f90)</a>	<a href="#">soilmodel_5</a>
	<a href="#">soil_traveltime_processes</a>
	<a href="#">decay_of_simplesubstance</a>
	<a href="#">decay_of_simplesubstance_sorbedphase</a>
	<a href="#">left_after_decay</a>
	<a href="#">release_from_pool</a>
	<a href="#">distribute_soil_load</a>
<a href="#">npc_soil_processes (npc_soil_proc.f90)</a>	<a href="#">soil_substance_processes_of_third_soillayer9</a>

## References

Eckersten, H., P.-E. Jansson, and H. Johnsson 1994. SOILN model – user’s manual 2nd edition, Division

of Agricultural Hydraulics Communications 94:4, Department of Soil Sciences, Swedish University of Agricultural Sciences, 58pp, Uppsala.

Hankin, B., Strömqvist, J., Burgess, C., Pers, C., Bielby, S., Revilla-Romero, B., and L. Pope, 2019. A New National Water Quality Model to Evaluate the Effectiveness of Catchment Management Measures in England. *Water* 2019, 11, 1612. DOI: 10.3390/w11081612

# Nitrogen and phosphorus processes in rivers and lakes

HYPE simulates concentration of inorganic nitrogen (IN), organic nitrogen (ON), soluble reactive phosphorus (SP) and particulate phosphorus (PP) in discharge and other surface waters. As output also the total nitrogen (TN) and phosphorus (TP) concentration can be had. In addition total suspended sediments (TS) can be simulated. It consists of the sum of suspended sediments (SS) and algae (AE) simulated concentrations.

## Basic assumptions

Transformations of nutrients take place in lakes and rivers. For lakes, the whole water volume take part. For rivers, which hold delayed water in a queue and in the damping box, the processes is performed only in the damping box.

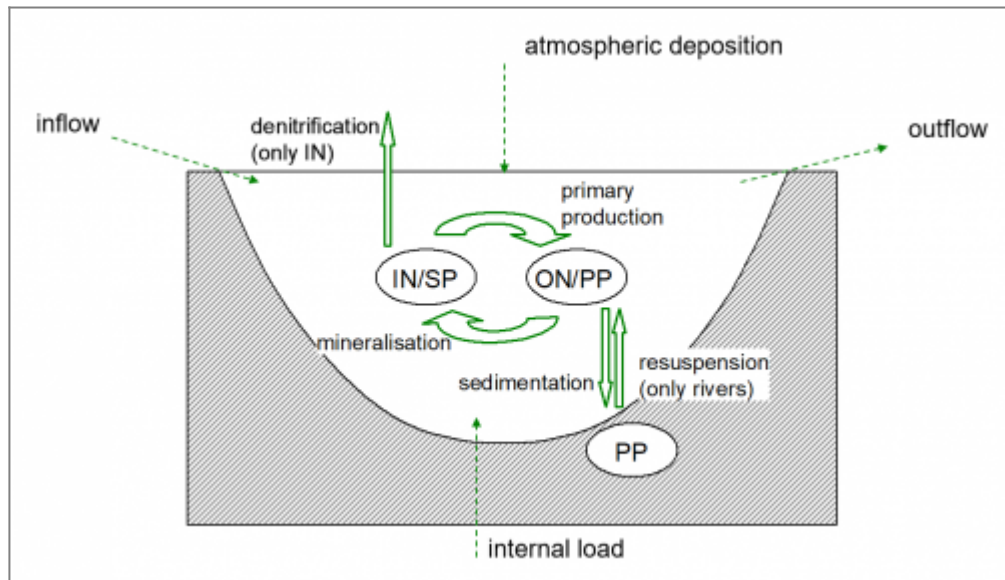


Figure 1: Sources and sinks of nitrogen and phosphorus in surface waters.

The processes of denitrification, primary production and mineralization have been implemented for both rivers and lakes. For particulate phosphorus (PP) there is an exchange with the river sediments. The rivers dimensions are used in the calculation of these processes. The width and depth of the watercourse are calculated from a number of empirical equations (for more information on these equations see “Modelling phosphorus transport and retention in river networks” by Jörgen Rosberg).

$$velocity = 10^{vel_1 \times mean\ flow^{vel_2} \times \left( \frac{flow}{mean\ flow} \right)^{vel_3}}$$

$$width = 10^{width_1 \times \left( \frac{flow}{velocity} \right)^{width_2 + width_3 \times \log_{10} \left( \frac{flow}{velocity} \right)}}$$

$$depth = \frac{\left( \frac{flow}{velocity} \right)}{width}$$

where  $vel1$ - $vel3$  and  $width1$ - $width3$  are lake region dependent parameters,  $flow$  is the flow of water in the watercourse ( $m^3/s$ ) and  $meanflow$  is a 365-day rolling average flow. The river's width is limited by the dead volume width and a parameter  $maxwidth$ . The rivers length is estimated at square root of catchment area or is given as input. The watercourses **bottom area** is calculated as the length times the width, where the maximum of the above-calculated width and dead volume width is used. The bottom area is used for the nutrient processes. If the river is a class and thus has a defined class area, the class area is used to estimate the bottom area instead.

The water temperature ( $T_w$ ) is used in some of the process calculations. It is calculated through weighting the air temperature ( $T$ ) and yesterday's water temperature. The weighting constant (similar to moving average period)  $T_{wdays}$  is set to 20 days for rivers and is by default 5 days for lakes. For lakes a depth dependent  $T_{wdays}$  can also be used. Then the weighting constant is set to the equivalent days of the lake's depths up to a maximum ( $maxpar$ , a parameter) plus 5 days. Note that the water temperature can fall below  $0^\circ C$ .

$$T_w = \left( 1 - \frac{1}{T_{wdays}} \right) \times T_w + \frac{1}{T_{wdays}} \times T$$

$$T_{wdays} = \begin{cases} MIN \left( MAX \left( lake_{depth}, 5 \right), 5 + maxpar \right) & lake \\ 20 & river \end{cases}$$

Alternatively the [water temperature calculated as a tracer](#) (called T2) may be used. This model option is set in info.txt.

## Links to file reference

Symbol	Parameter/Data	File
	$rivlen$ , $loc\_rivlen$	<a href="#">GeoData.txt</a>
$vel1$ , $vel2$ , $vel3$	$rivvel1$ , $rivvel2$ , $rivvel3$	<a href="#">par.txt</a>
$width1$ , $width2$ , $width3$	$rivwidth1$ , $rivwidth2$ , $rivwidth3$	
	$deadl$ , $deadm$ , $maxwidth$	
$maxpar$	$laketemp$	
$lake_{depth}$	$lake\_depth$	<a href="#">GeoData.txt</a> or <a href="#">LakeData.txt</a> or <a href="#">DamData.txt</a>
	$gldepo$ , $olldepth$ , $gldepi$ , $illdepth$	<a href="#">par.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">npc_surfacewater_processes (npc_sw_proc.f90)</a>	substance_processes_in_river
	substance_processes_in_lake
	river_characteristics
<a href="#">surfacewater_processes (sw_proc.f90)</a>	calculate_water_temperature
	set_water_temperature

## Denitrification

Denitrification, a sink for inorganic nitrogen in lakes and rivers, is a function of the bottom area, the IN concentration (*conc*) in water volume, water temperature ( $T_w$ ) and a rate parameter. The dependence on concentration is formulated as a half saturation equation. In the concentration function, the half saturation parameter ( $par_{half}$ ) is a general parameter, but it was in earlier HYPE versions a constant equal to 1.5 mg/L. Denitrification (*denitr*, kg/day) is limited to a maximum of 50% of the available IN pool.

$$tmpfcn = \begin{cases} 0 & T_w < 0 \\ \frac{T_w}{5} \times 2^{\frac{T_w - 20}{10}} & 0 < T_w < 5 \\ \frac{T_w - 20}{2^{10}} & else \end{cases}$$

$$concfcn = \frac{conc}{conc + par_{half}}$$

$$denitr = MIN \left( 0.5 \times IN_{pool}, par_{den} \times area \times concfcn \times tmpfcn \right)$$

The bottom area (*area*) is equal to the lake area for lakes, and watercourse width multiplied by the length for rivers (see above). The rate parameter ( $par_{den}$ ) depend on the water body. Local and main rivers have general parameters (*denitwrl*, *denitwrm*), while lakes in addition to use a general parameter (*denitwl*) can have lake specific values.

### Links to file reference

Symbol	Parameter/Data	File
$par_{half}$	<i>hsatINw</i>	<a href="#">par.txt</a>
$par_{den}$	<i>denitwrl</i> , <i>denitwrm</i>	
	<i>denitwl</i>	<a href="#">par.txt</a> or <a href="#">LakeData.txt</a>

### Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">npc_surfacewater_processes</a> ( <a href="#">npc_sw_proc.f90</a> )	<code>substance_processes_in_river</code>
	<code>substance_processes_in_lake</code>
	<code>denitrification_water</code>

## Primary production and mineralization

Primary production in lakes and rivers is a source of organic nitrogen and particulate phosphorus and a sink for inorganic nitrogen and soluble reactive phosphorus in the model. The reverse is true for mineralization. The processes are modelled together and only one is active at the time. Primary production and mineralization are controlled by two temperature functions. The first (*tmpfcn1*) is solely dependent on the water temperature ( $T_w$ ). It simulates the increased activity at warmer temperatures. The second (*tmpfcn2*) governs the relationship between primary production and mineralization and determines which one dominates. Net primary production is highest in spring (northern hemisphere) and changes into net mineralization when the temperature  $T_{10}$  is less than the temperature  $T_{20}$  in autumn. These two temperatures are calculated as the average water temperature of 10 and 20 days.

$$tmpfcn1 = \frac{T_w}{20.}$$

$$tmpfcn2 = \frac{(T_{10} - T_{20})}{5.}$$

$$tmpfcn = tmpfcn1 \times tmpfcn2$$

The primary production and mineralization processes are also governed by long-term average concentrations of modelled total phosphorus (*TP*) in the lake or watercourse through a concentration function (*TPfcn*) with the general half saturation parameter *hsatTP*. The half saturation parameter was in earlier HYPE versions a constant equal to 0.05 mg/L. If phosphorus is not modelled a long-term average total phosphorus concentration as a lake region dependent parameter (*tpmean*) is used. A limiting concentration (*lim*) can be set, and then the long-term average concentration is reduced before using it in the concentration function.

$$TPfcn = \frac{TP - lim}{(TP - lim) + hsatTP}$$

The estimated production/mineralization (*minprodNpot*, kg / day) is the potential transformation, and may be limited by the availability of nutrients. Only 50% of the available IN the pool (at the primary production) or 50% of the ON-pool (for mineralization) can be transformed. The potential phosphorus conversion (*minprodPpot*) is calculated in the same way, but with its own parameter (*wprodp*) and a factor for phosphorus/nitrogen ratio ( $NPratio = 1/7.2$ ). Similarly, there is a restriction against transforming maximum 50% of the SP and PP pools. The parameters *wprodn* and *wprodp* is generic or can be specified for each lake. The area is equal to lake area for lakes and bottom area for rivers (width multiplied by the length of the watercourse, see [Basic assumptions](#), or if the river is a class, the class' area). The water depth (*depth*) is the lake depth, and for the river the depth calculated [above](#).

$$minprodNpot = wprodn \times TPfcn \times tmpfcn \times area \times depth$$

$$\text{minprodPpot} = \text{wprodp} \times \text{TPfcn} \times \text{tmpfcn} \times \text{area} \times \text{depth} \times \text{NPratio}$$

For simulations of total suspended sediments, algae is simulated as a component of the suspended material. Nitrogen in algae is assumed to grow and decline with the same function as production and mineralisation of organic nitrogen. If nitrogen is simulated it uses the actual estimated production/mineralisation, but otherwise the potential production/mineralisation (*minprodNpot*) is used. The mineralisation of algae is limited to available amount, but the production is then unlimited.

## Links to file reference

Symbol	Parameter/Data	File
	<i>hsatTP, tpmean</i>	<a href="#">par.txt</a>
<i>lim</i>	<i>limsedPP</i>	
	<i>wprodn, wprodp</i>	<a href="#">par.txt</a> or <a href="#">LakeData.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">npc_surfacewater_processes</a> ( <a href="#">npc_sw_proc.f90</a> )	<code>substance_processes_in_river</code>
	<code>substance_processes_in_lake</code>
	<code>production_mineralisation</code>
	<code>calculate_lake_tpmean</code>
	<code>calculate_river_tpmean</code>

## Sedimentation and Resuspension

Sedimentation in **lakes** is a sink for particulate phosphorus (PP) and organic nitrogen (ON), as well as for suspended sediments (SS), algae (AE), and silica in algae (AS). Sedimentation (*sed<sub>lake</sub>*, kg/timestep) is calculated as a function of concentration (*conc*) in the lake and lake area (*area*, m<sup>2</sup>). The sedimentation rate (*par<sub>sedr</sub>*, m/timestep) is given by parameters (*sedon, sedpp, sedss, sedae, sedsi*) which are generic, but can be specified for each lake. The concentration used in the equation may be limited (*lim*, mg/L) by general parameters (*limsedON, limsedPP, limsedSS*), but not for AE (*lim*=0).

$$\text{sed}_{\text{lake}} = \text{par}_{\text{sed}} \times (\text{conc} - \text{lim}) \times \text{area} \times 10^{-3}$$

In the **river** we simulate both sedimentation and resuspension. No particles are removed from the simulation by sedimentation. For the original model they are redistributed over time and can come back through resuspension. For the alternative model it is possible to resuspend more particles than is previously settled. This double process is used for particulate phosphorus (PP) (described here) as well as for [suspended sediments](#) (SS) and [tracers](#) (T1). Particles in the sediments is collected in a pool (*sedimentpool*, kg).

For the **original model** the sediment pool will increase as particles from the water volume (*waterpool*, kg) settle at low water flows, and vice versa. At high water flow the sedimentation is lower and the resuspension higher, and more particles returns to the water. The net effect of these combined processes are given by the variable *sedresp*. It is governed by the current flow (*flow*) in relation to the bankful flow (*qbank*) and a general parameter *sedexp*. The net effect is determined by the sign of the variable ( $-1 < \text{sedresp} < 1$ ), and the variable size give the fraction of the pool that is transferred per day. The process will give either net sedimentation (*settl*, kg/day) or resuspension

(*resusp*, kg/day).

$$sedresp = \max \left( -1., \min \left( 1., \frac{qbank - flow^{sedexp}}{qbank} - \frac{flow^{sedexp}}{qbank} \right) \right)$$

$$settl = \begin{cases} sedresp \times waterpool & sedresp > 0 \\ 0 & sedresp < 0 \end{cases}$$

$$resusp = \begin{cases} -sedresp \times sedimentpool & sedresp < 0 \\ 0 & sedresp > 0 \end{cases}$$

where *flow* is the current river flow (m3/s) and *qbank* is the flow when river is filled to the brim. The latter flow is calculated as the second largest simulated flow in the last year. It is adjusted with a correction factor of 0.7 before use in the *sedresp* equation. The first alternative model is the same as the original model but the correction factor can be calibrated. It is given by general model parameter *qbank*.

The **second alternative model** is based on a simplified Bagnold equation (see Chapter on [Sediment](#)). It is used to simulate suspended sediment, and particulate phosphorus use it by exchanging the parameters for maximum sediment concentration (*spcon*, *spexp*) to parameters for maximum particulate phosphorus concentration (*suspconPP*, *suspexpPP*).

Links to file reference

Symbol	Parameter/Data	File
<i>area</i>	<i>slc_nn</i> , <i>area</i>	<a href="#">GeoData.txt</a>
<i>par<sub>sed</sub></i>	<i>sedpp</i> , <i>sedon</i> , <i>sedss</i> , <i>seds</i>	<a href="#">par.txt</a> or <a href="#">LakeData.txt</a>
	<i>sedae</i>	
<i>lim</i>	<i>limsedON</i> , <i>limsedPP</i> , <i>limsedSS</i>	
	<i>sedexp</i> , <i>qbank</i>	
<i>spcon</i> , <i>spexp</i>	<i>suspconpp</i> , <i>suspexppp</i>	

Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">npc_surfacewater_processes</a> ( <a href="#">npc_sw_proc.f90</a> )	<code>substance_processes_in_river</code>
	<code>substance_processes_in_lake</code>
	<code>lake_sedimentation</code>
	<code>river_sedimentation_resuspension</code>

Internal load

Lakes can leak phosphorus from the bottom e.g. release of phosphorus during oxygen deficiency or the mixing of the previously sedimented emissions. Internal load can be simulated for SP or PP or both



fractions of phosphorus. With two parameters the release of PP or SP is obtained for a specific lake ( $par_{prod}$ ,  $m/d$ ). The load to the lake ( $load$ ,  $kg/day$ ) is calculated as a function of lake water temperature ( $T_w$ ), average concentration of phosphorus in the lake ( $TPfcn$ ) and lake area ( $area$ ).  $TPfcn$  is currently not depending on simulated concentration as it is set to constant 0.1 mg/L.

$$load = \frac{par_{prod} \times TPfcn \times tmpfcn \times area}{1000}$$

$$tmpfcn = 0.86^{|T_w - 15|}$$

$$TPfcn = 0.1$$

## Links to file reference

Symbol	Parameter/Data	File
$par_{prod}$	$prodPP$ , $prodSP$	<a href="#">LakeData.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">npc_surfacewater_processes</a> ( <a href="#">np_sw_proc.f90</a> )	<a href="#">substance_processes_in_lake</a> <a href="#">internal_lake_load</a>

## Macrophyte uptake

For shallow waters in rivers and lakes macrophytes can grow and take up inorganic nutrients (IN and SP). The nutrients are lost to the model. Macrophyte uptake are controlled by a temperature function ( $tmpfcn$ ) and a concentration function ( $TPfcn$ ) for total phosphorus. The temperature function is composed of two parts; one exponential and one dependent on the water temperature ( $T_w$ ) above the average temperature of the last twenty days ( $T20$ ). The concentration function is the same half-saturation function as for production and mineralisation [above](#).

$$tmpfcn = \frac{\left(\frac{T_w}{20}\right)^{0.3} \times (T_w - T20)}{5}, T_w > 0 \text{ and } tempfcn > 0$$

The bottomarea ( $fracarea$ ) that is shallower than a production depth ( $proddep$  set by general parameters) is assumed to be active with macrophyte uptake. The river or lake are for this purpose assumed to be decreasing linearly with depth until twice the average depth.

$$fracarea = \frac{proddep}{\left(\frac{2 \times vol}{area}\right)} \times area$$

The uptake ( $upt$ ) is limited to maximum 50% of the available nutrients and the whole equation becomes:

$$upt = uptpar \times tmpfcn \times TPfcn \times fracarea$$

The uptake rate parameters (*uptpar*), different for IN and SP, are general. The process is similar to the [modelled macrophyte uptake in wetlands](#).

## Links to file reference

Symbol	Parameter/Data	File
<i>uptpar</i>	<i>muptn</i> , <i>muptp</i>	<a href="#">par.txt</a> or <a href="#">LakeData.txt</a>
	<i>muptnriv</i> , <i>muptpriv</i>	<a href="#">par.txt</a>
<i>proddep</i>	<i>muptdep</i> , <i>muptdepriv</i>	

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">npc_surfacewater_processes</a> ( <a href="#">np_sw_proc.f90</a> )	<code>substance_processes_in_lake</code>
	<code>macrophyte_uptaked</code>

# Water management

## Constructed wetlands for nutrient retention

For an overview of basic assumptions and explanation of variables see the [Basic assumptions section](#) in the Rivers and lakes chapter.

The wetlands that are simulated are small artificial ponds. They have an area and depth (*dep*), but their area is not taken into account in terms of precipitation and evaporation. The water flow passes through the wetlands without being affected, so it's just as nutrient traps that the wetland model is significant. There are two types of wetlands, just as for the rivers and lakes. They are situated before the river in the calculation scheme. The local wetland (*lrwet*) receives a share of the local runoff (*part*) the rest passes by unaffected. Wetlands in main rivers (*mrwet*) receive a portion of the flow in the main river and the rest passes unaffected.

### Wetland nutrient processes

In wetlands, retention of inorganic nitrogen is modelled (denitrification). For total phosphorus retention (sedimentation) and production (or release from sediments) of TP are modelled. The rates of these processes are constant coefficients (*teta*=1.2, *tkoeff*=20, *inpar*=2.3, *sedpar*= 0.09, and *uptpar*=0.1). The retention is limited to 99.9% of the substance in the wetland. The retention (*retIN*, *retTP*, g/d) depends on the rate parameter, the concentration in the wetland, wetland area, and for inorganic nitrogen also on 5-day-mean air temperature (*T5*). The production (*prodTP*, g/d) depends on a rate parameter, the concentration of the inflow to the wetland, wetland area, and a temperature function (30-day-mean air temperature, *T30*). The change in TP are divided equally between SP and PP.

$$retIN = \frac{inpar \times INconc \times T5 \times area}{1000.}$$

$$retTP = sedpar \times TPconc \times area$$

$$prodTP = uptpar \times TPIn \times area \times teta \left( T_{30} - tcoeff \right)$$

### Links to file reference

Parameter/Data	File
<i>lrwet_area</i> , <i>lrwet_dep</i> , <i>lrwet_part</i> , <i>mrwet_area</i> , <i>mrwet_dep</i> , <i>mrwet_part</i>	<a href="#">GeoData.txt</a>

### Links to relevant modules in the code

Modules (file)	Procedures
<a href="#">npc_surfacewater_processes</a> ( <a href="#">npc_sw_proc.f90</a> )	calculate_river_wetland
	calculate_riverwetland_np

References

Arheimer, B. and H.B. Wittgren, 2002. Modeling nitrogen removal in potential wetlands at the catchment scale, Ecol. Eng., 19: 63-80.

Tonderski, K.S., Arhemier, B. and Pers B.C., 2005. Modelling the Impact of Potential Wetlands on Phosphorus Retention in a Swedish Catchment, Ambio, 34(7): 544-551.

Constructed wetlands with water regulation capability

The wetlands are water classes, but simulated as a land class with special functions. If the soil is over saturated the standing water is the water volume of the wetland (*vol*, m3). If the soil is not over saturated, the wetland is dried out. The wetlands area (*area*, m2) is defined by the class area, and the depth (*w*) varies with flow. A threshold (*w0*) for the wetland outflow above the soil surface keep water in the wetland. The wetland outflow is determined by a rating curve above this threshold (see also [lake outflow](#)). The thresholds can be set by parameters or if parameters are not set it is equal to minus the streamdepth (from GeoClass).

$$out\ flow = k \times (w - w_0)^p$$

There are two types of wetlands; internal wetlands (*iwet*) and wetlands at the outlet of the subbasin (*owet*). Internal wetlands recieve a fraction (*ifraction*) of the runoff from other land classes (Figure 1). Outlet wetlands cannot be present in the same subbasin as an outlet lake. They recieve the flow from the main river of the subbasin (Figure 2).

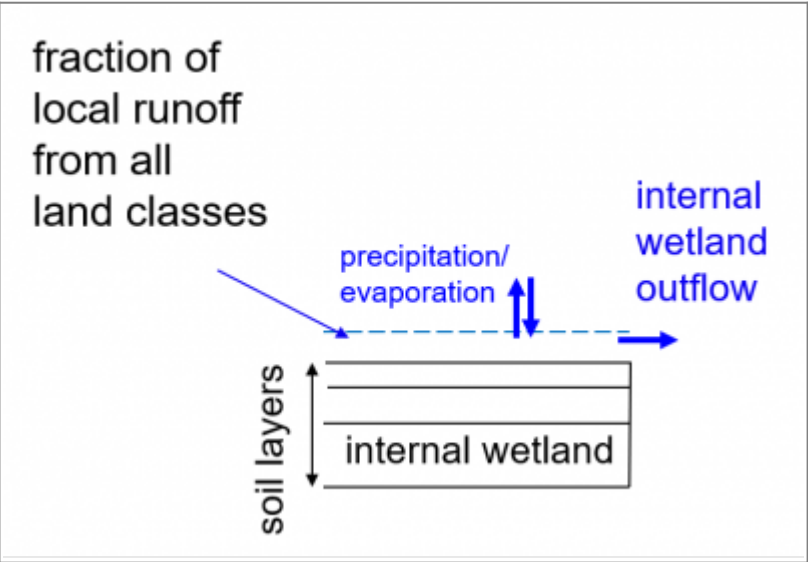


Figure 1 Water flows of internal wetland (iwet).

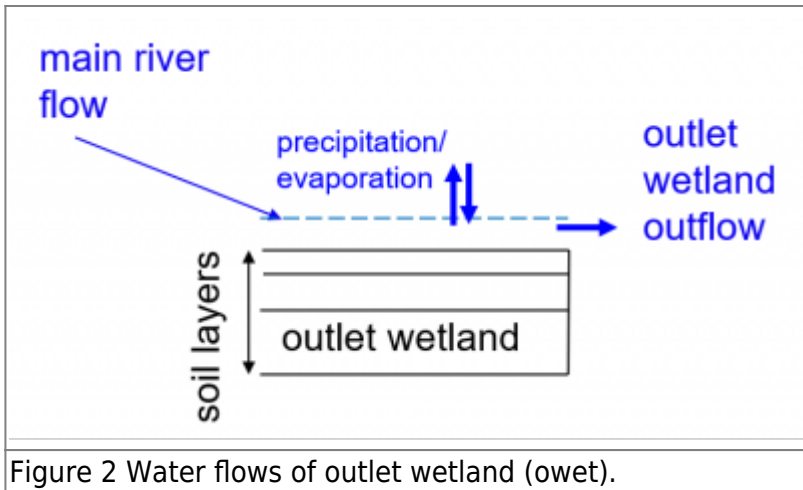


Figure 2 Water flows of outlet wetland (owet).

## Wetland nutrient processes

The concentration of the wetland (*conc*) is the concentration of soil water in soil layer 1. While calculating wetland nutrient processes only the nutrients in the water volume of the wetland in considered though. After that the nutrient concentration of the upper soil layer is updated.

Denitrification of inorganic nitrogen in the wetland is modelled as [denitrification in the soil water](#). Sedimentation of organic nitrogen, particulate phosphorus and suspended sediments are simulated (*sed*, g/d). Uptake of inorganic nutrients (IN and SP) are modelled as macrophyte uptake. The macrophytes are assumed to cover a part of the wetland area (*fracarea*). The covered fraction is calculated as the part that is shallower than a production depth (*proddep*) assuming the wetland area is decreasing linear with depth until twice the average depth of the wetland. The macrophytes are assumed to give residuals of equal amount of nutrient back to the sediment (i.e. immobile organic nutrient pools of soil layer one). The macrophyte uptake process (*upt*, g/d) depends on a rate parameter (*uptpar*), macrophyte fraction of wetland area, temperature (*tmpfcn*) and total phosphorus concentration (*TPfcn*). The temperature and TP functions are similar to the ones used by [primary production in lakes](#). The temperature function use 5- and 30-day mean air temperature (*T5*, *T30*). The half saturation concentration of TP is 0.05 mg/L (*hsatTP*). The sedimentation is limited to 99.9% of the substance in the wetland water, while macrophytes are limited to 50% of the dissolved inorganic nutrients.

$$sed = velpar \times area \times conc$$

$$upt = uptpar \times tmpfcn \times TPfcn \times fracarea$$

$$tmpfcn = \left( \frac{T_5}{20} \right)^{tmppar} \times \frac{(T_5 - T_{30})}{5}$$

$$TPfcn = \frac{TPconc}{TPconc + hsatTP}$$

$$fracarea = \frac{proddep}{\left( \frac{2 \times vol}{area} \right)} \times area$$

## Links to file reference

Symbol	Parameter/Data	File
<i>ifraction</i>	calculated from <i>iwetcatch</i>	<a href="#">GeoData.txt</a>
<i>w0</i>	streamdepth or <i>iwet0, owet0</i>	<a href="#">GeoClass.txt</a> or <a href="#">par.txt</a>
<i>k</i>	calculated from catchment area parameters <i>wetrate, grata</i>	<a href="#">par.txt</a>
<i>p</i>	<i>wetexp</i>	
<i>velpar</i>	<i>wlsed</i>	
<i>uptpar</i>	<i>wlmphtpin, wlmphtpst</i>	
<i>proddep</i>	<i>wlproddep</i>	
<i>tmppar</i>	<i>wltmpexp</i>	

## Links to relevant modules in the code

Modules (file)	Procedures
<a href="#">npc_surfacewater_processes (npc_sw_proc.f90)</a>	wetland_substance_processes
	wetland_watermodel
<a href="#">surfacewater_processes (sw_proc.f90)</a>	T2_processes_in_wetland
	get_wetland_threshold

# Irrigation

## General principles

Irrigation constitutes a key water management activity in many parts of the world. Therefore, the HYPE model has a routine to simulate irrigation. The representation of irrigation in the model is based on a set of principles. Firstly, the irrigation water demand is assessed. Subsequently, the demanded water is withdrawn from the defined irrigation water sources. HYPE can either withdraw water from defined sub-basins in the model domain (subject to availability), or from unlimited sources outside the domain. Finally, the withdrawn water is applied onto the classes from which the demand originated. In addition, water losses between demand, withdrawal, and application are taken into account (for withdrawals within the model domain).

A class is irrigated if the crop type associated with it is irrigated (defined in [GeoClass.txt](#)). A crop is irrigated if the irrigation input variables in the [CropData.txt](#) file are defined and non-zero (*plantday*, *lengthini*, *kcbini*, *lengthdev*, *lengthmid*, *kcbmid*, *lengthlate*, *kcbend*, *dlref*). Irrigation also requires appropriate values in the [MgmtData.txt](#) file (*gw\_part*, *regsrcid*, *irrdam*, *region\_eff*, *local\_eff*, *demandtype*) and the [par.txt](#) file (*pirrs*, *pirrg*, *sswcorr* etc.). See the [File Reference](#) for more details on each file and each parameter.

## Irrigation water demand

The irrigation water demand ( $W_{I,D}$ ) is calculated each day for each irrigated class (*j*) at the end of the soil water balance calculations. Two approaches to calculate  $W_{I,D,j}$  are implemented in HYPE, one for submerged crops (e.g. paddy rice) and one for non-submerged crops. The input variables *imm\_start* and *imm\_end* in [CropData.txt](#) define (1) the beginning and end of the submerged season,

and (2) if crops are submerged or not (zero is interpreted as a non-submerged crop).

## Non-submerged crops

For non-submerged crops, the calculations are based on the FAO-56 crop coefficient methods (Allen et al., 1998). The dual crop coefficient method is used because it is more specific than the single crop coefficient method, and more suitable for daily water balance models. Since transpiration is of primary interest in estimating crop water demand, the irrigation routine focuses on estimating potential transpiration ( $T_P$ ) with the basal crop coefficient ( $K_{CB}$ ) and the reference potential crop evapotranspiration ( $ET_0$ ):

$$T_P = K_{CB} \times ET_0$$

$ET_0$  follows the dynamics described above ( $ET_0 = e_{pot}$  here following Wisser et al. (2008)).  $K_{CB}$  depends on crop type and phenological stage, which is defined in CropData.txt.  $K_{CB}$  is constant during the initial development stage, then increases linearly during the development stage until it reaches the mid-season stage during which it is again constant. Finally,  $K_{CB}$  decreases linearly from the end of the mid-season stage until the end of the season. The dynamics of  $ET_0$  and  $K_{CB}$  produces a dynamic  $T_P$  profile (Figure 3). Allen et al. (1998) provide indicative values for  $K_{CB}$  (cf. their Table 17).

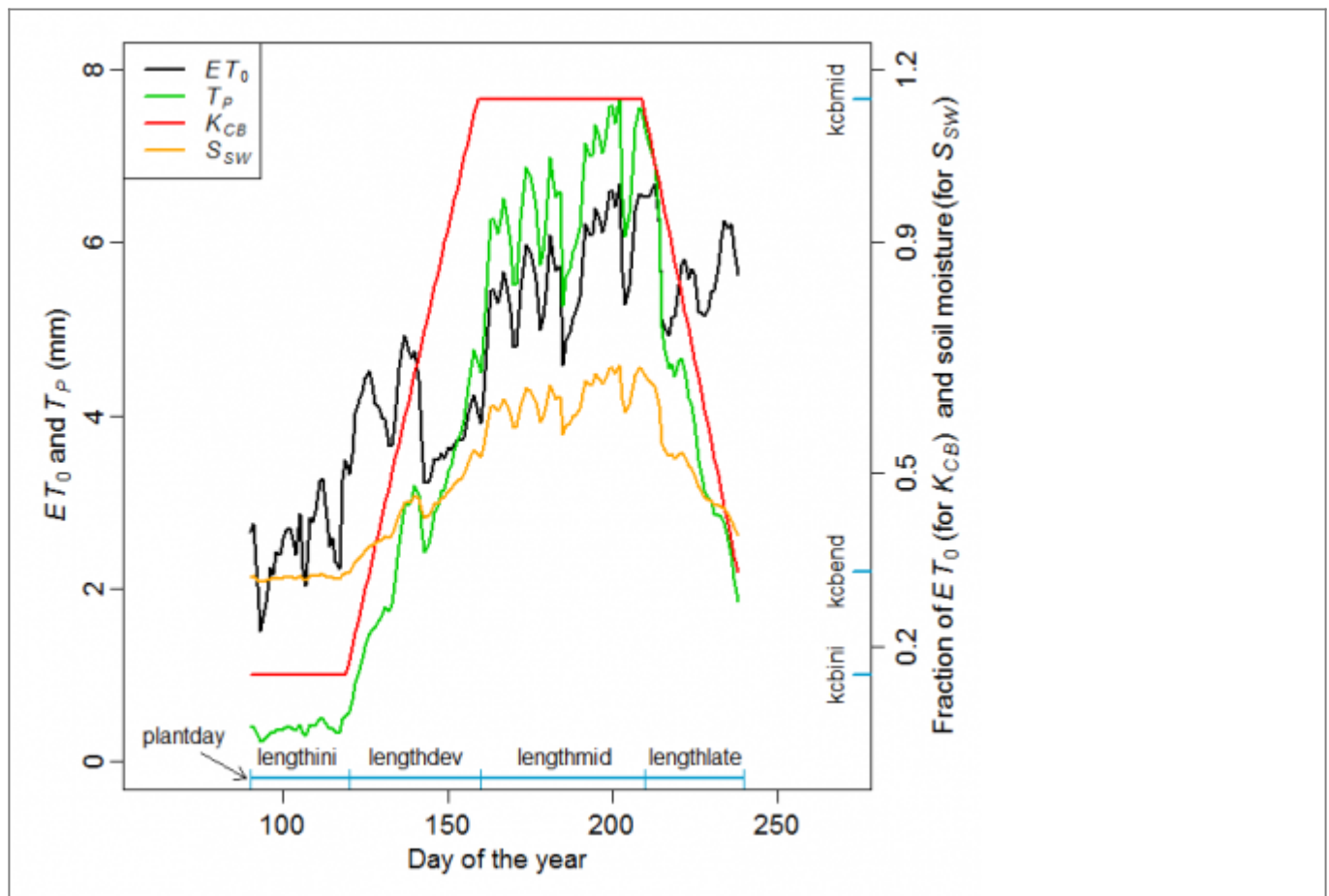


Figure 3 Illustration of  $ET_0$ ,  $T_P$ ,  $K_{CB}$ , and  $SSW$  for a typical maize crop on a medium coarse soil in southern Europe. Key input variables to define the  $K_{CB}$  profile are shown in blue.

On any given day, the model first calculates whether irrigation is needed, and then the amount required. The irrigation need is assessed by comparing the current soil water content ( $H$ ) with a dynamic irrigation threshold ( $S_{SW}$ , the soil water stress threshold):

$$if H < (S_{SW} \times P_{SSW CORR} \times AWC) \rightarrow irrigate$$

$H$  is the plant-available soil moisture (i.e. soil water above  $wcwp1$  and  $wcwp2$  in soil layers 1 and 2 respectively).  $AWC$  is the maximum plant-available water content in soil layers 1 and 2 (i.e. the sum of  $fc1$  and  $fc2$ ).  $S_{SW}$  is a fraction of  $AWC$  (defined upwards from  $wcwp$ ). Below  $S_{SW}$  the crop experiences water stress, creating a need for irrigation.  $S_{SW}$  varies from day to day and depends on the crop type and  $T_P$ :

$$S_{SW} = 1 - \left( DL_{ref} + 0.04 \times \left( 5 - \frac{T_P}{0.95} \right) \right)$$

$DL_{ref}$  is a crop-type specific reference depletion level (essentially the fraction of  $AWC$  that can be depleted before stress occurs, defined downwards from  $wcfc$ ). Allen et al. (1998) provide indicative values for  $DL_{ref}$  (cf. their Table 22). The  $S_{SW}$  equation is a slightly modified form of the original FAO-56 equation to account for the fact that only  $T_P$  is used here. A typical  $S_{SW}$  profile is shown in Figure 3. By default,  $S_{SW}$  is limited to the range 0.2 – 0.9, but it can be further refined with the parameter  $P_{SSW CORR}$  (*sswcorr* in par.txt) to maximum 1.

If irrigation is needed, the required irrigation amount ( $W_{I,D,j}$ ) can be calculated with three alternative methods in HYPE (chosen by the demandtype variable in MgmtData.txt):

(1) A constant  $W_{I,D,j}$  (defined by the *irrdemand* parameter in par.txt)

(2) Up to the field capacity:  $W_{I,D,j} = AWC - H$

(3) Up to a defined fraction of  $S_{SW}$  ( $P_{iwdfrac}$ , *iwdfrac* in par.txt):

$$W_{I,D,j} = \min \left[ \left( S_{SW} \times P_{SSW CORR} \times AWC - H \right) \times P_{iwdfrac}, \left( AWC - H \right) \right]$$

The fraction can be larger than 1. For example, to irrigate to a level 10% above  $S_{SW}$ ,  $P_{iwdfrac} = 1.1$ .  $W_{I,D,j}$  is, however, limited to  $AWC$ .



## Submerged crops

The irrigation of submerged crops aims to satisfy a target flooding level above the soil surface (Wisser et al., 2008). The target flooding level is a constant input parameter ( $P_{immdepth}$ , *immdepth* in par.txt). Irrigation is required if the water level of the top soil layer (H1) falls below the target flooding level:

$$if H_1 < WP_1 + FC_1 + EP_1 + P_{immdepth} \rightarrow irrigate$$

The irrigation water demand is equal to the amount needed to reach the target flooding level:

$$W_{I,D,j} = WP_1 + FC_1 + EP_1 + P_{immdepth} - H_1$$

If the submerged season is shorter than the crop season,  $W_{I,D}$  during the non-submerged period is calculated in the same way as for non-submerged crops. To maintain a desired flooding level and simulate terracing, for example, it may be necessary to adjust the soil and runoff parameters of the class with the submerged crops. When  $W_{I,D,j}$  has been calculated for each irrigated class in the sub-basin, the total field-scale irrigation water demand for the sub-basin ( $W_{I,D}$ ) is calculated:

$$W_{I,D} = \sum_{j=1}^N W_{I,D,j}$$

## Irrigation water withdrawal

Irrigation water can be abstracted from a set of water sources (Figure 4). Within a given sub-basin, water can be abstracted from the olake, the ilake, the main river, and from groundwater in a deep aquifer. In addition, water can be withdrawn from the olake and the main river of another sub-basin. These sources can be used on their own or in combination. Alternatively, HYPE can withdraw water from an unlimited source outside the model domain. This is specified with the *irrunlimited* code word in info.txt, and applies to all irrigated sub-basins.

Withdrawals are calculated directly after the local discharge and the upstream discharge has been combined to flow into the main river of a given sub-basin.

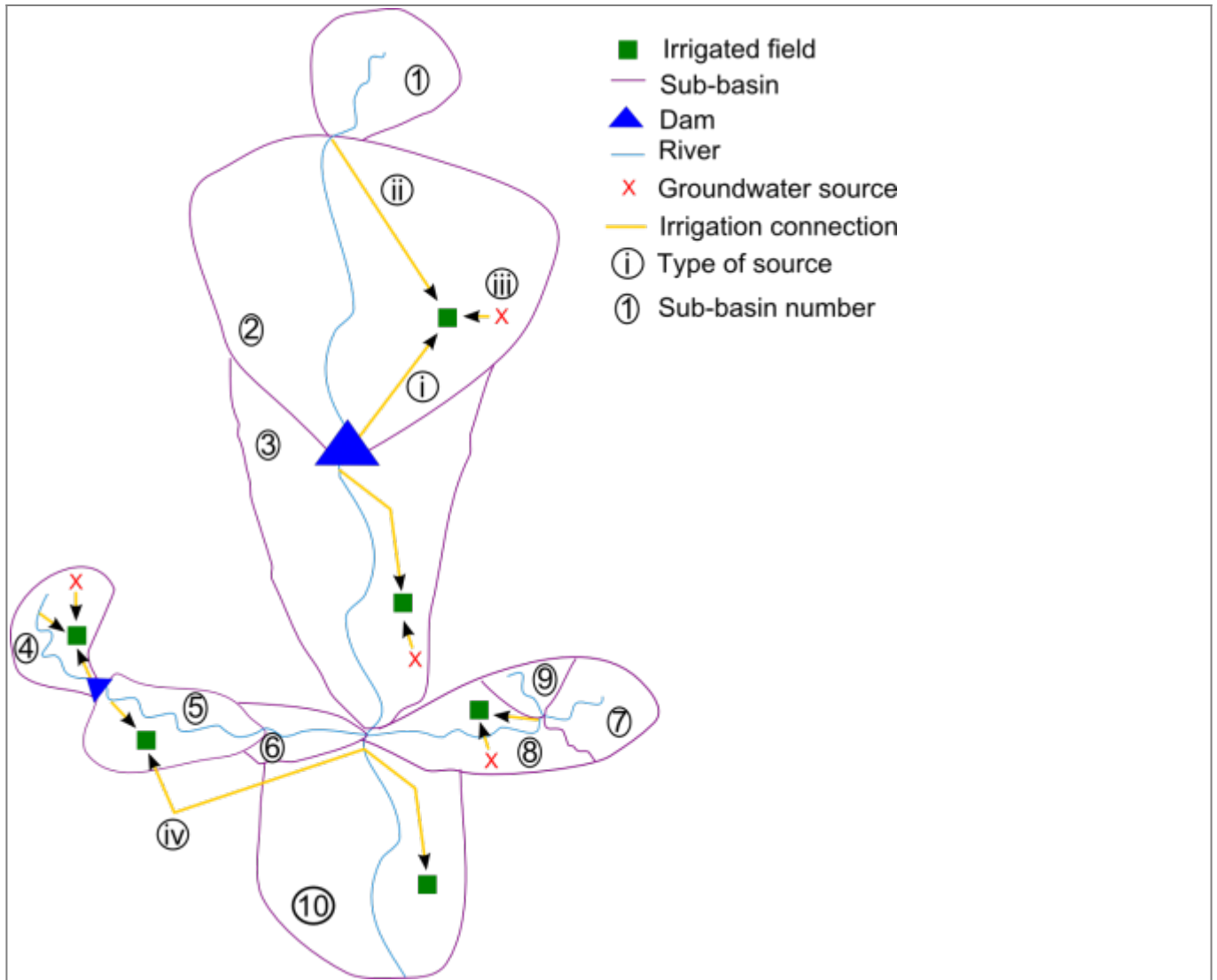


Figure 4: Schematic illustration of the available irrigation sources in HYPE. Irrigation water can be withdrawn from: (i) dams in the sub-basin (olake and ilake), (ii) the main river in the sub-basin, (iii) groundwater in a deep aquifer, and (iv) dams (olake) and main rivers in other sub-basins.

### Irrigation inefficiencies within the sub-basin

Before any withdrawal occurs, the field-scale  $W_{I,D}$  is scaled to sub-basin scale ( $W_{L,D}$ , the local sub-basin irrigation water demand). This is done in order to account for the often significant water losses between withdrawals and soil moisture replenishment. A simple user-provided scaling factor is applied:

$$W_{L,D} = \frac{W_{I,D}}{E_L}$$

The local efficiency ( $E_L$ , *local\_eff* in MgmtData.txt) represents the fraction of the withdrawn water within the sub-basin that infiltrates the irrigated soil.  $E_L$  accounts for losses in irrigation canals, ponds etc. within the sub-basin, and for on-field losses from irrigation equipment (e.g. sprinkler systems).  $W_{I,D}$  is not scaled if all withdrawals are from an unlimited source outside the model domain (i.e.  $E_L = 1$  if *irr unlimited* is y).

## Withdrawal from sources within the sub-basin

The model first attempts to withdraw water from sources within the sub-basin where the demand originated. The user specifies the proportion of water to be withdrawn from surface water and deep aquifer groundwater sources, respectively, in the MgmtData.txt file (*gw\_part*). A *gw\_part* value between 0 and 1 represents the long-term average proportion of surface and deep aquifer groundwater withdrawals for irrigation within the sub-basin. A *gw\_part* value of 0 indicates that all irrigation water exclusively comes from surface water sources, while a *gw\_part* value of 1 indicates that all irrigation water exclusively comes from deep aquifers. Based on *gw\_part*,  $W_{L,D}$  is split into the groundwater demand ( $W_{L,D,g}$ ), and the surface water demand ( $W_{L,D,s}$ ).

If any surface water demand exists, the model sequentially attempts to withdraw water from the olake, the ilake, and the main river. However, water withdrawal from olakes and ilakes is only calculated if the variable *irrdam* in MgmtData.txt is set to 1 for the sub-basin. Water withdrawals from the main river occur both from the inflow to the river reach and from the volume stored in the reach.

If the  $W_{L,D,s}$  volume is available at the source, the demanded water is withdrawn. Otherwise, only the available volume ( $V_s$ ) is withdrawn. The withdrawal can also be scaled with the user-defined parameter  $P_{I,S}$  (*pirrs* in par.txt):

$$W_{L,A,s}(1) = \min(W_{L,D,s}, V_{s1}) \times P_{I,S}$$

$$W_{L,D,s2} = W_{L,D,s} - \frac{W_{L,A,s}(1)}{P_{I,S}}$$

where  $W_{L,A,s}(1)$  is the abstracted water from the first surface water source in the sub-basin, and  $W_{L,D,s2}$  is the residual surface water demand. The residual demands ( $W_{L,D,s2}$  and below  $W_{L,D,s,1}$ ,  $W_{L,D,1}$  and  $W_{R,D2}$ ) are calculated without the  $P_{I,S}$  scaling in order to prevent erroneous source compensation due to scaling. If any demand remains, the next surface water source is probed in the same manner:

$$W_{L,A,s}(2) = \min(W_{L,D,s2}, V_{s2}) \times P_{I,S}$$

The total surface water withdrawal within the sub-basin ( $W_{L,A,s}$ ), and the remaining surface water demand ( $W_{L,D,s,1}$ ) is calculated accordingly, based on the abstractions from each source (*k*):

$$W_{L,A,s} = \sum_{k=1}^N W_{L,A,s}(k)$$

$$W_{L,D,s,l} = W_{L,D,s} - \frac{W_{L,A,s}}{P_{I,S}}$$

If any groundwater demand exists, the model withdraws a user-specified fraction of  $W_{L,D,g}$  from an unlimited source outside the model domain or if an aquifer is simulated connected to the subbasin from this aquifer. The first case conceptually represents a large deep aquifer source, which is currently outside of the model domain.  $W_{L,a,g}$  is the abstracted groundwater and  $P_{I,G}$  (*pirrg* in *par.txt*) is the groundwater withdrawal fraction:

$$W_{L,A,g} = W_{L,D,g} \times P_{I,G}$$

To simulate a more dynamic conjunctive use of groundwater and surface water sources, the model allows for compensation of remaining surface water demands from the groundwater source. This compensation is only allowed if both groundwater and surface water sources are used ( $0 < gw\_part < 1$ ), and if the *irrcomp* parameter is  $> 0$ . The *irrcomp* parameter defines the degree of compensation allowed, i.e. the fraction of the residual surface water demand which can be met through source compensation. The compensation algorithm is as follows: if any surface water demand remains ( $W_{L,D,s,1} > 0$ ) and the groundwater is not depleted, the groundwater withdrawal cycle is calculated once more using the scaled residual surface water demand. Finally, after possible source compensation, the remaining (surface) water demand at the sub-basin scale ( $W_{L,D,1}$ ) is calculated.

### Withdrawal from another sub-basin

The model can also simulate withdrawal from another sub-basin in the model domain ( $D_R$ , defined with the *regsrcid* input variable in *MgmtData.txt*). This withdrawal is calculated when the model reaches  $D_R$  in the calculation order, after possible local irrigation water abstractions in  $D_R$  and only if any irrigation demand remains in the sub-basin(s) connected to the regional source ( $W_{L,D,1} > 0$ ). For each connected sub-basin (i),  $W_{L,D,l,i}$  is scaled to represent the regional-scale demand from that sub-basin ( $W_{R,D,i}$ ). Again, a simple scaling factor is applied:

$$W_{R,D,i} = \frac{W_{L,D,l,i}}{E_{R,i}}$$

The regional efficiency ( $E_{R,i}$ , *region\_eff* in *MgmtData.txt*) represents the fraction of the withdrawn water at the regional source that reaches the connected sub-basin.  $E_{R,i}$  refers to the connected sub-basin. The regional scaling accounts for often significant water conveyance losses in large irrigation networks (in canals and dams etc.).

The total water demand from the regional source ( $W_{R,D}$ ) is then calculated as the sum of the demand from each connected sub-basin, scaled by a parameter controlling the strength of the regional connection ( $P_{regirr}$ , *regirr* in *par.txt*):

$$W_{R,D} = \left( \sum_{i=1}^N W_{R,D,i} \right) \times P_{regirr}$$

The regional demand can be met from two sources in sub-basin  $D_R$ : the lake and the main river. If the regional source sub-basin has an lake, and if the *irrdam* input variable is set to 1 for that sub-basin, the model attempts to withdraw  $W_{R,D}$  first from the lake and then the residual from the main river. If not, the model only attempts to withdraw  $W_{R,D}$  from the main river. The regional abstraction ( $W_{R,a}$ ) is limited by the volume available at the source ( $V_r$ ) and the scaling parameter  $P_{I,S}$ :

$$W_{R,A1} = \min(W_{R,D}, V_{r1}) \times P_{I,S}$$

$$W_{R,D2} = W_{R,D} - \frac{W_{R,A1}}{P_{I,S}}$$

$$W_{R,A2} = \min(W_{R,D2}, V_{r2}) \times P_{I,S}$$

$$W_{R,A} = W_{R,A1} + W_{R,A2}$$

where  $W_{R,A1}$  is the abstracted water from the first water source in  $D_R$ ,  $V_{r1}$  the volume available at the first source,  $W_{R,D2}$  the residual regional water demand after withdrawal from the first source (but prior to the  $P_{I,S}$  scaling),  $W_{R,A2}$  the abstracted water from the second water source in  $D_R$ , and  $V_{r2}$  the volume available at the second source.

### Substance concentrations of irrigation water withdrawals

The concentrations of the withdrawn water are the same as that of the irrigation water source. If the water originates from several sources, the volume-weighted concentration is calculated. If desired, the model can simulate sedimentation tanks, in which a defined fraction of the particulate phosphorous (*pp*) and organic nitrogen (*on*) settles:

$$C_{a,pp} = C_{src,pp} \times (1 - P_{cirrsink})$$

$$C_{a,on} = C_{src,on} \times (1 - P_{cirrsink})$$

where  $C_{a,a}$  is the concentration of the abstracted water after settling,  $C_{src}$  is the concentration of the source, and  $P_{cirrsink}$  is the concentration reduction fraction (*cirrsink* parameter in par.txt). To use sedimentation tanks in a region, the concentration reduction fraction needs to be set in par.txt ( $0 < cirrsink \leq 1$ ).

If the model withdraw water from an unlimited source outside the model domain, the concentration of

the water applied is assumed the same as the concentration of the receiving soil layer. Thus the concentration is assumed not to be affected by the irrigation.

## Irrigation water application

In the calculation order, the irrigation water application occurs the next time step the model reaches the sub-basin from which the demand originated (typically the following day). The water is applied to the soils at the beginning of the soil balance calculations, before the calculation of the natural processes.

The regionally abstracted water ( $W_{R,A}$ ) is first distributed to each connected sub-basin (i) according to their proportional demand, and then scaled to the local scale using the respective regional efficiency:

$$W_{R,A,i} = \frac{W_{R,A} \times W_{R,D,i} \times P_{regirr}}{W_{R,D} \times E_{R,i}}$$

For a given sub-basin, the total amount of abstracted water available at the local scale ( $W_{L,A,i,tot}$ ) is calculated and then scaled, using the local efficiency, to represent the water applied to the soil ( $W_{I,A,i}$ ):

$$W_{L,A,i,tot} = W_{L,A,i} + W_{R,A,i}$$

$$W_{I,A,i} = W_{L,A,i,tot} \times E_{L,i}$$

$W_{I,A,i}$  is then distributed onto each irrigated class in proportion to its water demand:

$$W_{I,A,j} = \frac{W_{I,A,i} \times W_{I,D,j}}{W_{I,D}}$$

$W_{I,A,j}$  is added to the soil water of class j as additional infiltration.  $W_{I,A,j}$  is added to the top soil layer. The soil layer may then hold more water than the pore volume of the soil layer. This will percolate into lower soil layers together with rain if the water is above field capacity. For unlimited irrigation:  $W_{I,A,j} = W_{I,D,j}$ ,  $P_{I,S} = 1$  and  $P_{I,G} = 1$ .

The water withdrawn from a regional source that does not reach connected sub-basins ( $W_{R,L}$ ) evaporates at the regional source ( $D_R$ ):

$$W_{R,L} = W_{R,A} - \sum_{i=1}^N W_{R,A,i}$$

Similarly, water losses due to local inefficiencies ( $W_{L,L}$ ) evaporate within the sub-basin itself. This applies to local losses from abstractions both within the sub-basin and from the regional source (where applicable):

$$W_{L,L} = W_{L,A,i,tot} - W_{I,A,i}$$

Evaporation due to regional and local inefficiencies proportionally concentrates substances in the withdrawn water. The substance concentrations in the irrigation water applications are hence higher than at the points of withdrawal (the mass remains the same while the volumes are reduced). However, if unlimited irrigation is simulated, the concentrations of the applied water are the same as in the layers to which water is added (i.e. causing no change in concentration).

## Links to file reference

Section	Symbol	Parameter/Data	File
		<i>cropid, reg</i>	<a href="#">CropData.txt</a>
		<i>mgmttype=1, subid</i>	<a href="#">MgmtData.txt</a>
Irrigation water demand		<i>imm_start, imm_end</i>	<a href="#">CropData.txt</a>
		<i>demandtype</i>	<a href="#">MgmtData.txt</a>
Non-submerged crops	$K_{CB}$ calculated from:	<i>plantday, lengthini, kcbini, lengthdev, lengthmid, kcbmid, lengthlate, kcbend</i>	<a href="#">CropData.txt</a>
	$DL_{ref}$	<i>dlref</i>	
	$P_{SSWCORR}$	<i>sswcorr</i>	<a href="#">par.txt</a>
	$P_{iwdfrac}$	<i>iwdfrac</i>	
	<i>AWC</i>	<i>wcfc1+wcfc2</i>	
Submerged crops	$WP_1, FC_1, EP_1$	<i>wcwp1, wcfc1, wcep1</i>	<a href="#">par.txt</a>
	$P_{immdepth}$	<i>immdepth</i>	
Irrigation water withdrawal		<i>irrdam, gw_part</i>	<a href="#">MgmtData.txt</a>
		<i>irrunlimited</i>	<a href="#">info.txt</a>
	$P_{I,S}$	<i>pirrs</i>	<a href="#">par.txt</a>
Irrigation inefficiencies within the sub-basin	$E_L, E_{L,i}$	<i>local_eff</i>	<a href="#">MgmtData.txt</a>
Withdrawal from sources within the sub-basin		<i>irrcomp</i>	<a href="#">par.txt</a>
	$P_{I,G}$	<i>pirrg</i>	
Withdrawal from another sub-basin	$D_R$	<i>regsrcid</i>	<a href="#">MgmtData.txt</a>
	$E_{R,i}$	<i>region_eff</i>	
	$P_{regirr}$	<i>regirr</i>	<a href="#">par.txt</a>
Substance concentrations of irrigation water withdrawals	$P_{cirrsink}$	<i>cirrsink</i>	<a href="#">par.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures	Section
irrigation_module (irrigation.f90)	calculate_irrigation	irrigation water withdrawal
		irrigation water application
	apply_irrigation	irrigation water application
	irrigation_abstraction_sink	substance concentrations of irrigation water withdrawals
	calculate_irrigation_water_demand	irrigation water demand
	calculate_kcb	
	irrigation_season	
	immersion_season	

## References

Allen, R.G., L.S. Pereira, D. Raes, and M. Smith 1998. Crop Evapotranspiration (guidelines for computing crop water requirements), FAO Irrigation and Drainage Paper, No. 56, FAO, Rome, Italy, 300 pp.

Wisser, D., S. Froking, E.M. Douglas, B.M. Fekete, C.J. Vörösmarty, and A.H. Schumann, 2008. Global irrigation water demand: Variability and uncertainties arising from agricultural and climate data sets, Geophysical Research Letters, Vol. 35, L24408, doi:10.1029/2008GL035296, 5 pp.

## Point sources

Information on point sources is located in the file [PointSourceData.txt](#). Point sources can be added to the model in different ways. They can be added as a constant source each time step, or with a constant that is changing some times during the simulation. They can also be added as time series of flow and concentration. For the first case all information of the point sources is found in the PointSourceData.txt file. For the second case the time series are given separate, and the file holds information on where the point sources are located. Simulated substances not having an concentration given in the file(s), will have concentration zero (default). This may be inappropriate for T2, water temperature.

## Constant or periodical point sources

### Nutrient

The model can handle up to three different types of point sources. They can used to simulate e.g. treatment plants, stormwater outlets, and industrial sources as separate types. All point sources have a constant flow, concentrations of total nitrogen and phosphorus, and fractions of IN and SP for a period of time. The time may be the whole simulation period, or different sources may be active during different parts of the simulation period. Point sources are added to the water in the main river.

### Sediment



Sediment point sources are handled similar to nutrient point sources. A point source has a constant flow, concentration of total suspended sediment (TS), and fraction of suspended sediments (SS). The point source may be active the whole simulation period, or different sources may be active during different parts of the simulation period. Point sources are added to the water in the main river.

### **Tracer T2 (water temperature)**

Water temperature may be added to the flow of nutrient and sediment point sources if T2 is simulated together with N and P. Water temperature point source may also be added on its own in the same way as tracer T1 in different locations of the river network.

### **Tracer T1**

Tracer T1 point source may be added in the same way as nutrient and sediment point sources. In addition, point sources of tracer T1 can be added to the local stream, the local lake, the main river or the outlet lake.

## **Time series of point sources**

Time series of point sources can be given in special file. In this case the PointSourceData.txt holds only information on which point sources exist in the model and where they can be found.

### **Nutrient and sediment**

The model can handle up to three different types of point sources. They can be used to simulate e.g. treatment plants, stormwater outlets, and industrial sources as separate types. The point sources are given as either daily, monthly or yearly time series. They are given a flow and a concentration for each nutrient species simulated, i.e. IN, ON, SP, PP, AE or SS. Point sources are added to the water in the main river.

### **Tracer T2 (water temperature)**

Water temperature may be added to the point source flow the same way as concentration.

### **Tracer T1**

Tracer T1 point source may be added to the point source flow the same way as other concentrations.

## **Negative point source**

A point source with “negative flow” denotes an abstraction of water. The abstraction of water can be

made from four different locations. The default is to remove it from the main river volume (including the queue) after all river processes have been calculated except outflow from the river. Alternatives are to abstract the water from the outlet lake volume, from an aquifer or from the main river inflow from upstream and from the river volume (and queue) proportionally. In the latter case the removal is done before any inflow to the main river is added (e.g. from upstream, point sources, or precipitation). The water is removed from the source, while the concentration is kept.

Negative point sources can be constant or periodically constant abstractions by giving a negative flow (a negative value). Alternatively they can also be given as time series. In the latter case the flow time series is positive but it is defined as a negative point source in PointSourceData.txt.

## Links to file reference

Section	Parameter/Data	File
Constant or periodical point sources - Nutrient	<i>subid, ps_vol, ps_type, ps_tpconc, ps_tnconc, ps_infrac, ps_spfrac, fromdate, todate</i>	<a href="#">PointSourceData.txt</a>
Constant or periodical point sources - Sediment	<i>subid, ps_vol, ps_type, ps_tsconc, ps_ssfrac, fromdate, todate</i>	
Constant or periodical point sources - Tracer T2	<i>subid, ps_vol, ps_type, ps_t2, fromdate, todate</i>	
	<i>subid, ps_vol, ps_type=0, ps_t2, fromdate, todate, ps_source</i>	
Constant or periodical point sources - Tracer T1	<i>subid, ps_vol, ps_type, ps_t1, fromdate, todate</i>	
	<i>subid, ps_vol, ps_type=0, ps_t1, fromdate, todate, ps_source</i>	
Time series point sources	<i>psid, subid, ps_type, ps_source</i>	<a href="#">PointSourceData.txt</a> and <a href="#">PSTIMESeries.txt</a>
Negative point source	<i>subid, ps_vol, fromdate, todate, ps_source</i>	<a href="#">PointSourceData.txt</a>
	<i>psid, subid, ps_source</i>	<a href="#">PointSourceData.txt</a> and <a href="#">PSTIMESeries.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">datamodule (data.f90)</a>	<code>load_pointsourcedata</code>
	<code>read_pointsourcedata</code>
	<code>get_current_pointsources</code>
<a href="#">npc_surfacewater_processes (npc_sw_proc.f90)</a>	<code>add_point_sources_to_main_river</code>
<a href="#">surfacewater_processes (sw_proc.f90)</a>	<code>point_abstraction_from_main_river</code>
	<code>point_abstraction_from_outlet_lake</code>
<a href="#">tracer_processes (t_proc.f90)</a>	<code>add_tracer_point_source_to_river</code>
	<code>add_tracer_point_source_to_lake</code>

## Water abstraction and transfer

Water transfer can be simulated by HYPE in different ways. One way is to represent abstraction by defining [negative point source](#) discharge. Another method is to use the [bifurcation functionality](#) that defines a branch through which the water is transferred to a downstream receiving subbasin. A third way is to define water transfer through water management (abstracting from subbasins with outlet

lakes and transferring to any other subbasin). The negative point source method has the disadvantage as a water transfer that the concentration of the water is specified by the user and not the concentrations of the water abstracted, but the abstraction can be located at different points throughout the catchment. The bifurcation method has the limitation that the receiving subbasin must be downstream. The management method can only take water from an outlet lake, and the water transfer will be delayed one time step. In all methods, the amount of water available on any given day will limit the amount that can be withdrawn or transferred.

## Water transfer through bifurcation

Water transfer through bifurcation lets you divert a part of the outflow from a subbasin through a branch to another downstream subbasin specified in BranchData. The flow diverted may be defined by BranchData parameters (fraction, max-, or minflow). The water transfer can alternatively be given as a recorded time series of demanded water transfer (given as dwtr in Xobs). This water will then be taken from the ordinary simulated outflow of the subbasin if there is enough flow, the remaining flow will go through the main channel. A third method, if you have an outlet lake, is to use two outlets. The flow of outlet 2 can be defined as the flow given by dwtr.

See also the general description of [bifurcations](#) and [Outlet lake with two outlets](#).

## Water transfer through negative point source

Abstraction of water can be defined as a [negative point sources](#). The abstracted water may be constant (or constant at different values for different periods), and thus the same amount of water is removed from the model every time step or given as a time series. The abstraction can be made from different points in the river network within the subbasin. The water is removed while the concentration is kept in the water remaining in the river. To simulate water transfer within the model domain, the same amount of water need to be added as a (positive) point source in another subbasin. This subbasin do not have to be downstream, since its (constant) flow and concentrations is defined in the file beforehand. However, there is no direct connection between the subbasins in which water is abstracted and released, respectively. In order to preserve the water balance, the user must define equally large flows abstracted and added. Since the concentration of the negative point source is defined in the file beforehand, the connection to the quality of the water abstracted water is not automatically preserved.

## Water transfer through water management

Water transfer between two subbasins can be defined through water management ([MgmtData.txt](#)) if the source is an outlet lake. The source subid and subid of the receiving subbasin is defined together with a constant flow. The water is taken from the outlet lake of the source subbasin and is given the concentration of the lake. The water is then transported to the other subbasin's main river and arrive there the next time step. The delay in flow is necessary to handle transfer upstream within a catchment. Several water transfer can be specified for the same source or receiving subbasin.

Alternatively a demanded flow time series from Xobs can be used instead of a constant flow. In this case only one water transfer can be specified for each source lake.

## Links to file reference

Section	Parameter/Data	File
Water transfer through bifurcation	<i>ALL</i>	<a href="#">BranchData.txt</a>
	<i>ldtype=5 and 6</i>	<a href="#">LakeData.txt</a>
	<i>lakeid, rate, exp, deltaw0, qprod1, qprod2, datum1, datum2, qamp, qpha, regvol, maxQprod, minflow, obsflow</i>	
	<i>dwtr</i>	<a href="#">Xobs.txt</a>
Water transfer through negative point source	<i>ps_vol, fromdate, todate, ps_source</i>	<a href="#">PointSourceData.txt</a>
	<i>psid, subid, ps_type, ps_source</i>	<a href="#">PointSourceData.txt</a> and <a href="#">PSTIMESeries.txt</a>
Water transfer through water management	<i>mgmttype=2</i>	<a href="#">MgmtData.txt</a>
	<i>subid, receiver, flow</i>	
	<i>dwtr</i>	<a href="#">Xobs.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures	Section
<a href="#">datamodule (data.f90)</a>	<a href="#">load_pointsourcedata</a>	negative point source
	<a href="#">load_management_data</a>	water management
<a href="#">npc_surfacewater_processes (npc_sw_proc.f90)</a>	<a href="#">add_point_sources_to_main_river</a>	negative point source
<a href="#">surfacewater_processes (sw_proc.f90)</a>	<a href="#">calculate_outflow_from_outlet_lake</a>	bifurcation
	<a href="#">point_abstraction_from_main_river</a>	negative point source
	<a href="#">add_point_sources_to_main_river</a>	
	<a href="#">point_abstraction_from_outlet_lake</a>	water management
	<a href="#">add_water_transfer_to_main_river</a>	
	<a href="#">water_transfer_from_outlet_lake</a>	

## Dams

Regulation of flow through dams is described in the Chapter about [Rivers and lakes](#). Dams of different purposes and regulation management can be simulated. See details in the sections on [Simple outlet lake or dam](#) and the special case of an [Outlet lake with two outlets](#).

# Deep processes

There are two model options for processes deep below ground; regional groundwater flow and aquifers. The regional groundwater flow model simulates groundwater flow between subbasins, but no explicit storage of deep groundwater. The aquifer model simulates aquifer storage and delay before regional groundwater reaches the destination.

## Regional groundwater flow

### Soil regional groundwater flow creation

Regional groundwater flow from a subbasin is calculated before the subbasin and class loops. This water is added to the downstream subbasin (in the class-loop part of the code) and/or to the outlet lake of the present subbasin (in the routing calculation part of the code). A subbasin dependent variable (*grwtolake*) governs the allocation between them. The outflow from a subbasin is calculated as a percentage of the water-retention capacity with a recession coefficient (*rcgrw*) e.g. for the lowest soil layer (*k*) the groundwater outflow for a class becomes:

$$q_{grw}(t) = rcgrw \times \left( soil(k) - wp(k) - fc(k) \right) \quad soil(k) > fc(k) + wp(k)$$

Groundwater flows are summarized for all classes in proportion to their areas and converted to volume flow out of the subbasin.

### Soil addition of regional groundwater flow

The groundwater flow to be added to the bottom layer (and mixed) is added in the class loop portion of the code. If the layer then becomes full of water, it will be pressed into the layer above and mixed, etc. Please note that if downstream subbasin is much smaller than the upstream subbasin, groundwater inflows can become large.

### Regional groundwater to outlet lake

With an input variable in GeoData.txt, an outlet lake may be flagged to receive inflow from the regional groundwater. The coefficient (*grwtolake*) specifies the portion of the produced regional groundwater flow which is led to the subbasins outlet lake. The part of the groundwater flow that is to go to the outlet lake is added to the lake volume in the routing part of the code.

### Links to file reference

Section	Symbol	Parameter/Data	File
Soil regional groundwater flow creation		<i>grwolake, grwdown</i>	<a href="#">GeoData.txt</a>
	<i>rcgrw</i>	<i>rcgrw</i>	<a href="#">par.txt</a>
	<i>wp(k)</i>	<i>wcwp, wcwp1, wcwp2, wcwp3</i>	
	<i>fc(k)</i>	<i>wcfc, wcfc1, wcfc2, wcfc3</i>	

Links to relevant procedures in the code

Modules (file)	Procedures	Sections
<a href="#">regional_groundwater_module</a> ( <a href="#">regional_groundwater.f90</a> )	<code>calculate_regional_groundwater_flow</code>	soil regional groundwater flow creation
	<code>calculate_soillayer_groundwaterflow_removal</code>	soil regional groundwater flow creation
	<code>add_regional_groundwater_flow_to_soil</code>	soil addition of regional groundwater flow
	<code>add_regional_groundwater_flow_to_olake</code>	regional groundwater to outlet lake

Aquifers

Aquifers are large water bodies covering several subbasins and located below the soillayers simulated by HYPE. They recieve water by percolation from land and river classes, and they return flow to a river (Figure 1). They can also be used as a water source for abstraction. The aquifer is a large water body that can have a passive volume partaking in mixing but not return flow (Figure 2).



Figure 1 Water flows of aquifer.

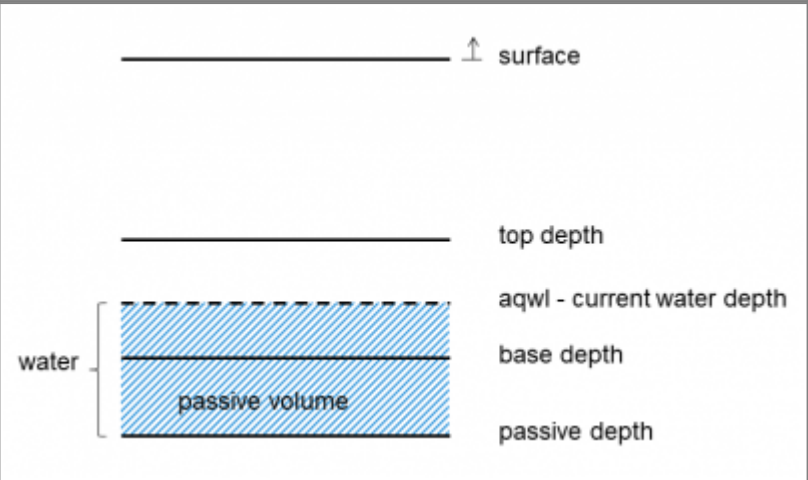


Figure 2 Structure of an aquifer.

Deep percolation

Deep percolation is determined by a recession coefficient ( $rc_{grw}$ ). The recession coefficient is set based on the recession parameter ( $rcgrwst$ ) which is soil type dependent, but if it is not set the general parameter ( $rcgrw$ ) is used instead. The recession coefficient can also be regionally adjusted by the correction parameter ( $aqpercorr$ ).

$$rc_{grw} = rcgrw \left( \text{or } rcgrwst \right) \times \left( 1 + aqpercorr \right)$$

Deep percolation from lowest soil layer in model is determined by the recession coefficient and the available water for runoff in the soil layer ( $qperc$  (mm/timestep)).

$$q_{perc}(t) = rc_{grw} \times (sm - fc - wp) \quad sm > fc + wp$$

Deep percolation from main river in model is determined by the recession coefficient and the water in the river ( $qperc$  (m3/timestep)).

$$q_{perc}(t) = rc_{grw} \times (waterdepth)$$

The water is removed from river water and the queue in proportion to their relative volumes.

## Deep percolation delay and aquifer recharge

Deep percolation from soil and rivers in all recharge subbasins of an aquifer is added together. This total deep percolation ( $q_{perc}$ ) is delayed before it recharges the aquifer ( $q_{rech}$ ).

$$q_{rech}(t) = (1 - coeff) \times q_{perc}(t) + coeff \times q_{rech}(t-1)$$

$$coeff = e^{-\frac{1}{delayts}}$$

The delay property ( $delayts$ , in timesteps per day) is specified for each aquifer ( $delay$  in days). It can be adjusted by the parameter region dependent model parameter  $aqdelcor$ .

## Return flow from aquifer

Outflow from the aquifer depends on the water in the aquifer ( $vol$ ) (not including passive volume) and a rate. The rate is determined by a aquifer property ( $retrate$ ) and a regional adjustment parameter ( $aqretcor$ , parameter region dependent).

$$q_{out} = retrate \times (1 + aqretcor) \times vol$$

The outflow may be divided between several receiving subbasins ( $retfrac$ ). The receiving subbasins may or may not be different from the subbasin recharging the aquifer. The specified fraction of outflow is added to the inflow to the main river of these subbasins.

## Nutrient processes related to aquifers

For most models the organic nitrogen, particulate phosphorus, and organic carbon in water percolating from soil and river stay in the soil/river, while temperature, inorganic nitrogen and SRP follow the water to the aquifer. In the case of a model using the simplified soil model based on travel time through soil and half life of pollutants (i.e. using load as input to HYPE) all substances follow the water to the aquifer.

Inorganic nitrogen is subjected to denitrification in the aquifer (see Chapter [Nitrogen and phosphorus in land routines - Denitrification](#)). For other substances an exponential decay can be applied in the aquifer. The process is governed by the model parameter *decay* (e.g. *ppdecaq*), which denote the half life of the substance in days. The decay will decrease the concentration (*conc*) of the substance in the aquifer.

$$conc = conc \times 2^{-\frac{ts}{decay}}$$

where *ts* is length of time step in days.

## Abstraction from aquifers

Aquifers can be used for abstraction of water. The abstraction is defined through PointSourceData for a subbasin. The water is removed from the aquifer below that subbasin. Water is taken from the aquifer water volume. This means that also water from the passive volume can be had.

## Links to file reference

Section	Symbol	Parameter/Data	File
Aquifers		<i>aquid, subid, parreg</i>	<a href="#">AquiferData.txt</a>
		<i>basedepth, passivedep, inidepth, porosity, area</i>	
Deep percolation	<i>rc<sub>grw</sub></i> calculated from	<i>rcgrw, rcgrwst, aqpercrr</i>	<a href="#">par.txt</a>
	<i>wp</i>	<i>wcwp, wcwp1, wcwp2, wcwp3</i>	
	<i>fc</i>	<i>wcfc, wcfc1, wcfc2, wcfc3</i>	
Deep percolation delay and aquifer recharge		<i>recharge</i>	<a href="#">AquiferData.txt</a>
	<i>delay</i>	<i>delay</i>	<a href="#">par.txt</a>
	<i>aqdelcorr</i>	<i>aqdelcorr</i>	
Return flow from aquifer	<i>retrate, retfrac</i>	<i>retrate, retfrac</i>	<a href="#">AquiferData.txt</a>
	<i>aqretcorr</i>	<i>aqretcorr</i>	<a href="#">par.txt</a>
Nutrient processes related to aquifers		<i>topdepth, temp, conc_IN, conc_SP</i>	<a href="#">AquiferData.txt</a>
		<i>denitaq, hsatINs, wcep, wcfc, wcwp</i>	<a href="#">par.txt</a>
	<i>decay</i>	<i>ondecaq, spdecaq, ppdecaq, ocdecaq, ssdecaq, aedecaq, t1decaq</i>	<a href="#">par.txt</a>

## Links to relevant procedures in the code



Modules (file)	Procedures	Sections
regional_groundwater_module (regional_groundwater.f90)	calculate_aquifer	deep percolation
	calculate_regional_groundwater_flow	
	calculate_soillayer_groundwaterflow_removal	
	calculate_river_groundwaterflow_removal	
	calculate_aquifer_percolation	
	calculate_percolation_delay	deep percolation delay and aquifer recharge
	calculate_delayed_water	
	calculate_aquifer_outflow	return flow from aquifer
	add_aquifer_flow_to_river	
	calculate_current_flow_from_aquifer	return flow from aquifer
		nutrient processes related to aquifers
	set_percolation_concentration	nutrient processes related to aquifers
npc_soil_processes (npc_soil_proc.f90)	soil_denitrification	nutrient processes related to aquifers
general_functions (general_func.f90)	exponential_decay	



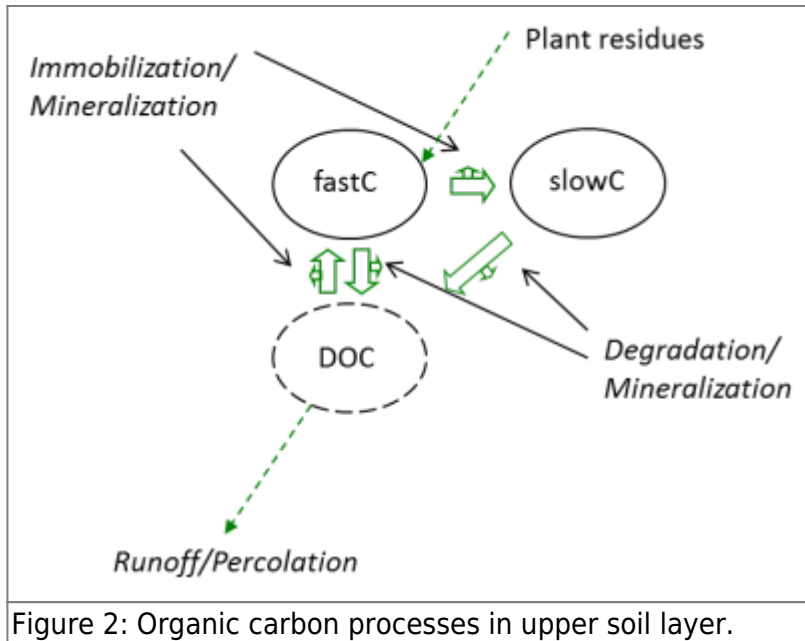


Figure 2: Organic carbon processes in upper soil layer.

## Production of humusC from fastC

Some of the litter fall is converted into humus. For HYPE this means that fastC (the pool where litter fall was added) is transformed to humusC in the uppermost soil layer. In Figure 2 the pool of humusC is denoted slowC because of its slower transformation rates.

The other soil layers ( $k$ ) also have a transition from fastC to humusC. The loss of fastC does not all end up in the humusC pool, but a proportion (parameter *minc*) is mineralized in the process. The transformation (*fasttohumus*, mg/m<sup>2</sup>/d) depends on soil moisture (*smfcn*) and temperature (*tmpfcn*), amount of OC in the pool (*fastC*) and a vegetation dependent parameter *klh*.

$$\text{fasttohumus}(k) = klh \times tmpfcn(k) \times smfcn(k) \times fastC(k)$$

## Turnover of fastC

Turnover of fastC is a sink for fastC and a source of dissolved OC in soil water (DOC) in all soil layers ( $k = 1-3$ ). The loss of fastC does not all go to the OCpool, but a proportion (parameter *minc*) is mineralized. Turnover (*transfC*, mg/m<sup>2</sup>/d) depends on a general parameter (*klo*), the temperature function (*tempfcn*), humidity function (*smfcn*) and the pool of fastC (*fastC*).

$$\text{transfC}(k) = klo \times tempfcn(k) \times smfcn(k) \times fastC(k)$$

In dry conditions a transfer in the opposite direction can also occur. The transformation of DOC to fastC is a decrease of DOC and a source of fastC in all soil layers ( $k = 1-3$ ). The loss of DOC is not all ending up in fastC but a proportion (parameter *minc*) is mineralized. Turnover (*doctofast*, mg/m<sup>2</sup>/d) depends on a general parameter (*kof*) and the pool of DOC (*OCpool*). The transfer is limited that the soil layer temperature must be less than 5 °C, the soil moisture (*sm*) must be less than field capacity and moisture function (*smfcn*) must be less than the parameter *koflim*.

$$\text{doctofast}(k) = kof \times OCpool(k)$$

## Turnover of humusC

Turnover of humusC is a sink for humusC and a source of DOC in all soil layers ( $k = 1.3$ ). The turnover rate of humusC is lower than that of fastC, why it is sometimes called slowC (e.g. in Figure 2). The loss of humusC does not all go to the soil water OC, but a proportion (parameter *minc*) is mineralized. Turnover (*transhC*, mg/m<sup>2</sup>/d) depends on a general parameter (*kho*), temperature function (*tempfcn*), humidity function (*smfcn*) and the pool of humusC (*humusC*).

$$transhC(k) = kho \times tempfcn(k) \times smfcn(k) \times humusC(k)$$

## Percolation

Organic carbon is lost from the soil water as it percolates down through the soil layers and where it is dissipated to become a regional groundwater flow. The decrease in concentration of percolating water during transport between the soil layers depends on soil moisture and temperature and a calibration parameter.

$$conc_{perc} = conc_{soillayer} \times (1 - par \times tmpfcn \times smfcn)$$

The soil moisture function and temperature function are the general functions described for soil processes. Percolation uses the nitrogen and phosphorus coefficients for the soil moisture function, not the parameters that the OC transformations uses. The parameter, *par* in the equation, is *kcgwreg* for regional groundwater flow formation and *koc* for percolation between soil layers. Both are general parameters.

## Delay of organic carbon in runoff

The OC transported by surface and groundwater runoff and tile drainage (*runoffC*) is collected in a temporary storage pool (*relpool* (kg/km<sup>2</sup>)).

$$relpool = relpool + runoffC$$

From the temporary pool organic carbon is released (*release* (kg/km<sup>2</sup>)) and then transported to the local river depending on the size of the total runoff (*runoff* (mm)). The parameters *ocfldelx* and *ocfldele* are general parameters.

$$release = MIN \left( relpool, relpool \times \left( runoff / ocfldelx \right)^{ocfldele} \right)$$

The released OC give the current OC concentration of runoff.

## Links to relevant procedures in the code

Modules (file)	Procedures	Section
npc_soil_processes (npc_soil_proc.f90)	initiate_soil_npc	initial values
	soil_substance_processes	production of humusC from fastC, turnover
	soil_carbon_pool_transformations	turnover
	doc_percolation_reduction	percolation
	carbon_runoff_delay	delay of organic carbon in runoff

## Riparian zone

Runoff from soil may flows through a riparian zone before it reaches the local river. Surface runoff and drainage water from drainage pipes reaches the local river without passing through the riparian zone. In the riparian zone the levels of OC are affected, while flows remain unchanged. The change in OC depends on soil temperature, class altitude (*elev* (in masl)), the water table (*gwat*) and its recent change, season and soil moisture (*sm*). The runoff concentration (*conc(i)*) of each soil layer (*k*) increases with the factor:

$$f(k) = 1 + ripz \times tmpfcn(k) \times \left( \frac{elev}{100} \right) \times f_{grw} \times f_{season} \times f_{sm}$$

$$conc(k) = f \times conc(k), \quad k = 1..3$$

The temperature function (*tmpfcn*) is the usual of soil processes (see above). The following equations describe the other process functions:

$$f_{grw} = e^{ripe \times gwat}$$

$$f_{season} = \begin{cases} rips & autumn \\ 1 & otherwise \end{cases}$$

$$f_{sm} = \begin{cases} 0 & sm \leq wp \\ f_2(sm) & wp < sm < pw \\ satact & sm \geq pw \end{cases}$$

$$f_2(sm) = \begin{cases} \min \left( 1, satact + \frac{(1-satact) \times pw - sm}{d \times \Theta_{upp}}, \frac{sm - wp}{d \times \Theta_{low}} \right) & rising\ grw \\ \min \left( 1, satact + \frac{(1-satact) \times pw - sm}{d \times \Theta_{upp}}, \frac{satact \times sm - wp}{d \times \Theta_{low}} \right) & sinking\ grw \end{cases}$$

The activation of riparian zone processes is based on land use. It is primarily thought to act on forest runoff. The land use dependent parameter *ripz* determines the overall level of increase in concentration in the riparian zone, and if set to zero no riparian zone processes are used. In addition, two general parameters can influence the effect of the riparian processes; *ripe* which determines the groundwater level dependence, and *rips* which determines the seasonal influence. Season division is determined by ten-day and twenty-day averages of air temperature (T10, T20). Autumn is defined as the period when T10 is less than T20. The soil moisture function is different for an increasing (rising) and sinking ground water table (Figure 3). It contains coefficients  $\Theta_{upp}=0.12$ ,  $\Theta_{low}=0.08$  and saturation (*satact* = 0.6). It depends on the soil moisture of all layers together (*sm*) and the water-holding capacity parameters; *wp* - wilting border and *pw* - total pore volume, in fractions of total soil layer thickness (*d*).

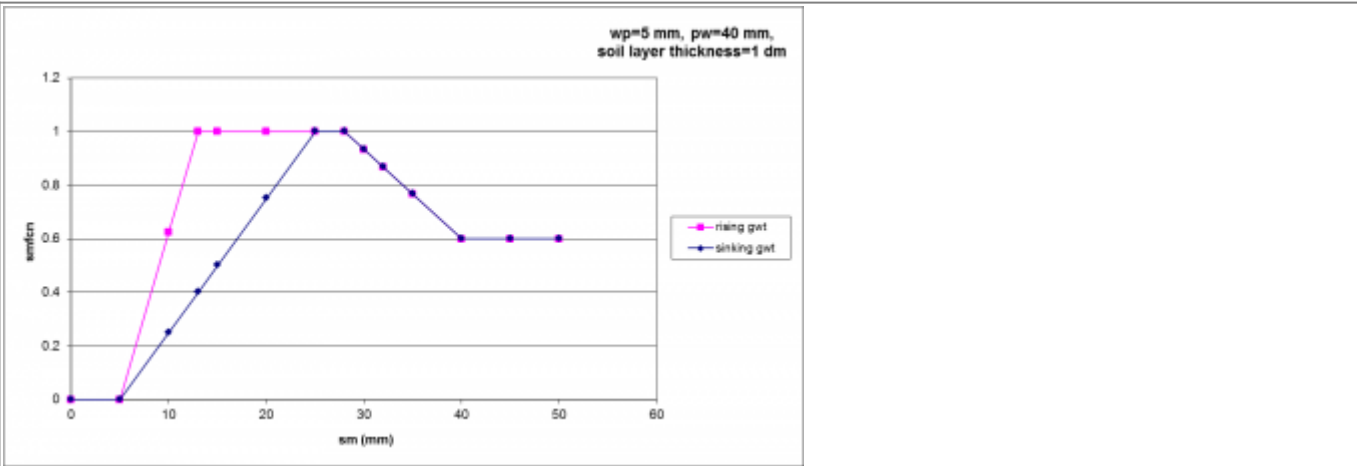


Figure 3: Example of riparian zone soil moisture function, and the dependence on changes in the groundwater levels.

Links to relevant procedures in the code

Modules (file)	Procedures
npc_soil_processes (npc_soil_proc.f90)	class_riparian_zone_processes
	riparian_moisture_factor

Rivers and lakes

The initial organic carbon concentration in rivers are assumed to be zero, while the lakes' concentration are set by the user. The parameter, *tocmean*, is lakeregion dependent, but can also be set for each lake separately.

Primary production and mineralization

Primary production is a source of organic carbon in rivers and lakes, while mineralization is a sink. Primary production and mineralization are calculated the same way as for nitrogen, but with its own calibration parameter (*wprodc*). The equations are repeated below. The production/mineralization depend on temperature and total phosphorus and lake area (*area*). The potential carbon transformation (*minprodCpot*, kg / day) is proportional to the potential nitrogen transformation (*minprodNpot*, see also [NP section](#)) with a transformation rate that depends on the carbon-nitrogen ratio (*NCratio* = 5.7). The calculated mineralization of organic carbon is limited to a maximum of 50%

of the available OC pool. If phosphorus is not modelled a long-term average total phosphorus concentration as a lake region dependent parameter (*tpmean*) is used. If set, the long-term average concentration of phosphorus is reduced by the general parameter *limsedPP* before using it in the concentration function thus reducing the production/mineralisation of OC. The water depth (*depth*) is the lake depth, and for the river the depth calculated [above](#).

$$tmpfcn1 = \frac{watertemp}{20.}$$

$$tmpfcn2 = \frac{(T_{10} - T_{20})}{5.}$$

$$tmpfcn = tmpfcn1 \times tmpfcn2$$

$$TPfcn = \frac{(TPconc - limsedPP)}{(TPconc - limsedPP + halvesatTPwater)}$$

$$minprodNpot = wprodc \times TPfcn \times tmpfcn \times area \times depth$$

$$minprodCpot = minprodNpot \times NCratio$$

## Sedimentation

Sedimentation in lakes is a sink for OC and works the same way as for organic nitrogen and particulate phosphorus. Sedimentation (*sedOC*, kg/day) is calculated as a function of OC concentration in lake water (*conc*) and lake area (*area*). The settling velocity parameter *sedoc* is general or can be specified for each lake.

$$sedOC = sedoc \times conc \times area$$

## Links to relevant procedures in the code

Modules (file)	Procedures	Sections
npc_surfacewater_processes (npc_sw_proc.f90)	substance_processes_in_river	primary production and mineralization
	oc_production_mineralisation	
	calculate_lake_tpmean	
	calculate_river_tpmean	
	substance_processes_in_lake	primary production and mineralization
	lake_sedimentation	sedimentation

## Links to file reference

Section	Symbol	Parameter/Data	File
Sources of organic material		<i>resc</i>	<a href="#">CropData.txt</a>
		<i>litterdays</i>	<a href="#">par.txt</a>
Soil processes		<i>humusc1, humusc2, humusc3, fastc1, fastc2, fastc3, koflim</i>	<a href="#">par.txt</a>
	$\theta_{low}$	<i>ocsoilslp</i> or 0.08	
	<i>satact</i>	<i>ocsoilsat</i> or 0.6	
	<i>minc, klh, klo, kof, kho</i>	<i>minc, klh, klo, kof, kho</i>	
	<i>par</i>	<i>kcgwreg</i> or <i>koc</i>	
	<i>ocfldelx, ocfldele</i>	<i>ocfldelx, ocfldele</i>	
Riparian zone	<i>elev</i>	calculated from <i>mean_elev</i> and <i>dhs1c_nn</i>	<a href="#">GeoData.txt</a>
	<i>ripz, ripe, rips</i>	<i>ripz, ripe, rips</i>	<a href="#">par.txt</a>
	<i>wp</i>	calculated from <i>wcwp, wcwp1, wcwp2, wcwp3</i>	
	<i>pw</i>	calculated from <i>wcwp, wcwp1-wcwp3, wcfc, wcfc1-wcfc3, wcep, wcep1-wcep3</i>	
	<i>d</i>		
			<a href="#">GeoClass.txt</a>
Rivers and lakes	<i>area, lakeregion</i>		<a href="#">GeoData.txt</a>
	<i>tocmean, wprodc, sedoc</i>	<i>tocmean, wprodc, sedoc</i>	<a href="#">par.txt</a> or <a href="#">LakeData.txt</a>
	<i>limsedpp</i>	<i>limsedpp</i>	<a href="#">par.txt</a>
		<i>tpmean</i>	
	<i>halfsatTPwater</i>	<i>hsatTP</i>	



# Tracers

HYPE can simulate tracers in addition to water flows, nutrients and organic carbon. Tracers are simulated as a substance given in [info.txt](#) and follow the water as a concentration. Two tracers are implemented;

- T1 is a general tracer that can have different characteristics that is governed by model parameters
- T2 is water temperature

## General tracer (T1)

T1 is the substance name for a general tracer. It can be used to simulate different substances, e.g. pathogens or O18 isotope. The tracer is simulated as a concentration in HYPE's water bodies, e.g. snow, soil water, surface water. In addition T1 can be simulated as adsorbed to soil, as a store above ground, and in river sediments.

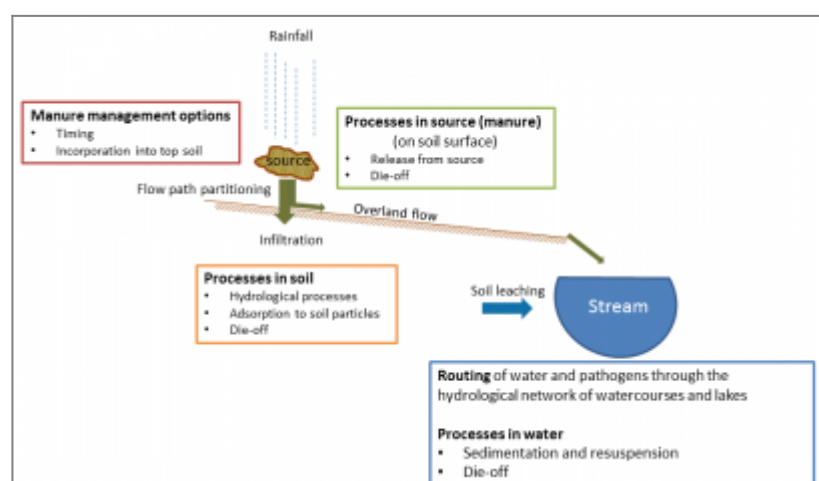


Figure 1: Example of T1 model used for simulating pathogens.

The following table shows how the unit of different tracers relates to each other and how they are related to input and output parameters and variables. The general unit (U) is used in tables of parameters and input data where the substance is not defined.

Variabel/parameter	General unit (U)	Weight (kg)	Number of pathogens (#)
Amount in manure/residuals ( <i>t1amount</i> )	U/ha	kg/ha	#/ha
Point source concentration ( <i>ps_t1</i> )	mU/m <sup>3</sup> or $\mu$ U/L	mg/L	thousandth part/m <sup>3</sup> or millionth part/L
Concentration of water as incoming source ( <i>cpt1</i> , <i>t1leaksoil</i> / <i>t1leakluse</i> )	mU/m <sup>3</sup> or $\mu$ U/L	mg/L	thousandth part/m <sup>3</sup> or millionth part/L
Initial concentration parameters ( <i>init1</i> , <i>init1sw</i> )	mU/m <sup>3</sup> or $\mu$ U/L	mg/L	thousandth part/m <sup>3</sup> or millionth part/L
Simulated amount T1 in or above soil ( <i>aT11</i> , <i>aT12</i> , <i>aT13</i> , <i>sT11</i> , <i>sT12</i> , <i>sT13</i> , <i>T1sf</i> )	U/km <sup>2</sup>	kg/km <sup>2</sup>	#/km <sup>2</sup>

Variabel/parameter	General unit (U)	Weight (kg)	Number of pathogens (#)
Simulated amount T1 in river sediment ( <i>Tsmr</i> , <i>Tslr</i> )	U	kg	#
Simulated sedimenting amount T1 in outlet lake ( <i>ola1</i> )	U	kg	#
Simulated concentration of water ( <i>coT1</i> , <i>ceT1</i> , <i>csT1</i> , <i>ccT1</i> , <i>clT1</i> , <i>Tcr1</i> , <i>Tcr2</i> , <i>Tcr3</i> , <i>Tcrd</i> , <i>Tcrs</i> )	mU/m3 or $\mu$ U/L	mg/L	thousandth part/m3 or millionth part/L
Observed concentration ( <i>reT1</i> )	mU/m3 or $\mu$ U/L	mg/L	thousandth part/m3 or millionth part/L

## Sources

There are four different ways to introduce the tracer T1 to the model:

1) T1 may be added as a concentration in precipitation (*cpt1*). This concentration is added as a time series in [Xobs.txt](#).

2) Point sources may be a source of T1 to surface waters. Point sources of tracer T1 can be added to the local stream, the local lake, the main river or the outlet lake.

3) A pool laid down on land can be a source of the tracer. The pool is defined similar to the manure for nutrients (*t1amount*). It can be added at a specific point in time or equally distributed over a period of time. The pool may be tilled down into the top soil layers also in similar fashion as manure.

4) The tracer can be introduced to HYPE in the form of typical concentrations for leakage from different land-uses and/or soil types. The concentration of the runoff from a class (*conc*) is calculated as the product of two model parameters:

$$conc = t1leakluse \times t1leaksoil$$

One parameter is land use dependent (*t1leakluse*) and one is soil type dependent (*t1leaksoil*).

The source of tracer by a typical concentration (introduction way 4) is not compatible with the sources concentration of precipitation (introduction way 1) and pool on the ground (introduction way 3).

## Processes

In its basic state, T1 act as a tracer passively following the water through the model (except for evaporation). By supplying input data and parameter values this behaviour may be changed to simulate a non-conservative substance, e.g., a patogen.

## Evaporation

As default T1 do not follow evaporating water. This is an exception from the basic assumption that T1 is a conservative tracer if no parameters are set. The tracer will follow evaporating water (with same concentration as the water it is evaporating from) if the model parameter *t1evap* is set to one. Default is zero. It is possible to set a value between zero and one so that the tracer partly follows evaporating water.

## Release from above ground storage

T1 can be released from a storage above ground (e.g. manure) by rain and snowmelt (flow  $q$  (mm/day)). The released tracer will follow the flow, as a concentration, and depending on the fate of the flow, the tracer may follow surface runoff, infiltration or other path ways. The release is governed by the model parameter  $t1rel$  (1/mm). A fraction ( $a_{rel}$ ) of the total amount of tracer in the store above ground is released and follows the flow according to equation:

$$a_{rel} = 1 - e^{-t1rel \times q}$$

## Exponential decay

Patogens grow and die over time. To simulate this, HYPE supplies a process for the combined effect as an exponential decay. The process is governed by the model parameter  $t1expdec$ , which denote the half-life of the tracer in days. Exponential decay is applied to tracer T1 in most forms; in soil water, river, lakes, the above ground storage, tracers adsorbed to soil and tracers in river sediment.

## Adsorption/desorption

Tracer adsorption/desorption to/from soil particles is calculated based on an equilibrium concentration ( $c_{eq}$  (e.g. number of pathogens/L or U/L)) of soil water. The equilibrium concentration is assumed to be reached within one time step of the model, typically one day, and this gives the amount of tracer adsorbed or desorbed during the time step is calculated. The equilibrium concentration ( $c_{eq}$ ) is calculated from the amount of tracers and soil characteristics. The equation is:

$$c_{eq} = \frac{t1freuc \times Atot}{\left( v + t1freuc \times b_{density} \times d \right)}$$

where  $Atot$  is the sum of tracer adsorbed and solved in soil water of the soil layer (U/km<sup>2</sup>),  $v$  the volume of water in the soil layer (mm),  $b_{density}$  is soil density (1300 kg soil/m<sup>3</sup>), and  $d$  is the thickness of the soil layer (m). The general model parameter  $t1freuc$  is an equilibrium concentration coefficient. In the case of pathogens, the model parameter describes the number of pathogens adsorbed to one kg of soil per pathogens in one litre of soil water (or in general unit (U/kg soil)/(U/L) ).

## Sedimentation/resuspension in river

Sedimentation or resuspension of tracers in rivers is calculated the same way as particulate phosphorus (described [here](#)) and suspended sediments (described [here](#)). There are two alternative models. For the original the process depends on the current flow ( $q$ ) in the river in relation to an reduced bankful flow ( $q_{bank}$ ) and the general model parameter  $t1sedexp$ . Bankful flow is the flow when the river is filled to the brim. This flow is calculated as the second largest simulated flow in the last year. It has been adjusted with the value 0.7 or by the parameter  $qbank$ . The fraction of the tracer that is resuspended or sedimented is determined by the equation:

$$a_{sres} = \max \left( -1, \min \left( 1, \frac{q_{bank} - q}{q_{bank}} \frac{t1sedexp}{t1sedexp} \right) \right)$$

For low flow, the fraction ( $a_{sres}$ ) is positive and sedimentation ( $F_{sed}$  (U/day)) occur, for high flow the fraction is negative and there will be resuspension from the sediments instead ( $F_{resusp}$  (U/day)):

$$F_{sed} = a_{sres} \times c_{river} \times v_{river}, a_{sres} > 0$$

$$F_{resusp} = -a_{sres} \times A_{sed}, a_{sres} < 0$$

where  $c_{river}$  is the concentration of T1 in river (U/m3),  $v_{river}$  is the volume of river (m3), and  $A_{sed}$  is amount of tracer T1 in the sediment (U).

An alternative model is the Bagnold equation (see sediment). The tracer sedimentation/resuspension with this model uses T1 specific parameters to calculate the maximum suspended T1 concentration in the river, general parameters *suspconT1* and *suspexpT1*.

Sedimentation in lakes

Sedimentation of tracer T1 in lakes is calculated the same way as e.g. particular phosphorus. The sedimentation is goverend by sink velocity (m/d), which is given by the general model parameter *t1sedvel*.

Links to file reference

Section	Symbol	Parameter/Data	File
General tracer (T1)		<i>init1, init1sw</i>	<a href="#">par.txt</a>
Sources		<i>cpt1</i>	<a href="#">Xobs.txt</a>
		<i>ps_t1</i>	<a href="#">PointSourceData.txt</a>
		<i>t1amount</i>	<a href="#">CropData.txt</a>
	<i>t1leakluse, t1leaksoil</i>	<i>t1leakluse, t1leaksoil</i>	<a href="#">par.txt</a>
Processes	<i>t1rel, t1sedexp, t1freuc</i>	<i>t1rel, t1sedexp, t1freuc</i>	<a href="#">par.txt</a>
		<i>t1evap, t1expdec, t1sedvel, qbank, suspcont1, suspexpt1</i>	
	<i>d</i>		<a href="#">GeoClass.txt</a>

Links to relevant procedures in the code

Modules (file)	Procedures
tracer_processes (t_proc.f90)	add_tracer_point_source_to_river
	add_tracer_point_source_to_lake
	soil_tracer_processes
	tracer_processes_in_river
	tracer_processes_in_lake
	decay_of_tracer
	decay_of_tracer_sorbedphase
	sorption_of_tracer
	sedimentation_resuspension_of_tracer
	T1_crop_sources
	T1_incorporation
	release_from_source

## Water temperature (T2)

Water temperature can be simulated by HYPE with the substance/tracer called T2. Water temperature is suitable to be simulated in a catchment model, because it relates to the hydrology and the landscape. The substance follows the water paths in HYPE and is calculated in degree Celsius.

### Sources

There are different ways to introduce heat into the water of HYPE.

Initial value of water temperature of HYPE's water bodies can be set by model parameters. The general parameter *iniT2* is used for some water bodies, i.e. soil, rivers and lakes. Aquifer temperature is given in [AquiferData.txt](#) (*temp*).

Snow and glacier melt water is assumed to hold zero degrees.

### Precipitation

Temperature of precipitation on land is set to air temperature (but limited to zero).

For water surfaces the temperature setting is more complex. Temperature of rain on water is set to air temperature (but limited to zero). Snowfall are assumed to have the average temperature of the water (uppertemp for lakes, mean temperature for rivers) for ice conditions, i.e. not to influence the temperature of the water. For snowfall on open water the temperature is calculated as dependent on sensible heat content and latent heat of freezing, taken into account different specific heat capacity of ice and water. Temperature of precipitation is a weighed average of these three components.

### Point source

Point sources influence the water temperature T2. Point sources may have a constant temperature set from indata (*ps\_t2*) or otherwise the water is assumed to be zero degrees. Point sources of tracer

T2 can be added to the local or main river, or internal or outlet lake.

## **Local diffuse source**

The water of local diffuse source is not allowed to influence the water temperature of the receiving water body (soil or local river).

## **Soil processes**

The primary process for changing water temperature is exchange of heat over a water surface. Apart from that, surface water temperature is determined by mixing water of different temperature. Exceptions have been made for some processes not to influence the water temperature. This because their effect on water temperature has not yet been included in full.

## **Soil temperature**

In addition to the “concentration” of T2 in soil water, the (solid) soil layer has their own temperature state variables. Temperature of soil, i.e. solid material, is calculated separately from the T2 temperature of soil water. The temperature of a soil layer depends on the air temperature and the deep soil layer temperature to different degree. Deep soil layer temperature is calculated separately. See Chapter [Soil temperature and snow depth](#).

Temperature T2 of soil water is not yet developed and is simply set to the temperature of soil. Still, snow melt are assigned the temperature T2 zero. Evaporation of soil water is not affected by T2 or affecting soil water T2.

## **Runoff**

Soil runoff temperature is set from soil layer temperatures. Tile drainage temperature is set to the temperature of the soil layer it originates from. Surface runoff gets the temperature from soil layer one (saturated overland flow) or rainfall, snowmelt and glaciermelt (excess infiltration).

## **Lake and river ice**

The river ice routines are performing best if the special classes for local and main river is set. The routines then use the forcing data and snow parameters defined for these classes. If the river classes is not set, forcing and snow parameters for lakes are used as a replacement. If no lake classes are provided, forcing and snow parameters are taken from slc class one, whatever that is.

## **Floating snow covered ice sheet model**

Lake and river ice is simulated with a simple thermodynamic ice model driven by air temperature and precipitation. It is to a large extent adopted from the models presented by Leppäranta (1983, 1993),

in which lake and river ice is represented by a floating snow covered ice sheet characterized by three layers of frozen water: black ice, snow ice, and snow (Figure 2, left). If the top of the ice layer is submerged under water, the depth of a slush layer (wet snow) is also defined (Figure 2, right).

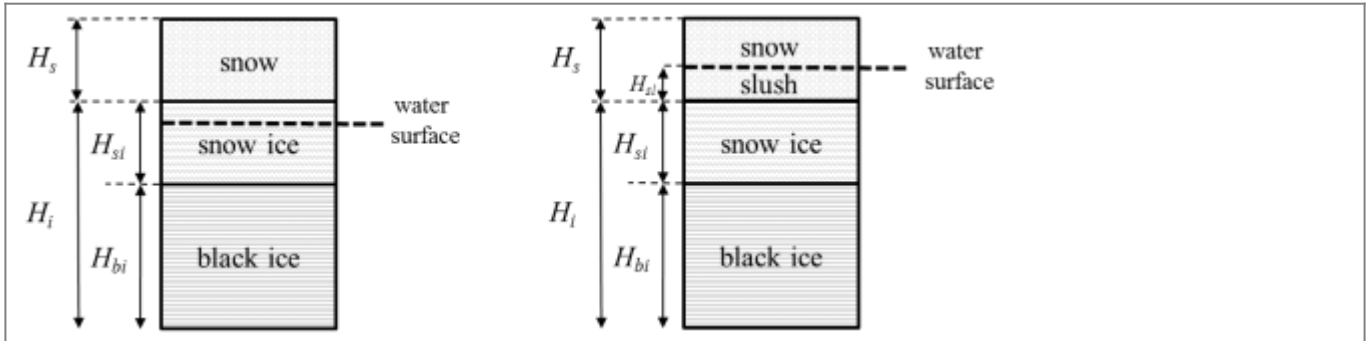


Figure 2: A floating snow covered ice sheet with (left) three layer of frozen water black ice, snow ice and snow and (right) with a slush (wet snow) layer formed when the top of the ice is sub-merged under water due to the load of the snow pack.

### Influence of the ice model on lake and river evaporation and water temperatures

It should be noted that in the current version of the HYPE model, the ice mass balance is independent from the mass balance of the lakes and rivers. No water is removed from lake or river water to form the ice cover and no water is added to the lake and river when the ice is melting. This simplification is reasonable only for rivers and lakes large enough to never freeze to the bottom. The ice model influences lake and river water only through its' impact on water temperatures and evaporation:

- evaporation from lakes and rivers is set to 0 in the presence of ice
- heating and cooling of lakes and rivers through surface heat exchange with the air is set to 0 in the presence of ice
- lake and river water temperature is adjusted to take into account the inflow of 0°C melt water during ice melt

Water temperature is also adjusted to take into account the heat needed to melt snowfall on ice-free lakes and rivers.

### Black ice growth

Black ice growth ( $\frac{dH_i}{dt}$  (cm/s)) is derived from the modified Stefan's equations as described by Leppäranta (1983) reduced by heat flow from water.

$$\frac{dH_i}{dt} = \frac{k_i}{\rho_i \times L_f} \times \frac{(T_f - T_a)}{\left( H_i + H_s \times \frac{k_i}{k_s} + \frac{k_i}{k_a} \right)} - \frac{\frac{qh \times 100}{\rho_i}}{T_f}$$

The first part is constant in HYPE.  $k_i$  is thermal conductivity of ice (0.022 J/°C/cm/s),  $\rho_i$  is density of ice (0.917 g/cm<sup>3</sup>), and  $L_f$  is latent heat of freezing (334 J/g).

$T_f$  is freezing temperature of lake or river water (°C), a parameter.  $T_a$  is air temperature (°C).  $H_i$  is ice depth (cm) and  $H_s$  is snow depth on the ice (cm).  $k_a$  is heat exchange coefficient with atmosphere (J/°C/cm²/s).  $k_s$  is thermal conductivity of snow (J/°C/cm/s). It is calculated from thermal conductivity of ice and snow and ice density, and a parameter  $k_{exp}$ :

$$k_s = k_i \times \left( \frac{\rho_s}{\rho_i} \right)^{k_{exp}}$$

where  $\rho_s$  is density of snow (g/cm³). Snow density changes over time depending on a change parameter (*sndens*).

$qh$  is heat flow from water during the time step (MJ/m²/day). For rivers it is calculated depending on river velocity (*vel*) and depth (*Hw*), but for lake it is constant, (parameter *liceqhw*, W/m²). The river equation is

$$qh = (T_w - T_f) \times \frac{Cwi \times vel^{0.8}}{Hw^{0.2}} \times unitf$$

but limited by minimum and maximum parameters (*riceqhm*, *riceqhm*).  $T_w$  is water temperature,  $Cwi$  is the heat exchange coefficient, a parameter. A unit transformation (*unitf*) is made from W/m² to MJ/m²/day.

The heat released when the black ice is formed at the bottom of the ice layer ( $\rho_i \times L_f \times \frac{dH_i}{dt}$ ) is conducted through the ice and the overlaying snow layer to the atmosphere. This heat flow is driven by the temperature gradient between the air above the ice or snow surface and the freezing temperature of the water assumed at the bottom of the ice layer, and is governed by the depths and thermal conductivities in the ice and snow layers, and the heat exchange coefficient in the air. It should also be noted that a term representing the heat flow from the water has been neglected compared to Leppäranta (1983).

The black ice growth equation is only used if the ice surface is above the water surface (in other words, if there is no slush layer on top of the ice to be frozen first) and if air temperature is below the freezing temperature  $T_f$ .

## Snow ice growth

If the mass of the snow layer exceeds the floating capacity of the ice, the bottom of the snow layer will be submerged in lake water, and a slush layer ( $H_{sl}$ ) will form with a depth according to:

$$H_{sl} = \frac{H_s \times \rho_s - H_i (1 - \rho_i)}{\rho_s / \rho_i}$$

As pointed out by Leppäranta (1993), the freezing of slush into snow ice releases less heat compared to black ice growth since the slush already consists of frozen snow particles, and this heat only need to be conducted through the snow pack. The growth of snow ice can thus be calculated by the



following equation from Leppäranta (1993):

$$\frac{dH_{bi}}{dt} = \frac{k_s}{\left( \rho_i \times L_f \left( 1 - \frac{\rho_s}{\rho_i} \right) \right)} \times \frac{(T_f - T_a)}{\left( H_s + \frac{k_i}{k_a} \right)}$$

where all variables are defined above for black ice growth.

### Ice melt and final ice break-up

Snow and ice melt is calculated with a simple temperature index model when air temperature is above 0°C. The snow on the ice has to be completely melted before any ice melt is calculated:

$$\frac{dH_i}{dt} = -p_{mt} \times T_a$$

Ice can in addition be melted internally by radiation.

$$\frac{dH_i}{dt} = -\frac{p_{mr} \times 100}{L_f \times \rho_i}$$

Final ice break-up is defined as the time step when either the ice thickness is zero or the ice porosity is less than a threshold. The thresholds are given by general parameters; different for river and lakes (*ricebupo*, *licebupo*).

### Initiation of ice growth

Ice growth is initiated when the solution of the surface heat balance for the open water results in a water temperature below the freezing point. The fraction of the water surface with new ice formation is then estimated by reducing the open water surface area until the surface heat balance solution results in a water temperature equal to the freezing point. The transition from fully ice-free to fully ice-covered conditions is thus smoothed by the gradual increase in fractional ice cover. For deep lakes, the temperature of an upper layer (EPI, Figure 3 below) is used for the onset of ice growth, whereas for shallow lakes and rivers, the mean water temperature is used.

### Surface water processes

River model option (add first)

Temperature T2 of a river is not affected by point sources which not have T2 (ps\_t2) set in PointSourceData. In this case the receiving main river flow will keep the temperature it had before

adding of the point sources. In earlier HYPE versions, temperature T2 of a river was not affected by irrigation withdrawal, abstractions, rural households or point source additions, constructed wetlands, or water returning from aquifers. To get this old function of HYPE, there is a modeloption to use (modeloption rivert2model 1).

## Lake basic assumptions

To simulate temperature as the substance T2 the lake is divided in two lake parts, lake temperature in a hypothetical epilimnion and hypolimnion (*uppertemp*, *lowertemp*) is handled in the model. The average temperature of the lake is also a state variable. The size of volumes related to the upper and lower temperatures are determined by the thermocline which is estimated for each outlet lake.

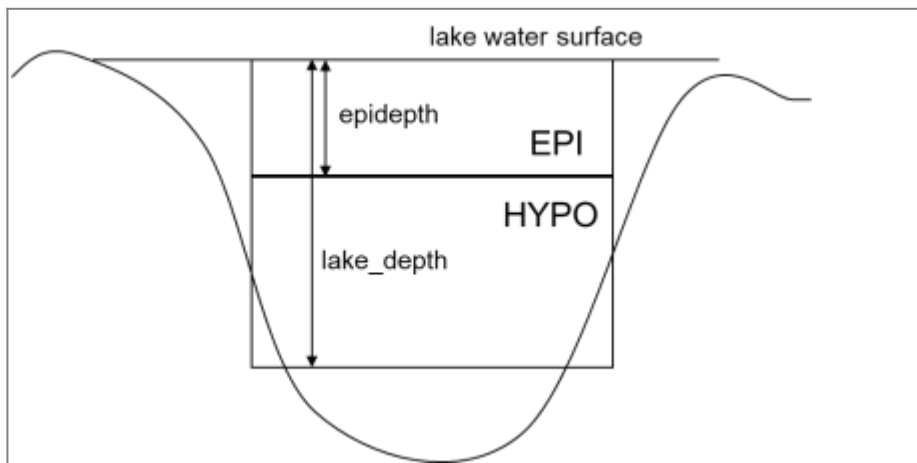


Figure 3: Division of a lake: epilimnion (EPI) and hypolimnion (HYPO).

Thermocline depth is estimated from lake area (Hanna, 1990). This average depth is adjusted for current changes by adding precipitation (*prec*) and inflow (*qin*) to the lake, and remove evaporation (*evap*).

$$epidepth = 6.95 \times lakearea^{0.185} + prec - evap + qin$$

where *lakearea* is (km<sup>2</sup>) and *epidepth* is in m below lake surface. Precipitation, evaporation and inflow is also in meter.

The adjusted depth (*epidepth*) is then used as the division of the lake volume into upper and lower if the lake water depth is larger than the *epidepth*. Otherwise the lake is treated as one single mixed volume in regard to T2 calculations.

## Evaporation

Lakes and rivers with given class area may have evaporation. The actual evaporating area is reduced if it is ice covered. Evaporation is not changing the T2 temperature of the river or lake. The latent heat is assumed to be taken care of by the water - atmosphere T2 exchange routine. River evaporation is removed proportionally from the water stores making up the river volume.

Evaporation from flooded floodplain of rivers or lakes also keep the temperature of the flooded water unchanged.

## Water - atmosphere T2 exchange

River and lake water atmosphere exchange is calculated during ice free conditions. Water surface heat balance is calculated based on a model by Piccolroaz et al (2013), with modifications for fractional ice cover, and calculation of fractional freezup area. This equation depends on four general parameters; *tcf*-temperature difference coefficient, *scf* - solar radiation coefficient, *ccf* - constant coefficient and *lcf* - linear coefficient. The parameters comes in one set for rivers (*tcfriver*, *scfriver*, *ccfriver*, *lcfriver*) and one set for lakes (*tcflake*, *scflake*, *ccflake*, *lcflake*).

A simpler T2 exchange model based only on the temperature difference between air and water is also available. It has two general heat transfer parameters (*t2trlake*, *t2trriver*, unit J/m2/s/degree).

For rivers with no specified river area as a class, a river area is calculated for use in the water - atmosphere T2 exchange and ice processes. The river area is calculated as described in Section [Nitrogen and phosphorus processes in rivers and lakes](#). In addition the river area may be reduced at low volume/flow. This is controlled with parameters (see [River and Lakes - Evaporation](#)).

## T2 stratification

At spring and autumn circulation, T2 temperature is averaged out between upper and lowertemp. During periods with stratification a heat transfer between the uppertemp and lowertemp is assumed. This heat transfer depend on the general parameter *upper2deep* (J/m2/s/degree).

## Lake outflow

Lake outflow T2 temperature may be determined by the average T2 of the lake. To use this method set parameter *t2mix*. For ilakes the *t2mix* method is always used.

For outlet lakes another method for determining lake outflow temperature is the default method. This method assigns uppertemp to outflow water if the lake is stratified and the thermocline is deeper than the lake threshold (*thres*). For other conditions a mixture of upper and lowertemp is used to set the outflow temperature (*t2out*).

$$t2out = fraction \times uppertemp + (1 - fraction) \times lowertemp$$

$$fraction = \frac{epidepth}{(lakewst - thres)}$$

where *lakewst* is the current depth of the lake.

## References

- Hanna 1990. Evaluation of models predicting mixing depth, Can. J. Fish. Aquat. Sci., 47:940-947.
- Leppäranta, M. 1983. A Growth Model for Black Ice, Snow Ice and Snow Thickness in Subarctic Basins. Nordic Hydrology, 14(2):59-70.

Leppäranta, M. 1993. A review of analytical models of sea-ice growth, Atmosphere-Ocean, 31(1):123-138.

Piccolroaz, S., M. Toffolon, and B. Majone 2013. A simple lumped model to convert air temperature into surface water temperature in lakes, Hydrol. Earth Syst. Sci., 17:3323-3338, doi:10.5194/hess-17-3323-2013.

## Links to file reference

Section	Symbol	Parameter/Data	File
Sources		$iniT1$	<a href="#">par.txt</a>
		$temp$	<a href="#">AquiferData.txt</a>
		$ps\_t2$	<a href="#">PointSourceData.txt</a>
Lake and river ice	$T_f$	$ricetf, licetf$	<a href="#">par.txt</a>
	$k_a$	$kika = \frac{k_i}{k_a}$ calculated from $ricekika, licekika,$	
	$k_{exp}$	$ricekexp, licekexp$	
	$sndens$	$ricesndens, licesndens$	
	$p_{mt}$	$ricetmelt, licetmelt$	
	$p_{mr}$	$ricermelt, licermelt$	
	$meff$	$ricewme, licewme$	
	$vel, Cwi$	$rivvel, ricecwi$	
		$riceqhm, riceqhm, liceqhw, ricebupo, licebupo$	
Surface water processes	$tcf$	$tcfriver, tcflake$	<a href="#">par.txt</a>
	$scf$	$scfriver, scflake$	
	$ccf$	$ccfriver, ccflake$	
	$lcf$	$lcfriver, lcflake$	
		$t2trlake, t2trriver, upper2deep, t2mix$	
	$thres$	$lake\_depth$	<a href="#">GeoData.txt</a> or <a href="#">LakeData.txt</a> or <a href="#">DamData.txt</a>

## Links to relevant procedures in the code

Modules (file)	Procedures	Section
<a href="#">tracer_processes (t_proc.f90)</a>	<code>add_tracer_point_source_to_river</code>	point source
	<code>add_tracer_point_source_to_lake</code>	
<a href="#">npc_surfacewater_processes (npc_sw_proc.f90)</a>	<code>add_point_sources_to_main_river</code>	

Modules (file)	Procedures	Section
<a href="#">npc_soil_processes</a> ( <a href="#">npc_soil_proc.f90</a> )	set_class_precipitation_concentration_and_load	precipitation
<a href="#">surfacewater_processes</a> ( <a href="#">sw_proc.f90</a> )	add_T2_concentration_in_precipitation_on_water	evaporation
	calculate_river_evaporation	
	calculate_lake_epilimnion_depth	lake basic assumptions
	calculate_T2_transfer_upper2lower	T2 stratification
	T2_processes_in_river	water - atmosphere T2 exchange
	T2_processes_in_lake	
	calculate_watersurface_heatbalance	Lake and river ice
	ice_processes_in_river	
	ice_processes_in_lakes	
	calculate_icedepth	
	calculate_snow_on_ice	
	riverice_riverwater_interaction	
	calculate_lakeice_lakewater_interaction	
<a href="#">soil_processes</a> ( <a href="#">soil_proc.f90</a> )	calculate_snowmelt	
	calculate_snowdepth	
	calculate_snow	sources
<a href="#">regional_groundwater</a> ( <a href="#">regional_groundwater.f90</a> )	initiate_aquifer_state	
<a href="#">soilmodel_default</a> ( <a href="#">soilmodel0.f90</a> )	soilmodel_0	soil temperature and runoff
<a href="#">glacier_soilmodel</a> ( <a href="#">glacier_soilmodel.f90</a> )	soilmodel_3	
<a href="#">soilmodel_4</a> ( <a href="#">soilmodel4.f90</a> )	soilmodel_4	

# Sediment

## Introduction

Sediments are simulated as suspended sediments (SS) and nitrogen in algae (AE). The sum of the two **substances** is an additional output (total suspended sediments, TS). Suspended sediments are not simulated in the soil of HYPE. Suspended sediments are first introduced into the runoff of the soil by soil erosion.

The main **states** are concentration of SS and AE in all water stores of HYPE, i.e. soil, river, lakes, but also snow, aquifers (in the store and on the move there or away), irrigation water, floodplain water, and water transfer (that is delayed one time step) could hold suspended sediments and algae. In the current model though, concentration of SS and AE is only positive in river and lakes, and flows originating from them. In addition two pools of settled sediment are simulated; a pool of delayed sediment in runoff, and a pool of (temporarily) settled sediment in river. Note: No store of “settled sediment” of the soil is simulated, i.e. HYPE has an unlimited source of soil for erosion.

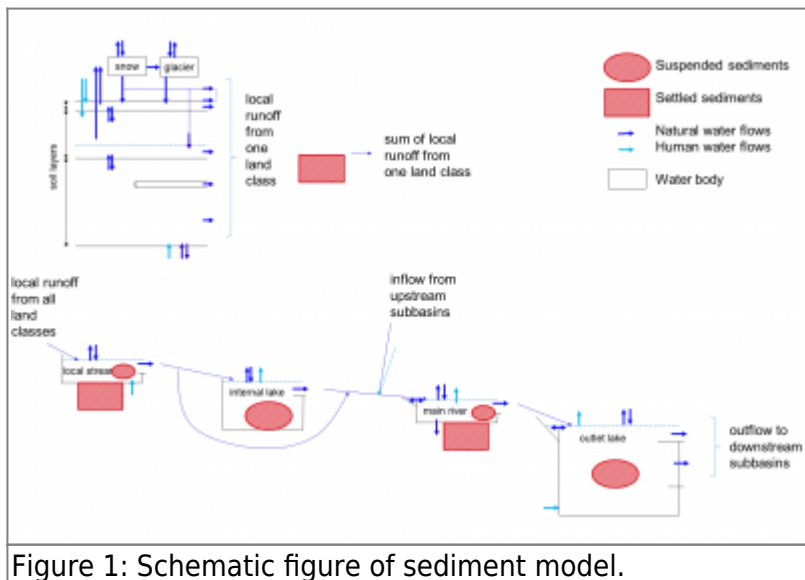


Figure 1: Schematic figure of sediment model.

## Sources of sediment load

### Soil erosion

The main source of sediment in HYPE is from soil erosion. Soil erosion is modelled in several steps. **First** particles are mobilized from the soil by rain or surface runoff. The result is here called “mobilized sediments”. The mobilized sediments are assumed suspended in an “eroding flow”, which is surface runoff plus macropore flow. **Secondly**, if there is surface runoff, the mobilized sediments are transported away from the field with the surface runoff and macropore flow (“eroded sediment”). A filtering can be applied to reduce the amount of eroded sediment. If there is no surface runoff, there will be no transport of eroded sediments. For SS, the eroded sediments are taken from an infinite pool. For PP, the eroded particulate phosphorus is taken from the soil pools. **Thirdly**, the eroded sediments in macropore flow and surface runoff (and tile runoff though that is generally zero) are delayed in a temporary pool. The release of suspended sediments from this pool is determined by the

total runoff from the class, and is following the total runoff off the land.

### Step 1: Mobilization of particles from soil

HYPE has two alternative models for mobilization; soil erosion model 1 and soil erosion model 2.

*Soil erosion model 1 (MMF-based model)*: The first model is based on Morgan-Morgan-Finney erosion model (Morgan et al., 1984) and calculates particles mobilized by rainfall energy and surface runoff. The kinetic energy in rainfall is calculated as a function of rainfall (*rain*, mm/ts) and day of the year (*dayno*). If the precipitation falls as snow, or if it falls on snow-covered ground or if it is smaller than 5 mm/day no mobilization occurs in the model. Some of the raindrop's energy can be absorbed by vegetation. Crop cover is defined as the portion of land that is sheltered from raindrops; for a description of how this is calculated, see Chapter [Crop cover and ground cover](#). The factor  $common_{cropcover}$  is the sheltering effect that the main and secondary crops give together. It varies over the year due to crop growth and management. The mobilization by rain (*MobilisedRain*, g/m2/ts) is also influenced by soil erodibility (soil dependent parameter *soilerod* (g/J)).

$$Rainfall_{energy} = rain \times \left( 8.95 + 8.44 \times \log_{10} \left( rain \times 2 \times \left( 0.257 + 0.09 \times \sin \left( 2\pi \times (dayno - 70) / 365 \right) \right) \right) \right)$$

$$MobilisedRain = Rainfall_{energy} \times (1 - common_{cropcover}) \times soilerod$$

When surface runoff occurs, soil particles are eroded and carried away as the soil surface is exposed to shear forces. The mobilization (*MobilisedSR*, g/m2/day) is calculated from the surface runoff (*sflow*, mm/day), subbasin average slope, a parameter for soil cohesion (*soilcoh* (kPa) soil type dependent), and a general parameter (*sroexp*). This type of erosion can be mitigated by protective vegetation or vegetation residues that are in contact with the ground. The calculation of this reducing factor (*groundcover*) is described in Chapter [Crop cover and ground cover](#). The factor  $common_{groundcover}$  is the combined effect of the main and secondary crops.

$$MobilisedSR = \frac{(sflow \times 365)^{sroexp} \times (1 - common_{groundcover}) \times \frac{1}{0.5 \times soilcoh} \times \sin \left( \frac{slope}{100} \right)}{365}$$

All mobilized particles are not staying mobilized, because sometimes the transport capacity of the particle-bearing water (*eflow*) will not suffice for the task. If this is the case, a *transportfactor* reduces the particle amount mobilized:

$$transportfactor = MIN \left( 1.0, (eflow / 4)^{1.3} \right)$$

Finally mobilized sediment (*mobilSed*, kg/km2/day) is calculated as the sum of rain and surface runoff caused mobilization as:

$$mobilSed = 1000 \times (MobilisedRain + MobilisedSR) \times transport\ factor$$

*Soil erosion model 2 (HBV-SED based model)*: The second model is based on HBV-SED model (Lidén, 1999; Lidén et al., 2001) and calculates particles mobilized by rain (*rain*). When the ground is frozen or there is snow on the ground, the no particles are mobilized. The mobilization also depends on soil and land characteristics in the form of model parameters and data on subbasin characteristics.

$$mobilSed = 1000 \times \left( \frac{slope}{5} \right)^{erodslope} \times erodluse \times erodsoil \times \frac{EI}{erodindex} \times rain^{erodexp}$$

The parameters *erodslope*, *erodexp* and *erodindex* are general and thus same for the whole model domain. The parameters *erodluse* and *erodsoil* are land-use and soil type dependent. Subbasin input are: *slope*, the subbasins' average slope, and an erosion index, *EI*. The many parameters give the possibility to simulate erosion as dependent on slope, soil type, land use or subbasin. In addition an adjustment can be made with a monthly model parameter *erodmon* to help match seasonality of erosion.

$$modileSed = mobileSed \times erodmon(month)$$

where *month* is the current month.

## Step 2: Transport of eroded sediments off the field

If at one time step there is no surface runoff, there will be no transport of suspended sediment. The mobilised sediment is returned to the soil. Still, calculation continues with step 3, and earlier delayed eroded sediment may reach the local stream. If there is surface runoff, a fraction of the mobilised sediments is assumed to go with surface runoff (*sflow*). If there is surface runoff and tile runoff and macropore flow, a fraction of the mobilized sediments is assumed to travel with the macropore flow (*mflow*). The respective fractions depend on the respective size of the two flows, but are reduced by filtering of the particles. The filtering depends on landuse, the proximity of agricultural land to water, presence of buffer strips etc. The filtering effect on suspended sediments in surface runoff (*srfilt*) is parameterized with land use parameters and subbasin input, while the filtering effect on suspended sediments of macropore flow is parameterized with a soil type parameter (*macrofilt*).

$$srfilt = otherfilt + alfa \times \left( 1 + bufferpart \times (bufferfilt - 1) \right) + innerfilt \times (1 - alfa)$$

$$erodedSed = \frac{(srfilt \times sflow + macrofilt \times mflow) \times mobilSed}{eflow}$$

$$eflow = sflow + mflow$$

The total amount of eroded sediment (*erodedSed*, kg/km<sup>2</sup>/day) is the sum of contributions from surface flow and macropore flow. For SS, the eroded sediments are taken from an infinite pool. For PP, the eroded particulate phosphorus is calculated from the eroded sediment (see [Soil erosion](#)) and then taken from the solid soil pools (*humusP* and *partP*). This difference exists because HYPE does not simulate any pool of "soil sediments".



### Step 3: Suspended sediment reaching the local stream

The concentration of suspended sediments of the runoff leaving the soil is calculated based on total runoff (*runoff*, mm/ts) and a pool of sediment particles (*relpool*). The pool collects the suspended sediments in the respective runoff pathways, i.e. primarily the eroded sediment from surface runoff and macropore flow calculated in the previous step. The pool constitutes a temporary delay. The release of sediment (*release* (kg/km<sup>2</sup>)) from the pool temporary is calculated with two general parameters (*pprelmax* and *pprelexp*):

$$release = MIN \left( relpool, relpool \times \left( runoff / pprelmax \right)^{pprelexp} \right)$$

The suspended sediments released are following the total runoff off the land. The release divided by the total runoff gives a suspended sediment concentration, which is used for all runoff pathways. This concentration is set to all runoff from the class, i.e. the sediments are transported by all flow paths off the land.

The sediment may be removed from the temporary pool between the land surface and the local stream if addition to the pool is made for the time step through erosion by surface runoff or through tile drainage. A 1st order decay, with parameter *eroddecay*, is used so that sediment doesn't remain detached forever.

### Rural household outlets and point sources

Suspended sediments may be added to the model as diffuse rural households outlets (private sewers) or point sources. Similar to other substances in HYPE, they are added as flow with concentration of the total suspended sediments and with a fraction determining the suspended part, while nitrogen in algae make up the rest ([GeoData.txt](#) [PointSourceData.txt](#)). To add only suspended sediments set the fraction to 1. The diffuse source is divided into two parts, where the division is determined by a general parameter (*locsoil*). One part is added directly to the local river. The other part is added to the water of the bottom soil layers of each class in the subbasin. The distribution between classes is done proportional to the area of each class. Point sources are added in the main river. Point sources are described in the Chapter on Water management ([point sources](#)) and rural household diffuse source in the Chapter on Nitrogen and phosphorus in land routines ([rural sources](#)).

### Links to file reference

Section	Symbol	Parameter/Data	File
Soil erosion		<i>soilerod</i> , <i>soilcoh</i> , <i>sreroexp</i> , <i>erodslope</i> , <i>erodluse</i> , <i>erodsoil</i> , <i>erodindex</i> , <i>erodexp</i> , <i>erodmon</i> , <i>bufferfilt</i> , <i>innerfilt</i> , <i>otherfilt</i> , <i>macrofilt</i> , <i>pprelmax</i> , <i>pprelexp</i> , <i>eroddecay</i>	<a href="#">par.txt</a>
	<i>slope</i>	<i>slope_mean</i>	<a href="#">GeoData.txt</a>
	<i>El</i>	<i>eroindex</i>	
	<i>alfa</i>	<i>close_w</i>	
	<i>bufferpart</i>	<i>buffer</i>	

Section	Symbol	Parameter/Data	File
Rural household outlets and point sources		<i>subid, ps_vol, ps_type, ps_tsconc, ps_ssfrac, fromdate, todate</i>	<a href="#">PointSourceData.txt</a>
		<i>loc_ts, loc_vol, loc_ss</i>	<a href="#">GeoData.txt</a>
		<i>locsoil</i>	<a href="#">par.txt</a>

## Link to relevant processes in the code

Modules (file)	Procedures	Sections
<a href="#">npc_soil_processes (npc_soil_proc.f90)</a>	particle_processes_for_runoff	Soil erosion
	calculate_hbvsed_erosion	
	calculate_erosion	
<a href="#">datamodule (data.f90)</a>	load_pointsourcedata	Rural household outlets and point sources
	read_pointsourcedata	
	get_current_pointsources	
<a href="#">npc_surfacewater_processes (npc_sw_proc.f90)</a>	add_point_sources_to_main_river	
	add_diffuse_source_to_local_river	
<a href="#">npc_soil_processes (npc_soil_proc.f90)</a>	local_diffuse_source	

## Soils

Suspended sediments are not simulated in the soil calculations of HYPE. Suspended sediments are first introduced into the runoff of the soil.

## Sediment in rivers and lakes

If the siltation model option is set, the sedimentation of lakes is added to a sediment pool. The sedimentation of rivers is always added to a sediment pool.

## Primary production and mineralization

Primary production in lakes and rivers is affecting sediment simulation by being a source of algae organic nitrogen (AE). Mineralisation, as a sink of AE, is the reverse process. Nitrogen in algae is assumed to grow and decline with the same function as [production and mineralisation of organic nitrogen \(ON\)](#). The production and mineralisation processes are modelled together and only one is active at the time. If nitrogen is simulated by HYPE it uses the actual estimated production/mineralization for AE, otherwise the potential production/mineralisation (*minprodNpot*) is used for AE. The actual production is limited by the availability of inorganic nutrients. If sediments are simulated without simultaneously simulating nitrogen, the mineralisation of algae is limited to available amount, but the production is unlimited.

The potential production/mineralisation is depending on temperature ( $T_w$ ,  $T10$ ,  $T20$ ), long-term total phosphorus concentration ( $TP$ ), surface area of water body ( $area$ ) and a parameter ( $wprodn$ ).

$$minprodNpot = wprodn \times TP_{fcn} \times tmp_{fcn} \times area$$

$$TPfcn = \frac{TP - limsedPP}{(TP - limsedPP) + hsatTP}$$

$$tmpfcn = tmpfcn1 \times tmpfcn2$$

$$tmpfcn1 = \frac{T_w}{20.}$$

$$tmpfcn2 = \frac{(T_{10} - T_{20})}{5.}$$

Temperatures T10 and T20 are calculated as the average water temperature ( $T_w$ ) of 10 and 20 days. They determine if production or mineralization dominates at the current time. The water temperature is calculated through weighting the air temperature (T) and yesterday's water temperature (see [basic assumptions of rivers and lakes](#)).

The phosphorus function ( $TPfcn$ ) is dependent on a general half saturation parameter  $hsatTP$ , and a limitation parameter  $limsedPP$ . If phosphorus is simulated by HYPE the long-term average concentration ( $TP$ ) is calculated from that, otherwise a lakeregion parameter is used ( $tpmean$ ).

## Sedimentation of lakes

Sedimentation in lakes is a sink for SS and AE and works the same way as for [nutrients](#).

Sedimentation ( $sed$ , kg/day) is calculated as a function of concentration in lake water ( $conc$ ) and lake area ( $area$ ). The settling velocity parameter ( $sedss$ ,  $sedae$ , m/timestep) is general. The concentration used in the equation may be limited ( $lim$ ) by general parameter ( $limsedSS$ ) for SS, but not for AE ( $lim=0$ ).

$$sed = par_{sed} \times (conc - lim) \times area$$

## Sedimentation/Resuspension in rivers

In the river we simulate both sedimentation and resuspension. No particles are removed from the simulation by sedimentation. For the original model, and a variant, they are redistributed over time and can come back through resuspension based on flow. For the other alternative model it is possible to resuspend more particles than is previously settled. This double process of sedimentation and resuspension is used for suspended sediments (SS). Particles in the sediments is collected in a pool ( $sedimentpool$ , kg).

For the **original model** the sediment pool will increase as particles from the water volume ( $waterpool$ , kg) settle at low water flows, and vice versa. At high water flow the sedimentation is lower and the resuspension higher, and more particles returns to the water. The net effect of these combined processes are given by the variable  $sedresp$ . It is governed by the current flow ( $flow$ ) in relation to the bankful flow ( $qbank$ ) and a general parameter  $sedexp$ . The net effect is determined by the sign of the variable ( $-1 < sedrespfactor < 1$ ), and the variable size give the fraction of the pool that is transferred per day. The process will give either net sedimentation ( $settl$ , kg/day) or resuspension ( $resusp$ , kg/day).

$$sedresp\ factor = \max \left( -1, \min \left( 1, \frac{q_{bank} - flow^{sedexp}}{q_{bank}} - \frac{flow^{sedexp}}{q_{bank}} \right) \right)$$

$$settl = \begin{cases} sedresp\ factor \times waterpool & sedresp > 0 \\ 0 & sedresp < 0 \end{cases}$$

$$resusp = \begin{cases} -sedresp\ factor \times sedimentpool & sedresp < 0 \\ 0 & sedresp > 0 \end{cases}$$

where *flow* is the current river flow ( $m^3/s$ ) and *qbank* is the flow when river is filled to the brim. The latter flow is calculated as the second largest simulated flow in the last year. It is adjusted with a correction factor of 0.7 before use in the *sedresp* equation. The first alternative model is the same as the original model but the correction factor can be calibrated. It is given by the general model parameter *qbank*.

The **second alternative model** is based on a simplified Bagnold equation (Neitsch et al., 2011). Suspended sediment concentration is compared to a calculated maximum amount of sediment that can be transported in the river to determine if sedimentation or resuspension occurs. The maximum transport depends on the peak velocity across the river ( $v_{peak}$ ). This is estimated from flow in the river (*Q*) using the hydraulic geometry relationship ([river velocity](#)) which gives the average river velocity and a peak adjustment factor (*vpeak*).

$$v_{peak} = v_{peak} \times k \times Q^m$$

where *vpeak*, *k* and *m* are general parameters. The maximum amount of sediment that can be transported in the reach ( $c_{max\ SS}$ , mg/L) is calculated using the peak velocity and two general model parameters (*spcon* and *spexp*).

$$c_{max\ SS} = spcon \times v_{peak}^{spexp} \times 1000000.$$

If the simulated HYPE suspended sediment concentration (*conc\_SS*) is larger than the maximum transport concentration, then sediment is deposited in the temporary pool. Resuspension occur in the opposite case. The net transport (*sedresp*=*settl*+*resusp*):

$$sedresp = \begin{cases} (conc_{SS} - c_{max\ SS}) \times riverbox & conc_{SS} > c_{max\ SS} \\ (conc_{SS} - c_{max\ SS}) \times riverbox \times suspch & conc_{SS} < c_{max\ SS} \end{cases}$$

where *riverbox* is volume of water in attenuation box of the river and *conc\_SS* its concentration of suspended sediments. The effect of channel erodibility and channel vegetation cover is captured by a river channel erodability/vegetation cover modification factor, *suspch*.

As default the amount of sediment that can be resuspended is limited by the amount of sediment that has been deposited before (i.e. the temporary pool (*sedimentpool*)). This can be changed by using a

parameter *addsed* which allows users to apply a limit to the amount of sediment that can be resuspended in HYPE in excess of the amount of sediment in the temporary pool. If *addsed*=0, then the amount of sediment that can be resuspended is limited to the amount of sediment in the temporary pool. If *addsed*=1, then there is no limit on the amount of sediment that can be resuspended (the full amount from equation above will be resuspended). For values in between the additional resuspension increases linearly:

$$resuspension = sedimentpool + (-sedresp - sedimentpool) \times addsed$$

## Siltation

It is optional to simulate the lake sediment in HYPE. If the siltation model option is not used the settled sediment (Sedimentation of lakes above) is lost from the system. If the siltation option is used then the settling sediment add to a sediment pool. This is true for all substances that sediment in lakes, but only the substances SS and AE are used further. The settled substances may be compressed over time and their density thus affected. They may fill up the lake and affect its water depth and outflow. They may be removed or flushed at regular or irregular intervals.

The density of sediments may be constant in time and space. It is the given by *ldseddens*, a general parameter, or determined by the subbasin's soil particle fractions ( $d_{ini}$ ). The sediment density may also increase with time through compactation ( $d_{comp}$ ). This is modelled to depend on the sediment age.

$$d_{ini} = c_{sand} \times p_{sand} + c_{silt} \times p_{silt} + c_{clay} \times p_{clay}$$

where soil fraction ( $p_{sand}, p_{silt}, p_{clay}$ ) are given per subbasin and coefficients ( $c_{sand}, c_{silt}, c_{clay}$ ) depend on reservoir operation mode.

$$d_{comp} = d_{ini} + 0.434 \times K_{ini} \times \left( \frac{T}{T-1} \times \ln(T) - 1 \right)$$

$$K_{ini} = k_{sand} \times p_{sand} + k_{silt} \times p_{silt} + k_{clay} \times p_{clay}$$

where coefficients  $k_{sand}, k_{silt}, k_{clay}$  depend on reservoir operation mode, and  $T$  is the sediment age in years. For ages less than one year the initial density ( $d_{ini}$ ) is used.

There are four reservoir operation modes (*resop\_mode*) implemented in HYPE (Strand and Pemberton, 1982);

1. sediment always submerged or nearly submerged,
2. normally moderate to considerable reservoir drawdown,
3. reservoir normally empty,
4. riverbed sediments.

Table of coefficients for sediment density for the different reservoir operation modes (*resop\_mode*).

Coefficient	resop_mode 1	resop_mode 2	resop_mode 3	resop_mode 4
$c_{sand}$	1550	1550	1550	1550
$c_{silt}$	1120	1140	1150	1170
$c_{clay}$	416	561	641	961
$k_{sand}$	0	0	0	0
$k_{silt}$	91	29	0	0
$k_{clay}$	256	135	0	0

## Flushing

Flushing is used to remove sediment that has accumulated in a reservoir over time. It is a sediment management method that can be regulated in different ways. The flushing modes are depending on capacity, sediment age or day of year. In each case a starting day is found and a rate of flushing calculated. Additionally an ending day may be set or the flushing is ended when another threshold is met.

Flushing can be simulated for outlet lakes where the reservoir operation mode is not set to river sediments (`resop_mode == 4`). Flushing is not simulated in ilakes. Sediment density is assumed to be constant while sediment flushing is operating. The flushed sediment may be removed from the simulation or transported with the outflow to the downstream subbasin.

### Flush mode 1 - Capacity

Start to remove accumulated sediments from reservoir after available reservoir storage capacity drops below a specified threshold ( $sm\_start$ , a fraction). Flushing stops when available reservoir storage capacity (fraction) exceeds the second threshold which is calculated as the start level plus the capacity that is to be restored ( $sm\_caprest$ ). The flushing rate (capacity/day) is calculated as the capacity to restore spread over the number of days specified with variable  $sm\_duraday$ . Note that the actual duration of flushing for each event may vary depending on how much upstream sediment enters the reservoir while flushing is occurring delaying the restoration of the capacity.

### Flush mode 2 - Day of year

Start to remove accumulated sediments from reservoir at a specified day of the year ( $sm\_start$ ). The flushing may be every year or specified to occur at another frequency, e.g. every other year ( $sm\_yfreq$ ). Flushing stops after a duration given by  $sm\_duraday$ . Flushing rate (capacity/day) is calculated as storage capacity to flush, specified with variable  $sm\_caprest$  (fraction of capacity), over the duration of the flushing. Note that the amount of reservoir storage capacity that is restored for each event may vary depending on how much upstream sediment enters the reservoir while flushing is occurring. Note that all of the sediment may be flushed from the reservoir before the planned end date if the amount of sediment in the reservoir is not greater than or equal to the flushing rate times the number of days of flushing, basically there isn't enough sediment in the reservoir, so the reservoir runs out of sediment before the end of the flushing period.

## Links to file reference

Section	Symbol	Parameter/Data	File
Primary production and mineralization	$area, lakeregion$		<a href="#">GeoData.txt</a>
		$wprodn$	<a href="#">par.txt</a> or <a href="#">LakeData.txt</a>
		$hsatTP, limsedPP, tpmean$	<a href="#">par.txt</a>
Sedimentation in lakes	$par_{sed}$	$sedss, sedae$	<a href="#">par.txt</a> or <a href="#">LakeData.txt</a>
	$lim$	$limsedSS$	<a href="#">par.txt</a>
Sedimentation and resuspension in rivers		$sedexp, qbank, vpeak, addsusp$	<a href="#">par.txt</a>
	$k, m$	$hygeomk, hygeommm$	
	$spcon, spexp$	$suspconss, suspexpss$	<a href="#">par.txt</a> or <a href="#">GeoData.txt</a>
	$suspch$	$suspch, suspchannel$	
Siltation		$ldseddens$	<a href="#">par.txt</a>
	$P_{sand}, P_{silt}, P_{clay}$	$sand, silt, clay$	<a href="#">GeoData.txt</a>
	$resop\_mode$	$sm\_resop$	<a href="#">LakeData.txt</a> <a href="#">DamData.txt</a>
		$sm\_mode, sm\_dest$	
	$sm\_start, sm\_duraday, sm\_caprest, sm\_yfreq$	$sm\_start, sm\_duraday, sm\_caprest, sm\_yfreq$	

## Link to relevant processes in the code

Modules (file)	Procedures	Sections
<a href="#">npc_surfacewater_processes (npc_sw_proc.f90)</a>	$substance\_processes\_in\_river$	Sediment in rivers and lakes
	$substance\_processes\_in\_lake$	
	$production\_mineralisation$	Primary production and mineralization
	$calculate\_lake\_tpmean$	
	$calculate\_river\_tpmean$	Sedimentation in lakes
	$lake\_sedimentation$	
	$river\_sedimentation\_resuspension$	

## Diagnostic output variables - sediment load

The load of sediment leaving a subbasin is calculated for suspended sediments (cSSL) and total sediments (cTSL).

Net load of main river and outlet lake can be calculated for suspended sediments (SS) and total sediments (TS). The output variables ID are nLSS and nLTS. Net load of main river and outlet lake is calculated from the inflow and outflow of substance to the main river and outlet lake in a subbasin as one system. The inflow is composed of upstream coming inflow and local inflow to the main river. The outflow is the outflow of the subbasin through the main and secondary branch. The net load is the load of outflow minus the load of inflow. A bit of caution, do not mistake this output as the net load from a catchment, i.e. the load in the main river leaving the catchment.

Included in the net load is the effect of production/mineralisation of algae in the lake (for TS), point sources. Reduction of the net load is the effect of sedimentation, but also abstraction of water for irrigation or other purposes. In the case of zero outflow, or if the incoming load is larger than the load leaving the system, the net load becomes negative. Negative load could thus be caused by e.g. sedimentation, but also temporary holding up the sediment in the system. For subbasins holding a part of a lake, i.e. a lake basins, it is also possible that the water flow between the lake basins is (temporary) opposite of what is defined as downstream in GeoData. In this case the outflow of the subbasin is an inflow, and is withdrawn from the net load. Thus the net load of main river and outlet lake become negative.

## References

Lidén, R., 1999. A new approach for estimating suspended sediment yield, *HESS*, 3(2):285-294.

Lidén, R., J. Harlin, M. Karlsson, and M. Rahmberg, 2001. Hydrological modelling of fine sediments in the Odzi River, Zimbabwe, *Water SA*, 27(3): 301-314.

R. P. C. Morgan, D. D. V. Morgan, and H. J. Finney, 1984. A predictive model for the assessment of soil erosion risk, *Journal of Agricultural Engineering Research*, vol. 30, pp. 245-253.

Neitsch, S.L., J.G. Arnold, J.R. Kiniry and J.R. Williams, 2011. "Soil and Water Assessment Tool Theoretical Documentation Version 2009", Texas Water Resource Institute Technical Report No. 406, Texas A&M University, Texas.

Strand, R.I. and E.L. Pemberton, 1982. "Reservoir Sedimentation", Technical Guidelines for Bureau of Reclamation, Denver, Colorado.



# Silica

The silica routines were first released in HYPE version 5.15.0. Two new substances were introduced; dissolved silica (DS) and algal silica (AS). Together they form the sum of silica (SI).

## Source of silica

### Weathering

Soil chemical weathering is the source of (dissolved) silica in the model. *Weathering* (mg/m<sup>2</sup>) in the model is a function of soil temperature, depth of soil layer, soil type and geology. The equation is

$$Weathering = rate \times f_{depth} \times f_{temp} \times weathcorr \times soilthick$$

where *rate* is a soil type dependent weathering rate parameter (mg/m<sup>3</sup>), and *soilthick* is the thickness of each soil layer.

Silica weathering is dependent on soil organic acids, which are more prevalent in the upper humus rich soil layer. Therefore a function of soil depth was deployed, assuming an exponential decline with soil depth (*d*).

$$f_{depth} = 2^{-\frac{d}{weathdep}}$$

where the depth is calculated from soil surface to the middle of the soil layer. *weathdep*, a general model parameter (m), is the depth where the activity is half of that at the surface.

The weathering is also temperature dependent with the activity increasing with increasing temperatures. It is implemented in the model accordingly:

$$f_{temp} = e^{-\frac{weathKT}{R} \times \left( \frac{1}{T} - \frac{1}{T_0} \right)} \times frozensoilcorr$$

where *weathKT* (J/mol) is the rock activation energy (general model parameter), *R* is the universal gas constant, *T* is the soil temperature of the soil layer (Kelvin) and *T<sub>0</sub>* is a reference temperature. To account for lower water-soil contact during frozen or partly frozen soil conditions, the weathering is reduced when soil temperature is below 0 °C according to a correction factor:

$$frozensoilcorr = 1 - \frac{soiltemp}{(soiltemp - 1)}$$

where *soiltemp* is the soil temperature (this time in °C).

The factor *weathcorr* is subbasin weather correction factor that may be used to account for spatial differences in weathering that is not explained by the other factors, such as for example lithology.

## Point sources

It is technically possible to add pointsources for dissolved silica and algae silica.

## Links to relevant procedures in the code

Modules (file)	Procedures
<a href="#">npc_soil_processes (npc_soil_proc.f90)</a>	weathering

## Soil pool initial values

The initial concentration of dissolved (and algae) silica in soil is zero.

## Rivers and lakes

The initial silica concentration in rivers are assumed to be zero, while the lakes' concentration are set by the user. The parameter, *iniSi*, is general.

## Primary production and mineralization

The dissolved silica and the algal silica interact. Net uptake of DS and production of AS occur during spring and early summer in a similar way as for other nutrients. The process is reversed during late summer and early autumn. This net production is simulated as a function of water temperature and phosphorus concentration in the water body.

The production/mineralization depend on temperature and total phosphorus and lake area or river bottom area (*area*). The potential carbon transformation (*minprodSIpot*, kg / day) is calculated with the following equations:

$$tmpfcn1 = \left( \frac{T}{20} \right)^{\exp}$$

$$tmpfcn2 = \frac{(T_{10} - T_{20})}{5}$$

$$tmpfcn = tmpfcn1 \times tmpfcn2$$

$$TPfcn = \frac{(TPconc - plimSi)}{(TPconc - plimSi + hsatTP)}$$

$$minprodSIpot = wprodsi \times TPfcn \times tmpfcn \times area \times depth$$

where *T* is the water temperature(°C) and *T10* and *T20* are the average water temperatures of the

last 10 and 20 days, respectively. *exp* is a general model parameter (*sitmpexp*). *TPconc* is the current total phosphorus concentration in the waterbody and *plimSi* and *hsatTP* are general model parameters. The water depth (*depth*) is the current lake water depth, and for the river the depth use is calculated as described [here](#). The calculated mineralization of algal silica is limited to a maximum of 50% of the available algae pool, and the production is limited to a maximum of 50% of the available dissolved silica pool.

## Sedimentation

Sedimentation in lakes is a sink for algae silica and works the same way as for organic nitrogen and particulate phosphorus. Sedimentation (*sedSI*, *kg/day*) is calculated as a function of AS concentration in lake water (*conc*) and lake area (*area*). The settling velocity parameter *sedsi* is general or can be specified for each lake.

$$sedSI = sedsi \times conc \times area$$

## Links to relevant procedures in the code

Modules (file)	Procedures	Sections
<a href="#">npc_surfacewater_processes</a> ( <a href="#">npc_sw_proc.f90</a> )	substance_processes_in_river	primary production and mineralization
	production_mineralisation	
	substance_processes_in_lake	primary production and mineralization
	lake_sedimentation	sedimentation

## Links to file reference

Section	Symbol	Parameter/Data	File
Sources of silica	<i>weathcorr</i>	<i>weathcorr</i>	<a href="#">GeoData.txt</a>
	<i>rate</i>	<i>weathrate</i>	<a href="#">par.txt</a>
	<i>weathKT, weathdep</i>	<i>weathKT, weathdep</i>	
		<i>iniSi</i>	
	<i>soilthick</i>		<a href="#">GeoClass.txt</a>
Rivers and lakes	<i>area, lakeregion</i>		<a href="#">GeoData.txt</a>
	<i>wprodsi, sedsi</i>	<i>wprodsi, sedsi</i>	<a href="#">par.txt</a> or <a href="#">LakeData.txt</a>
	<i>plimSi, hsatTP</i>	<i>plimsi, hsatTP</i>	<a href="#">par.txt</a>
	<i>exp</i>	<i>sitmpexp</i>	